



Eclipse Che / CodeReady Workspaces

L'IDE pour coder vos applications cloud native

Sun Tan
Che Software Engineer

Stevan Le Meur
Developer Tools PM

WHERE WE ARE TODAY



Development is done on each developers' desktop.



Production is moving to a Kubernetes distribution like OpenShift.

Vast majority of an organization's developers are not experts at containers or Kubernetes.

WHERE WE ARE TODAY: OPTIONS



Development is done on each developers' desktop.



Production is moving to a Kubernetes distribution like OpenShift.

Option 1

Train developers on Containers and Kubernetes



Long learning curve with a small number of developers who will become experts.

Option 2

Make containers and Kube “invisible” to the developers



Short learning curve. Expert developers can continue to use OpenShift directly.

THE KEY QUESTION

What's needed to allow an authorized developer to contribute to a project without deep containers and Kubernetes knowledge?

...that's more complicated than expected.

WHAT DEVELOPERS NEED TO CONTRIBUTE

- Project sources
- Dependencies
- Developer Tools: language servers, debuggers, testing tools, security tools, etc...
- Commands
- Build and packaging tools
- Terminal
- Operating system
- Web server / application server
- Database
- (All other runtime components)

Everything is versioned and needs updating.

If anything is different across the team inconsistent behaviors can result.

TRADITIONAL LAPTOP APPROACH

**Shareable
with all**

- Project sources
- Dependencies

**Shareable
with some**

- Developer Tools: language servers, debuggers, testing tools, security tools, etc...
- Commands
- Build and packaging tools
- Terminal

**Hard to
share
with
anyone**

- Operating system
- Web server / application server
- Database
- (All other runtime components)

A laptop makes it hard to share and secure everything a developer needs.

THE ECLIPSE CHE / CODEREADY WORKSPACES METHOD

Managed in a
containerized
Workspace
hosted in an
IT-Managed
OpenShift
cluster.

- Project sources
- Dependencies
- Developer Tools: language servers, debuggers, testing tools, security tools, etc...
- Commands
- Build and packaging tools
- Terminal
- Operating system
- Web server / application server
- Database
- (All other runtime components)

Easy to share and secure across the team.

Eclipse Che / CodeReady Workspaces

Make developing container-based applications and services on Kubernetes platforms easy.

1. **Accelerates projects** and onboarding of developers.
2. **Removes inconsistencies** and “it works on my machine...” delays.
3. **Protects source code** by removing it from hard-to-secure laptops.

Key use cases

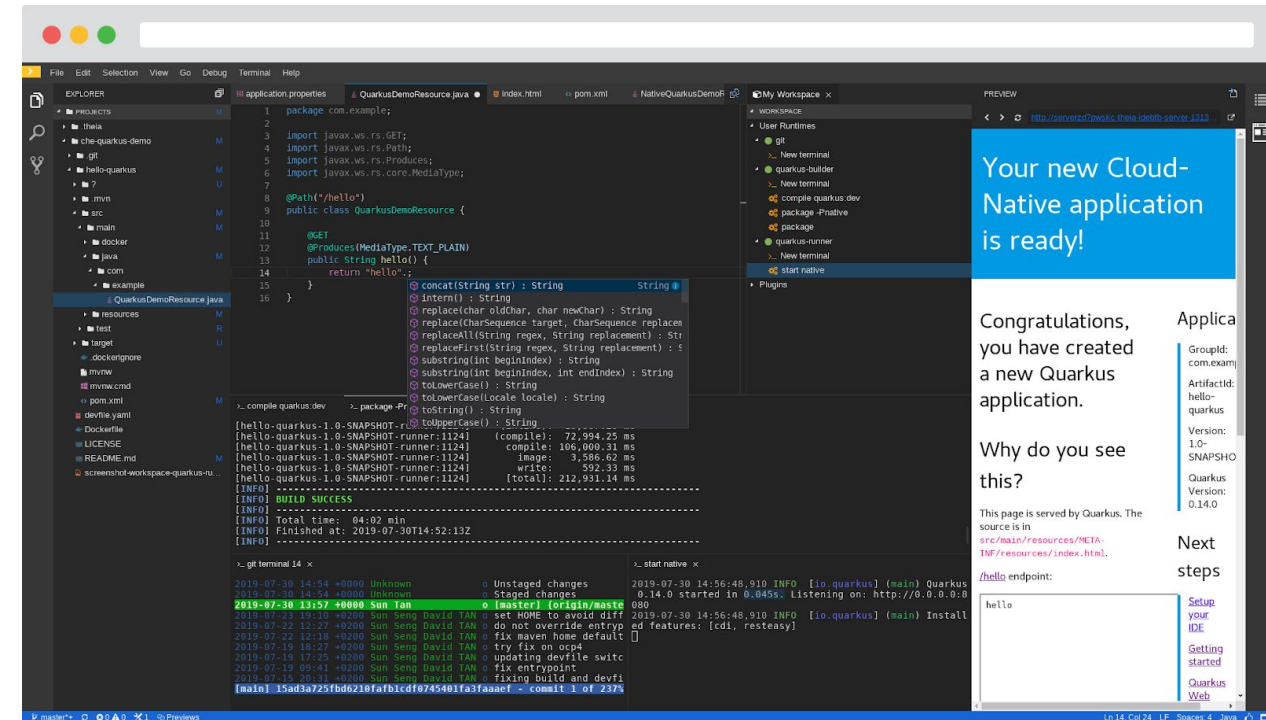
- **Accelerate container development** for teams that aren't yet Kubernetes experts
- **Replace slow VDI** for “untrusted” developers working in the company
 - Third-party consultants
 - High turnover / offshore developers
- **Accelerate “task-based” (not daily) use cases**
 - Diagnosing and fixing failed CI jobs
 - Fixing issues in legacy codebases
 - Stakeholder or QA/Docs reviews of new code

