

# Red Hat CodeReady Workspaces

OpenShift Kubernetes-Based Developer Workspaces and IDE

David Johnston  
Solution Architect  
Red Hat, Inc.

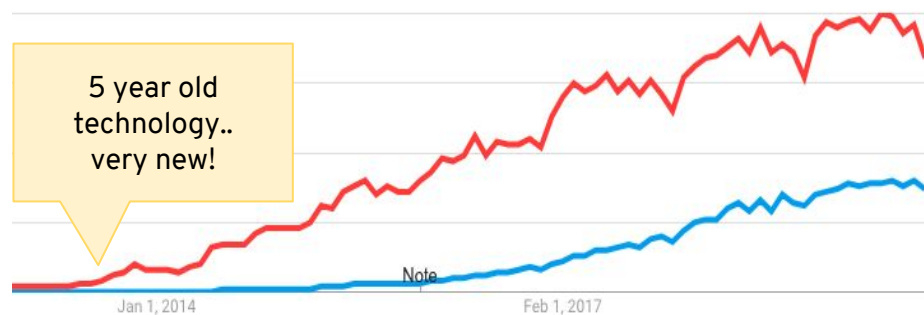
“>80% of my development teams aren't  
Kubernetes and container experts”

## Challenge

- Development happens on laptops: not in Kubernetes, sometimes in containers
- IT is moving to Kubernetes in production
- Devs have programming knowledge, but little experience with containers and Kubernetes
- Training every developer to become a container and Kube expert isn't viable: too hard, too long

**Need a way to make devs productive on Kube now,  
but enable their learning journey going forward**

Google Search Trend: Kubernetes and Linux Containers

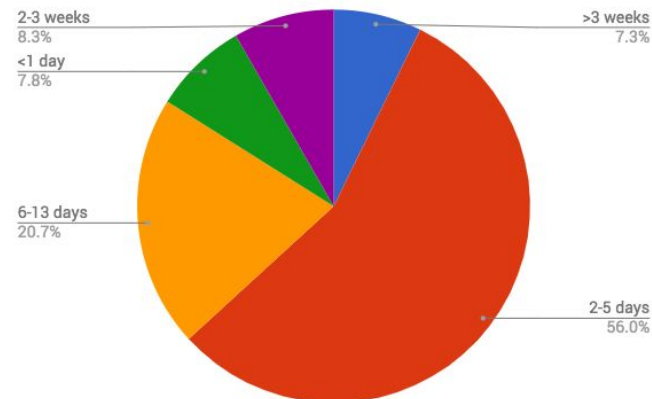


# “My remote development offices and development consultants aren’t as efficient as I need them to be”

## Challenges

- **Consultants often take 5-10 days to get setup** on internal systems
  - Get approved laptop
  - Get access to relevant tools
  - Get access to shared dev clusters
  - ...
- **Remote development offices are inefficient** due to high turnover and ongoing setup and ramp-up
  - VDI solutions are slow and painful for developers: makes them less agile and effective
- **Goals should be:**
  - Near-instant provisioning of a new consultant or developer
  - Ability to reproduce product environment in development (to speed releases and reduce “it works on my machine” issues)

