Move your VS Code extension into Eclipse Che

Florent Benoit
Eclipse Che 7
Devfile to share workspaces

```
apiVersion: 1.0.0
metadata:
  generateName: python-projects:
    - name: python-hello-world
      source:
        type: git
        location: "https://github.com/che-samples/python-hello-world.git"
components:
  - type: chePlugin
    id: ms-python/python/latest
    memoryLimit: 512Mi
  - type: dockerimage
    alias: python
    image: quay.io/eclipse/che-python-3.7:nightly
    memoryLimit: 512Mi
    mountSources: true
commands:
  - name: run
    actions:
      - type: exec
        component: python
        command: "python hello-world.py"
        workdir: ${CHE_PROJECTS_ROOT}/python-hello-world
```
Che 7: swappable editor
Registries in Eclipse Che

- plug-in registry
- devfile registry
- workspace
Manage all plugins including VS Code extensions
Eclipse Che 7: The new, the noteworthy and the future plans!

This afternoon: 5PM
Seminarraum 5
Develop VS Code extensions
How to start?

- **Scaffolding: Yeoman generator**
  - `yo code`

- **Develop**
  - Intellisense from Eclipse Che IDE

- **Run**
  - Run inside Eclipse Che IDE

- **Debug**
  - Run inside Eclipse Che IDE
VS Code extensions hosted mode

Hosted / development mode

Two instances of Theia

- Main instance to write plug-ins
- A new instance to run plug-ins
Demo

Demo #1

Develop and debug VSCode extension

youtube
Publish VS Code extensions on Eclipse Che registries
Registries in Eclipse Che

- plug-in registry
- devfile registry
- workspace
---
apiVersion: 1.0.0
metadata:
  name: my-che-workspace
components:
  -
    type: chePlugin
    id: redhat/java11/latest

https://che-plugin-registry.openshift.io/v3/plugins/
apiVersion: 1.0.0
metadata:
  name: my-che-workspace
components:
- type: chePlugin
- reference: https://my-registry/plugins/florent/hello/0.1/meta.yaml
Create its own registry

Build a custom registry based on existing one

- Add / update / remove plug-ins
- Rebuild docker image
Create its own registry on github

Or use a simple github project
Demo #2

VSCode extension with custom plugin registry

youtube
VS Code extensions and Eclipse Che
ECLIPSE CHE

**BENEFITS**

- **Eclipse Che Workspaces**
  - plug-ins per workspace

- **Plug-in Registry**
  - Global / per instance

- **Multi-user / auth**
  - OpenID Connect / LDAP / Keycloak

- **Cloud Ready**
  - Kubernetes / OpenShift
@eclipse-che/plugin namespace

- Interact easily with Eclipse Che API
  

```typescript
export namespace workspace {
  export function getCurrentWorkspace(): Promise<cheApi.workspace.Workspace>;
  export function getAll(): Promise<cheApi.workspace.Workspace[]>;
  export function getAllByNamespace(namespace: string): Promise<cheApi.workspace.Workspace[]>;
  export function getById(workspaceId: string): Promise<cheApi.workspace.Workspace>;
  export function create(config: cheApi.workspace.WorkspaceConfig, params: KeyValue): Promise<any>;
  export function update(workspaceId: string, workspace: cheApi.workspace.Workspace): Promise<any>;
  export function deleteWorkspace(workspaceId: string): Promise<any>;
  export function start(workspaceId: string, environmentName: string): Promise<any>;
  export function startTemporary(config: cheApi.workspace.WorkspaceConfig): Promise<any>;
  export function stop(workspaceId: string): Promise<any>;
  export function getSettings(): Promise<KeyValue>;
}
```

- Simple code to use

```typescript
import * as che from '@eclipse-che/plugin';

... await che.workspace.getCurrentWorkspace();
```
System Runtime dependencies / Language server protocol
Java LSP needs Java, PHP requires PHP, etc

Tools dependencies
$ apt-get install / yum install

Native libraries
ECLIPSE CHE

Standalone VS Code

Java installed on your system

Eclipse Che with containers

Java 11 container
Plugin 1
Che IDE Endpoint

Java 13 container
Plugin 2
Che IDE Endpoint

Main IDE Container
Plugin 3

JSON RPC
How sidecar endpoint is working?

Che Remote Theia endpoint

```bash
FROM eclipse/che-theia-endpoint-runtime:next
RUN apk --no-cache add openjdk11 -repository=http://dl-cdn.alpinelinux.org/alpine/edge/community
ENV JAVA_HOME /usr/lib/jvm/default-jvm/
FROM eclipse/che-theia-endpoint-runtime:next
RUN apk --no-cache add fortune
```
Che Remote Theia endpoint: some limitations

- Needs to inherit from a dedicated Docker image
- Workaround with multi-staged builds
- Strong link between Theia image and sidecar images
  - (better to use same version of theia on main theia and sidecars)
Che Remote Theia endpoint: some limitations

- Needs to inherit from a dedicated Docker image
- Workaround with multi-staged builds
- Strong link between Theia image and sidecar images
  - (better to use same version of theia on main theia and sidecars)
Upcoming Eclipse Che 7.4 : endpoint improvements

- Create a single all-in-one binary for this endpoint
  - Using nexe tool on Eclipse Che Theia remote package
- Copy this binary and run it
- Allow to use existing dockers image from docker registries

```yaml
FROM docker.io/openjdk:14-alpine
```

- ⇒ no need to build images, use image name in plugin’s meta.yaml
apiVersion: v2
publisher: redhat
name: vscode-xml
version: latest
type: VS Code extension
displayName: XML
title: XML Language Support by Red Hat
description: This VS Code extension provides support for creating and editing XML documents, based on the LSP4XML language server, running with Java.
icon: https://www.eclipse.org/che/images/logo-eclipseche.svg
repository: https://github.com/redhat-developer/vscode-xml
category: Language
firstPublicationDate: '2019-04-19'

containers:
- image: "docker.io/eclipse/che-remote-plugin-runner-java11:next"
  name: vscode-xml
  memoryLimit: "768Mi"

extensions:
- https://github.com/redhat-developer/vscode-xml/releases/download/0.7.0/vscode-xml-0.7.0-3205.vsix
Demo #3

VSCode extension with custom runtime

youtube
Status
<table>
<thead>
<tr>
<th>Namespace/Not</th>
<th>Theta master</th>
<th>Theta v0.11.0</th>
<th>Theta v0.10.0</th>
<th>VSCODE master</th>
<th>VSCODE 1.36.0</th>
<th>VSCODE 1.36.1</th>
<th>VSCODE 1.36.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Breakpoint</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BreakpointsChangeEvent</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CancellationToken</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CharacterPair</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Clipboard</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CodeAction</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CodeActionContext</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CodeActionKind</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CodeActionProvider</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CodeActionProviderMetadata</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CodeLens</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

VS Code extensions

https://che-incubator.github.io/vscode-theia-comparator/status.html
Theia-vscodecov checker

https://github.com/theia-ide/theia-vscodecov

$ cd my-vscode-extension
$ npx theia-vscodecov
Conclusion
• Improve VS Code API compliance
• Alternate VS Code extension marketplace
Get started with Eclipse Che

Sources
github.com/eclipse/che

Dev List
che-dev@eclipse.org

Docs
http://eclipse.org/che/docs

Mattermost
https://mattermost.eclipse.org

Try it out : https://che.openshift.io
Thank you

Red Hat is the world’s leading provider of enterprise open source software solutions. Award-winning support, training, and consulting services make Red Hat a trusted adviser to the Fortune 500.