Che 7

CodeReady Workspaces 2

Eclipse Che 7 / CodeReady Workspaces 2

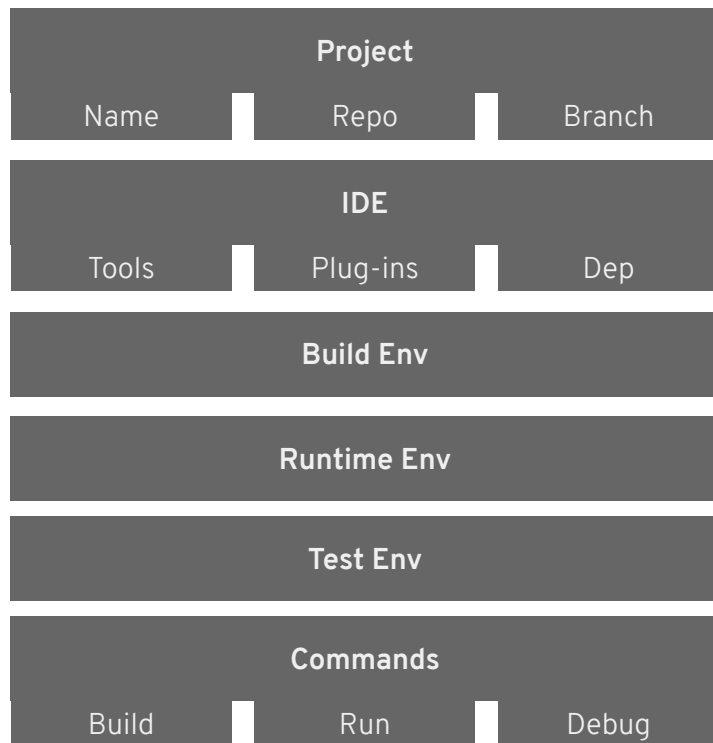
- **Kubernetes-based developer workspaces:** containerized developer workspaces include the project kubernetes application runtime in the development environment.
- **New Editor:** New default web-based editor provides a VSCode like experience in the browser.
- **VSCode extension compatibility**
- **OpenShift VSCode Plug-in**
- **Devfile:** developer environments as code.
- **Easier to Monitor and Administrate:** Prometheus and Grafana dashboards.



CodeReady Workspaces 2.0 - Shipped independently from 4.2, shortly after

DEVFILE

Devfile: Developer environment as code



Devfile Definition

The devfile provides easy-to-configure, highly reproducible definitions of portable developer environments.

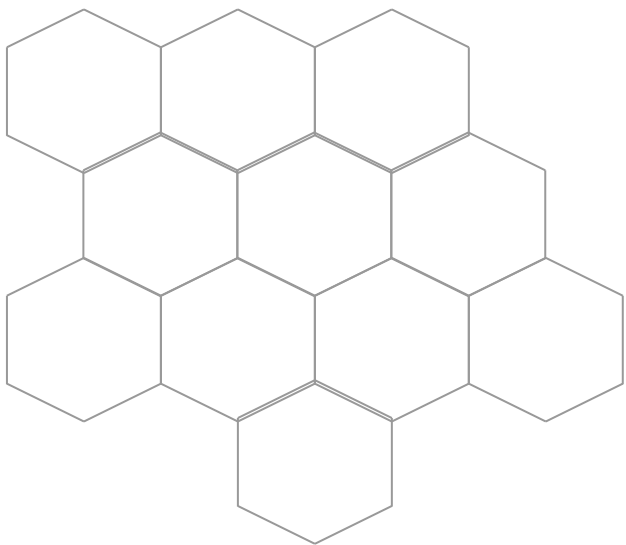
It is a declarative abstraction of a replicable developer workspaces, which includes the runtime environments, the source code of the projects mapped to repositories and the tools, plugins and commands needed to code, build, test, run and debug a project.