Average Time To First Project Contribution



# CODEREADY WORKSPACES TEAMS SPEND MORE TIME CODING



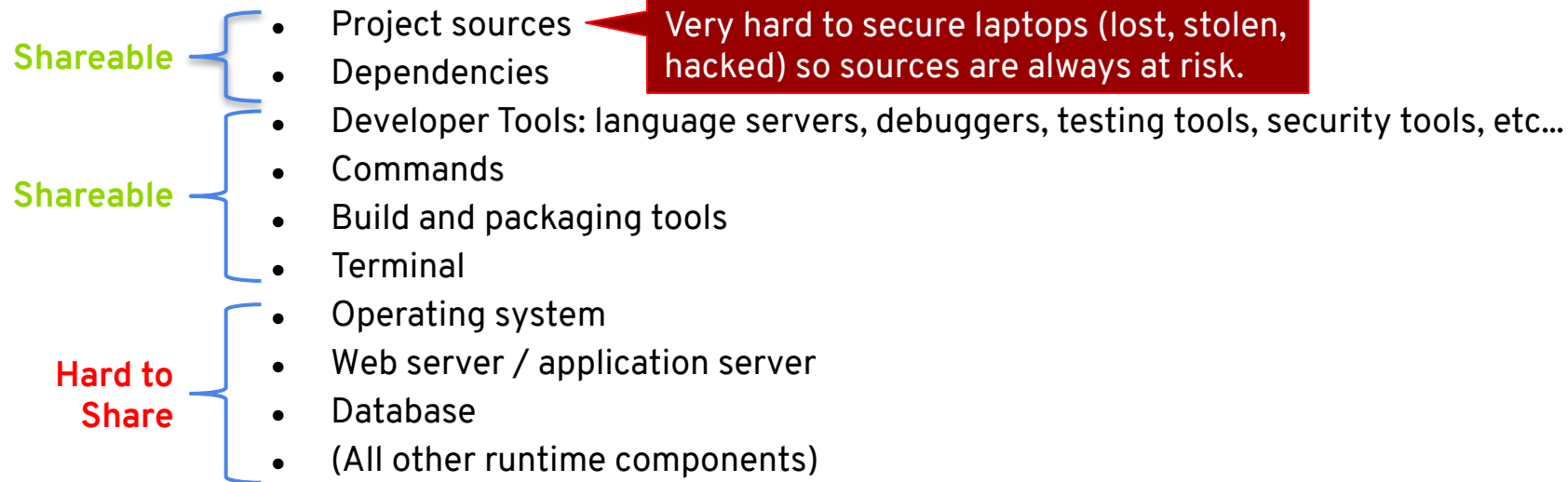
## 30% MORE TIME FOR CODING.



# RESULTS: MORE COMMITS, FASTER TURNAROUND FOR FIXES

	BEFORE	AFTER
COMMITTS / DAY / ENGINEER	5	20
FIX TO PRODUCTION (HOURS)	368	4

## Traditional IDE + Laptop Approach



**A laptop solution makes it hard to replicate, share and secure everything the developer needs.**

# CODEREADY WORKSPACES METHOD

Easy to Share  
and Secure

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)

# CODEREADY WORKSPACES

Make developing container-based applications and services on the Red Hat OpenShift Kubernetes platform 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.
4. **Eliminates the need** to run hypervisors on non-Linux laptops



# Red Hat CodeReady Workspaces 2.x

## a Cloud-Native IDE for OpenShift



### Kubernetes with Zero Effort

Embed a CodeReady Workspaces link in a project repo or issue tracker and anyone with a browser can be contributing code in <2 minutes.



### Protect your Code

Source code is never cloned to a hard-to-secure laptop. Code stays in an IT controlled sandbox, but one that reacts as quickly as a developer's laptop.



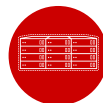
### Developer Environment as Code

Developer environments are codified with a Devfile: consistent; reproducible. Store them in the repo for auditability and to provide a GitOps experience.



### Openshift Developer Perspective Integration

Workspaces are accessible from the OpenShift console, making the onboarding even smoother.



### Air-Gap Install

Deploy on your OpenShift cluster, behind your firewall. Air-Gap capabilities. Easy to monitor and administrate.

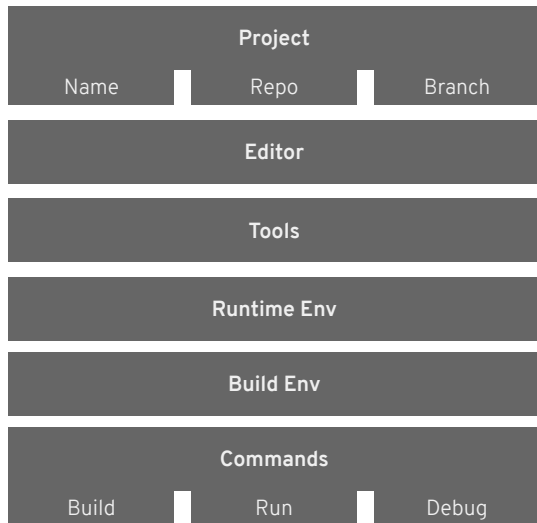


### New Built-in Editor

New browser based editor, providing a fast desktop like experience. Compatibility with Visual Studio Code extensions.

