



UI5 tooling

<https://github.com/SAP/ui5-tooling>

EVOLUTION

- Maven
 - SAP internal only
 - Java based



- Grunt
 - Open Source
 - Grunt Task Runner based



- New ui5-tooling
 - Open Source
 - Node.js based

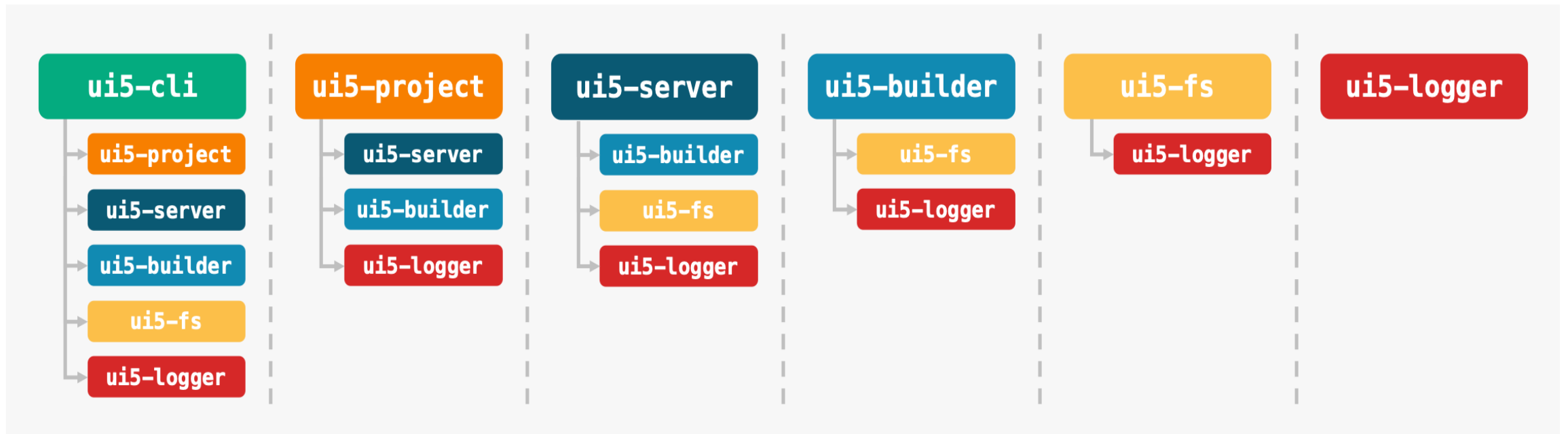




WHAT FOR?

- Default build tool in SAP Web IDE
- Use for local developments
- Integrate with CI/CD pipelines
- Extend ui5-builder with Custom Tasks
- Extend ui5-server with Custom Middleware
- Dependency management
- Consume NPM modules

MODULES



<https://github.com/SAP/ui5-tooling>

ui5-cli

<https://github.com/SAP/ui5-cli>

- UI5 Command Line Interface
- Commands:
 - **build**
 - Build project in current directory
 - **serve**
 - Start a webserver for the current project
 - **tree**
 - Outputs the dependency tree of the current project to stdout. It takes all relevant parameters of ui5 build into account.
 - **init**
 - Initializes the UI5 Tooling configuration for an application or library project

ui5-server

<https://github.com/SAP/ui5-server>

\$ ui5 serve [options]

- UI5 Server provides web server capabilities for the current project
- Supports HTTP/2
- Certificates for HTTPS or HTTP/2
- Extensible, custom middleware

ui5-server

Extensibility: custom middleware

<https://sap.github.io/ui5-tooling/pages/extensibility/CustomServerMiddleware>

ui5.yaml x

```
1 specVersion: "1.0"
2 type: application
3 metadata:
4   name: my.application
5 server:
6   customMiddleware:
7     - name: myCustomMiddleware
8       mountPath: /myapp
9       afterMiddleware: compression
10  configuration:
11    debug: true
```