Devfile Example

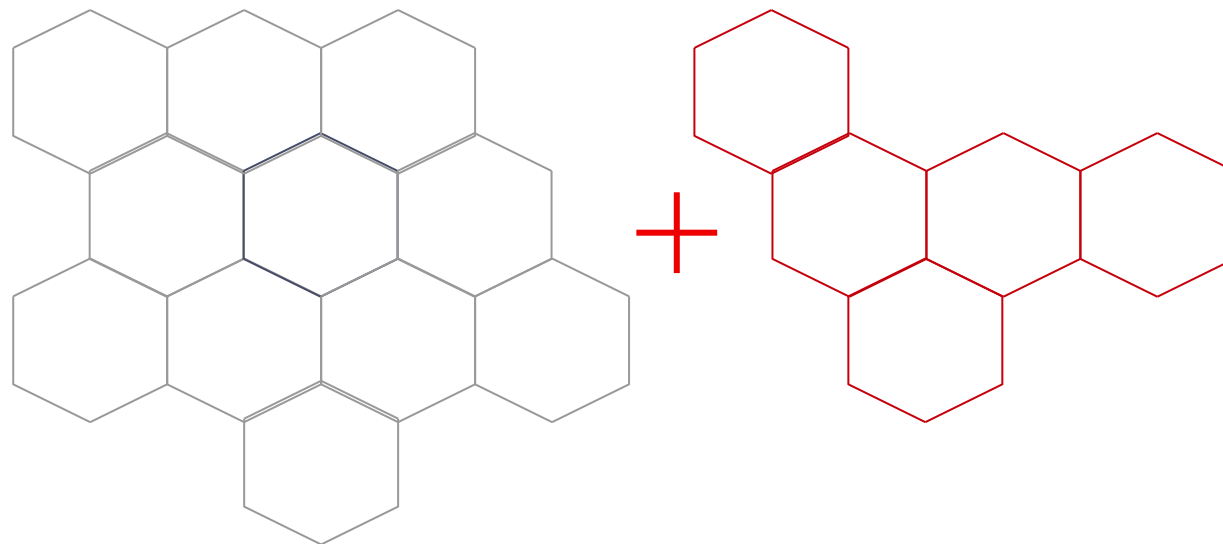
- 1 Project information
- 2 List of components of the workspace
- 3 Plugin component
- 4 Runtime image
- 5 Env variables to configure the container
- 6 End-points definition

```
---
apiVersion: 1.0.0
metadata:
  generateName: java-web-vertx-
1 projects:
  - name: java-web-vertx
    source:
      type: git
      location: "https://github.com/che-samples/web-java-vertx"
2 components:
3   - type: chePlugin
      id: redhat/java/latest
4   - type: dockerimage
      alias: maven
      image: quay.io/eclipse/che-java8-maven:nightly
5   env:
      - name: JAVA_OPTS
        value: "-Duser.home=/home/user"
      - name: MAVEN_OPTS
        value: "${JAVA_OPTS}"
      memoryLimit: 512Mi
6   endpoints:
      - name: '8080/tcp'
        port: 8080
      mountSources: true
      volumes:
        - name: m2
          containerPath: /home/user/.m2
```

“Dev mode” your kubernetes application

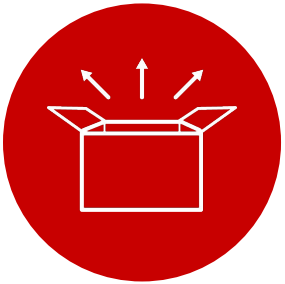


Use your K8S application
definition



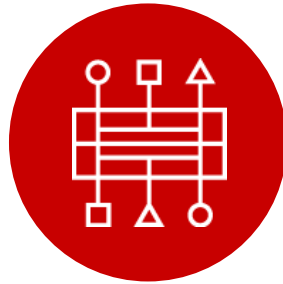
Supercharge your tools and start coding

Devfile: Made for Team



Manage Consistency

First class support of K8S, manage the complexity of developer environments take out the pain.



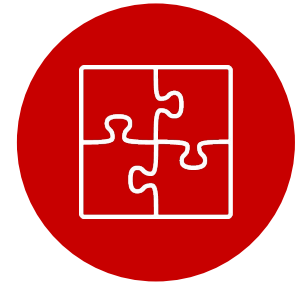
Easy to Integrate

Devfiles are easy to integrate with any tools. Developer environments can be created from anywhere at anytime.