# 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

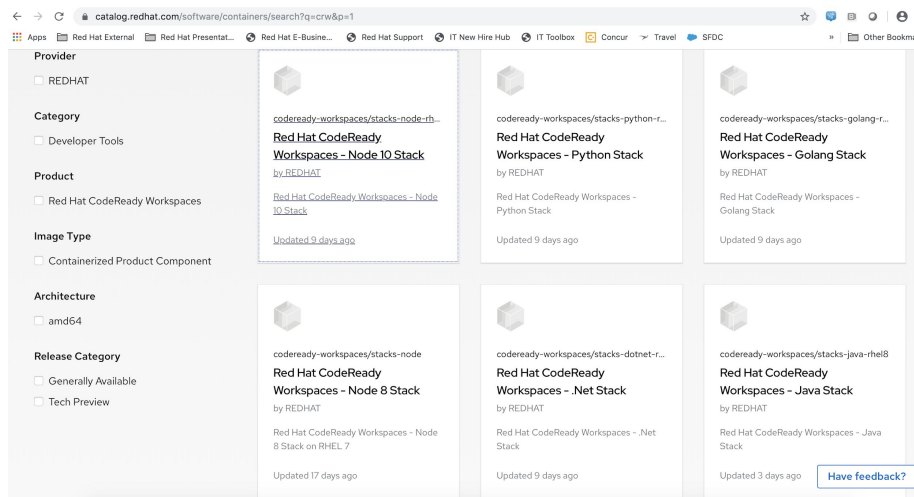
```
---
apiVersion: 1.0.0
metadata:
  generateName: java-web-vertx-
1 projects:
  - name: java-web-vertx
    source:
      type: git
      location: "https://github.com/che-samples/web-java-vertx"
2 components:
3 - type: chePlugin
  id: redhat/java/latest
4 - type: dockerimage
  alias: maven
  image: quay.io/eclipse/che-java8-maven:latest
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
```

# List of out-of-the-box stacks

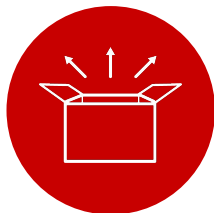
Versions

.NET	.Net 2.1
C/C++	C/C++
GO	Go 1.11.5
JAVA-EAP-MAVEN	OpenJDK 8, Maven 3.5, EAP 7.2
JAVA-MAVEN	OpenJDK 8, Maven 3.5
JAVA-SPRINGBOOT	OpenJDK 8, Maven 3.5
JAVA-VERTEX	OpenJDK 8, Maven.35
JAVA-THORNTAIL	OpenJDK 8, Thorntail 2.5
NODEJS	NodeJS 10
NODEJS-MONGODB	NodeJS 10, MongoDB 3.4
PHP	PHP 7.1
PYTHON	Python 3.6
RED HAT FUSE	

All stacks are built from the available images on the container catalog and UBI based.

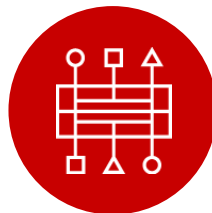


## Devfile: Made for Team



### Manage Consistency

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



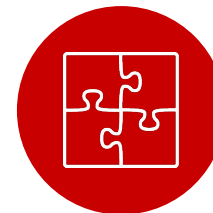
### Easy to Integrate

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



### Simple to Share

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



### Extensible

Customize per task, with plug-in and developer preferences

# The IDE Experience

# New Editor: VSCode in the browser

Containerizing the Editor: zero install and automate configuration

The screenshot displays a web-based IDE interface. The main area is a code editor showing a Java file named `QuarkusDemoResource.java`. The code includes imports for `javax.ws.rs.GET`, `javax.ws.rs.Path`, `javax.ws.rs.Produces`, and `javax.ws.rs.core.MediaType`. A `@GET` annotation is used to define a REST endpoint that returns the string "hello".

Below the code editor is a terminal window showing the output of a Maven build command. The output indicates a successful build for the `hello-quarkus-1.0-SNAPSHOT` project, including compilation and packaging steps.