custom-middleware.js x

```
1 /**
2  * Custom UI5 Server middleware example
3  *
4  * @param {Object} parameters Parameters
5  * @param {Object} parameters.resources Resource collections
6  * @param {module:@ui5/fs.AbstractReader} parameters.resources.all Reader or Collection to read resource
7  *                                     root project and its dependencies
8  * @param {module:@ui5/fs.AbstractReader} parameters.resources.rootProject Reader or Collection to read
9  *                                     the project the server is started in
10 * @param {module:@ui5/fs.AbstractReader} parameters.resources.dependencies Reader or Collection to read
11 *                                     the projects dependencies
12 * @param {Object} parameters.options Options
13 * @param {string} [parameters.options.configuration] Custom server middleware configuration if given
14 * @returns {function} Middleware function to use
15 */
16 module.exports = function({resources, options}) {
17   return function (req, res, next) {
18     // [...]
19   }
20 };
```


ui5-builder

<https://github.com/SAP/ui5-builder>

\$ ui5 build [options]

- UI5 Build produces production-ready folder/files from your project
- Supports application | library | module
- Certificates for HTTPS or HTTP/2
- Extensible, custom tasks

ui5-builder

SAP Web IDE Full-Stack

<https://help.sap.com/viewer/825270ffffe74d9f988a0f0066ad59f0/CF/en-US/a3d3182aa1044cf3a0aa577158a977d6.html>

package.json x

```
1 {
2   "name": "neo-ui5",
3   "version": "0.0.1",
4   "description": "",
5   "private": true,
6   "devDependencies": {
7     "@ui5/cli": "1.4.3",
8     "@sap/ui5-builder-webide-extension": "1.0.4"
9   },
10  "scripts": {
11    "build": "ui5 build --include-task=generateManifestBundle generateCachebusterInfo"
12  },
13  "ui5": {
14    "dependencies": [
15      "@sap/ui5-builder-webide-extension"
16    ]
17  }
18 }
```

ui5.yaml x

```
1 specVersion: '1.0'
2 metadata:
3   name: neo-ui5
4   type: application
5   builder:
6     customTasks:
7       - name: webide-extension-task-updateNeoApp
8         afterTask: generateVersionInfo
9         configuration:
10           destDir: dist
11           appFolder: webapp
12           namespace: mlauffer
13       - name: webide-extension-task-updateManifestJson
14         afterTask: webide-extension-task-updateNeoApp
15         configuration:
16           appFolder: webapp
17           destDir: dist
18       - name: webide-extension-task-lint
19         afterTask: webide-extension-task-updateManifestJson
20         configuration:
21           destDir: dist
22           appFolder: webapp
23           namespace: mlauffer
24       - name: webide-extension-task-resources
25         afterTask: webide-extension-task-lint
26         configuration:
27           namespace: mlauffer
```

ui5-builder

Extensibility: custom tasks

<https://sap.github.io/ui5-tooling/pages/extensibility/CustomTasks>

ui5.yaml x

```
1 # In this example configuration two custom tasks are defined:
2 # 'babel' and 'generateMarkdownFiles'.
3 specVersion: "1.0"
4 type: library
5 metadata:
6   name: my.library
7 builder:
8   customTasks:
9     - name: babel
10       beforeTask: generateComponentPreload
11     - name: generateMarkdownFiles
12       afterTask: uglify
13   configuration:
14     color: blue
```

custom-task.js x

```
1 /**
2  * Custom task example
3  *
4  * @param {Object} parameters Parameters
5  * @param {module:@ui5/fs.DuplexCollection} parameters.workspace DuplexCollection to read and
6  * @param {module:@ui5/fs.AbstractReader} parameters.dependencies Reader or Collection to read
7  * @param {Object} parameters.options Options
8  * @param {string} parameters.options.projectName Project name
9  * @param {string} [parameters.options.configuration] Task configuration if given in ui5.yaml
10  * @returns {Promise<undefined>} Promise resolving with <code>undefined</code> once data has been processed
11  */
12 module.exports = async function({workspace, dependencies, options}) {
13   // [...]
14 };
```

ui5-migration

<https://github.com/SAP/ui5-migration>

\$ ui5-migration [options]

- Optimizes outdated code according to UI5 best practices
- Replaces deprecated APIs
- Still in BETA
- Modified code needs manual review (kinda buggy yet)