Simple to Share

Devfile live with source code, are easy to modify, fork and share.



Extensible

Customize per task, with plug-in and developer preferences

DEMO



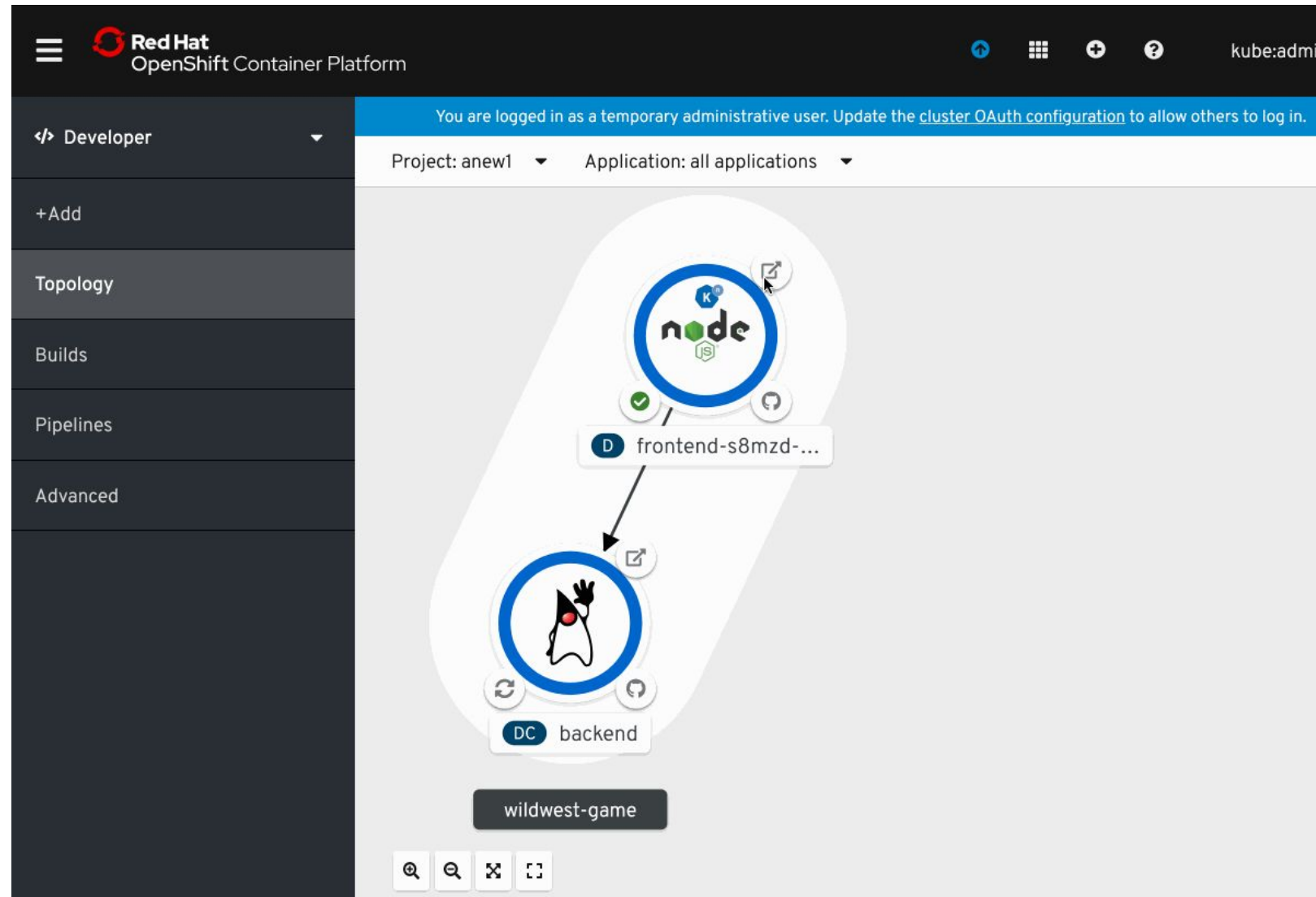
OPENSIFT 4.2 INTEGRATION

OpenShift Edit with CodeReady Workspaces


From the Developer perspective

-
- See the applications components
-


All OpenShift developer tool UIs will be surfaced here...though some (like Eclipse Che / CodeReady Workspaces) will be links out to unique UIs.



Thank you

 linkedin.com/showcase/red-hat-developer

 youtube - bit.ly/2YRIWTk

 facebook.com/redhatdeveloperprogram

 twitter.com/rhdevelopers