On the right side of the interface, there is a sidebar with a blue header that reads "Your new Cloud-Native application is ready!". Below this, there is a "Congratulations" message and a "Why do you see this?" section. The sidebar also includes a "Next steps" section with a "Setup your IDE" link.

The interface also features a file explorer on the left, a terminal window at the bottom, and a sidebar on the right with a "Plugins" section.

Extended Eclipse Theia, to provide a top-of-the-art editor experience.

Built-in:

~ Languages Server Protocol

~ Debug Adapter Protocol

Compatible with VSCode extensions

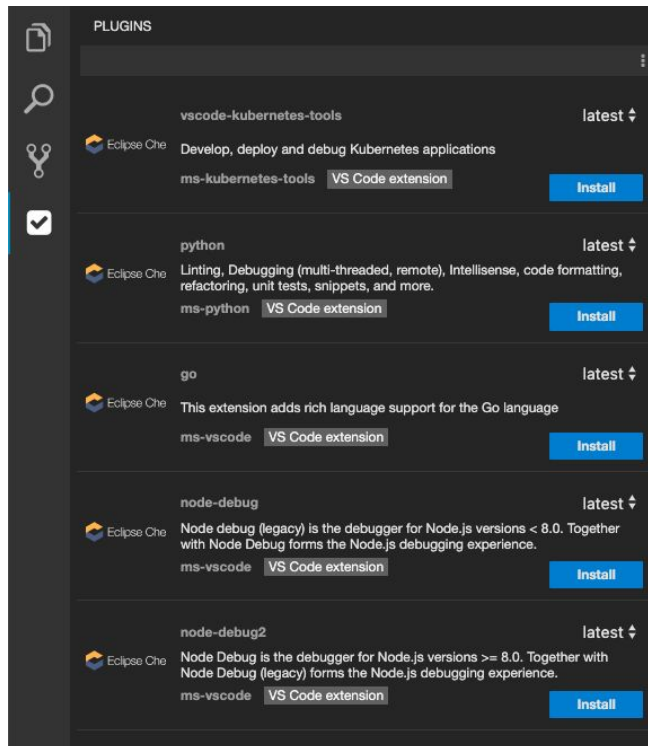


# List of Plug-ins

	Description	Version
JAVA LS	Java Language Support (Intellisense and Debugger)	
TYPESCRIPT LS	Typescript Language Support (Intellisense and Debugger)	
NODE DEBUG	Node >= 8.0 Support (Debugger)	
OPENSIFT CONNECTOR	Interact with an OpenShift Cluster	
KUBERNETES	Develop, deploy and debug Kubernetes applications	
DEPENDENCY ANALYTICS	Red Hat Dependency Analytics plug-in	
CAMEL TOOLING	Camel Language Support	
YAML LS	YAML Language Support	
XML LS	XML Language Support	

	Description	Version
OMNISHARP	.Net Language Support	
GO	GO Language Support	
PHP	PHP Language Support (Intellisense and Debugger)	
CPPTOOLS	C/C++ Language Support (Intellisense and Debugger)	
PROJECT INITIALIZER	A lightweight extension based on Red Hat launcher to generate	
WSDL2REST	Extension supporting the mapping of an existing SOAP service to a REST service.	
PYTHON	Python Language Support	

## Customize with VSCode extensions



- Visual Studio Code extensions compatibility
- Extensions packaged with their dependencies
- Plug-in registry with predefined set of plug-ins

**All Red Hat extensions available for VSCode will be available with CodeReady Workspaces 2.**

# Red Hat CodeReady Dependency Analytics

Analyze application stack for open source vulnerabilities, licenses, popularity & maintainability.

Analyze, Report, Update directly from IDE

Support for Java, Javascript, Python applications

The screenshot displays the Red Hat CodeReady Dependency Analytics interface within an IDE. The interface is divided into several sections:

- Security Issues:** Shows a summary of vulnerabilities. Total issues found: 3. Highest CVSS Score: 7.5 / 10. A progress bar indicates the score. Below, it shows the number of dependencies with this CVSS score: 2.
- Licenses:** Provides a recommended license for the application stack and lists potential license conflicts, unknown licenses, and restrictive licenses.
- Add-ons:** Lists complementary dependencies that can be added to the application stack.
- Dependency Details:** Provides a breakdown of analyzed dependencies, including analyzed, transitive, and unknown dependencies.

Below the summary cards, there is a section titled "Dependencies with security issues in your stack". It includes a table of direct dependencies with security issues:

# Dependencies	No. of CVE(s)	Highest CVSS Score	CVE ID of highest CVSS score
1	1	7.5/10	CVE-2017-18342

Details of the dependency: **pyyaml**

Affected Direct Dependency	Affected Version	GitHub Statistics:	Contributors	Forks	Dependent Repos	Stars	Usage
connexion	1.1.15		19	203	50593	731	NA

The interface also shows the CVE ID(s): CVE-2017-18342.

Enable developers to find and remedy issues without leaving source code editor



# Kubernetes-Based Workspaces

# Workspaces: Developer Sandbox centrally hosted on OpenShift

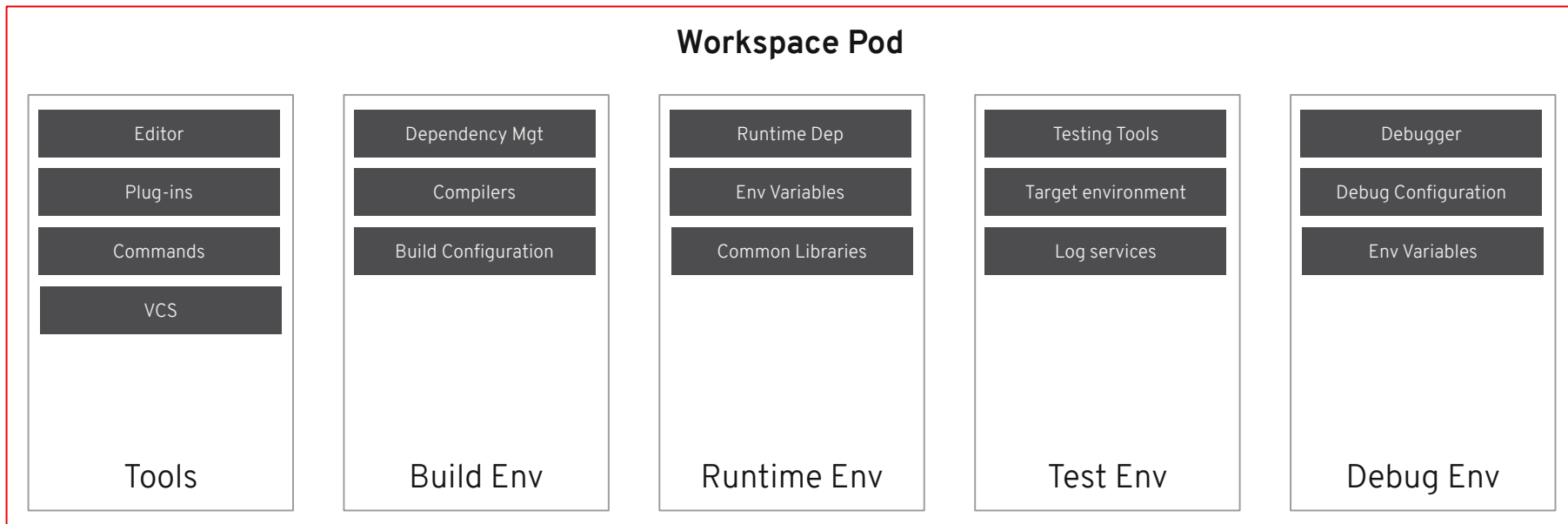
A CRW workspace is a **developer sandbox running on OpenShift**, where everything needed to code on a project is packaged into containers.

It provides:

- Application runtimes
- Build tools
- Development tools: Browser based editor + plugins
- Project source code repositories

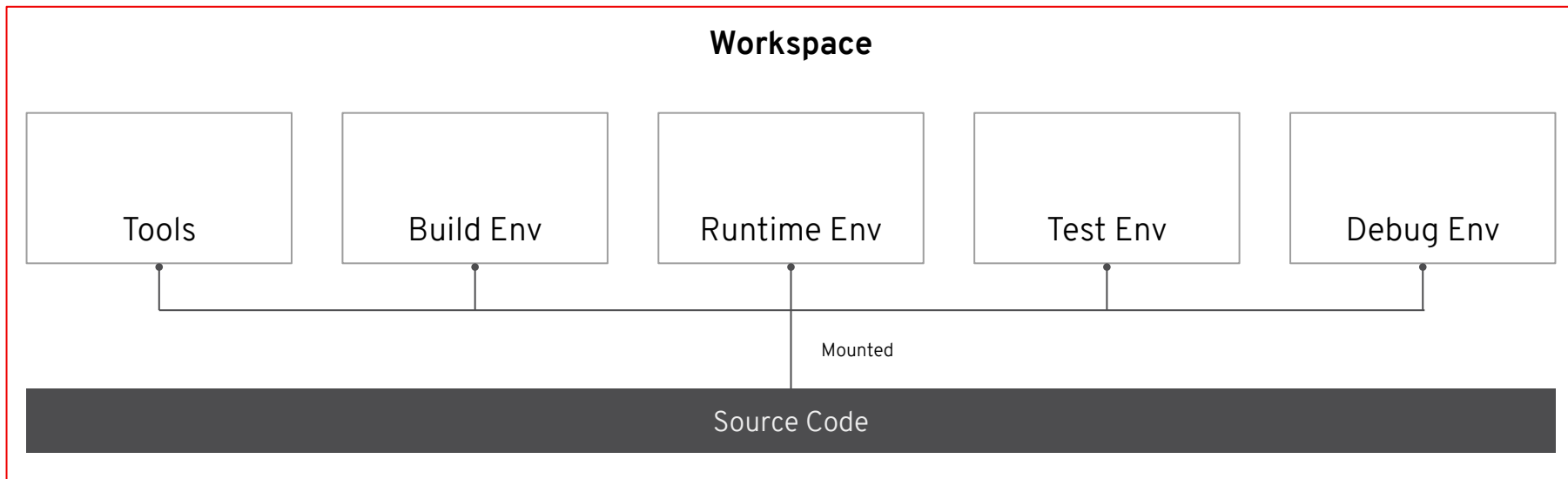
# Workspaces: Developer Sandbox

Containerizing everything you need to develop, build, run, test and debug your application.



## Workspaces: Developer Sandbox

Source code is getting mounted in your workspace, accessible from all containers.



# DEMO



# Install as Operator on OpenShift

You are logged in as a temporary administrative user. Update the [cluster OAuth configuration](#) to allow others to log in.

Project: workspaces ▾

- Ancore (0)
- Apicurio Project (0)
- AppDynamics (0)
- Appsody (0)

[Show 65 more](#)

CAPABILITY LEVEL

- Basic Install (7)
- Seamless Upgrades (7)
- Full Lifecycle (1)
- Deep Insights (0)
- Auto Pilot (0)

**Nexus Operator**  
provided by m88i Labs

Nexus Operator to deploy and manage Nexus 3.x servers

**OpenShift Pipelines Operator**  
provided by Red Hat

OpenShift Pipelines is a cloud-native CI/CD solution for building pipelines using

**OpenShift Serverless Operator**  
provided by Red Hat, Inc.

Provides a collection of API's based on Knative to support deploying and serving of

**Red Hat CodeReady Workspaces**  
provided by Red Hat, Inc.

A Kube-native development solution that delivers port:

✓ Installed

**Ripsaw**  
provided by Red Hat Performance

Ripsaw is a benchmark operator to benchmark k8s and certain applications.

**TriggerMesh**  
provided by TriggerMesh, Inc

A serverless management platform that runs on Knative. TriggerMesh provides continuous delivery of

TRY IT NOW

[HTTPS://CHE.OPENSIFT.IO](https://che.openshift.io)

# THANK YOU



[linkedin.com/company/red-hat](https://www.linkedin.com/company/red-hat)



[youtube.com/user/RedHatVideos](https://www.youtube.com/user/RedHatVideos)



[facebook.com/redhatinc](https://www.facebook.com/redhatinc)



[twitter.com/RedHat](https://twitter.com/RedHat)