Coding in Containers
Reduce setup times and collaborate using CodeReady Workspaces

Ortwin Schneider, Middleware Solution Architect

Red Hat

WHERE WE ARE TODAY

Development is done on a desktop with containers.

Production is moving to a Kubernetes distribution like OpenShift.

These are not equivalent environments, even though they both use containers.

WHAT'S DIFFERENT?

Development is done on a desktop with containers.

Production is moving to a Kubernetes distribution like OpenShift.

Docker or linux process → Kubernetes infrastructure

Single user load → Concurrent load

No threats → External threats and attacks

Expected user path → Unexpected user paths

THE KEY QUESTION

What’s needed to allow an authorized developer to contribute to a project quickly, easily and safely without relying on containers on the laptop?

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 IDE + LAPTOP APPROACH

Managed in Git

- Project sources
- Dependencies

Managed in the IDE

- Developer Tools: language servers, debuggers, testing tools, security tools, etc...
- Commands

Managed on the Laptop

- Build and packaging tools
- Terminal
- Operating system
- Web server / application server
- Database
- (All other runtime components)

This set is needed for each service that is used. Microservices mean many more of these sets to manage.

TRADITIONAL IDE + LAPTOP APPROACH

Shareable

- Project sources
- Dependencies

Very hard to secure laptops (lost, stolen, hacked) so sources are always at risk.

Shareable

- Developer Tools: language servers, debuggers, testing tools, security tools, etc…
- Commands

Hard to Share

- Build and packaging tools
- Terminal
- Operating system
- Web server / application server
- Database
- (All other runtime components)

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

Easy to Share and Secure

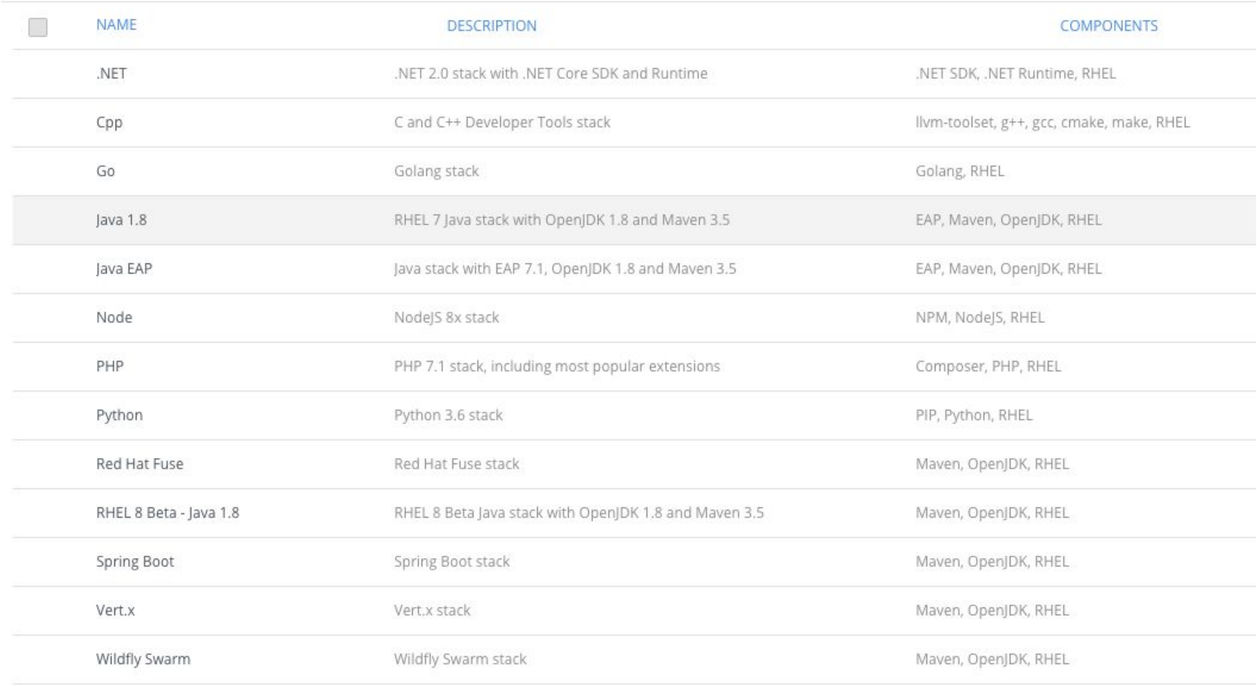
Managed in a containerized Workspace hosted in an IT-Managed OpenShift cluster.

- 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)

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.

CODEREADY WORKSPACES: STACKS



□	NAME	DESCRIPTION	COMPONENTS
	.NET	.NET 2.0 stack with .NET Core SDK and Runtime	.NET SDK, .NET Runtime, RHEL
	Cpp	C and C++ Developer Tools stack	llvm-toolset, g++, gcc, cmake, make, RHEL
	Go	Golang stack	Golang, RHEL
	Java 1.8	RHEL 7 Java stack with OpenJDK 1.8 and Maven 3.5	EAP, Maven, OpenJDK, RHEL
	Java EAP	Java stack with EAP 7.1, OpenJDK 1.8 and Maven 3.5	EAP, Maven, OpenJDK, RHEL
	Node	NodeJS 8x stack	NPM, NodeJS, RHEL
	PHP	PHP 7.1 stack, including most popular extensions	Composer, PHP, RHEL
	Python	Python 3.6 stack	PIP, Python, RHEL
	Red Hat Fuse	Red Hat Fuse stack	Maven, OpenJDK, RHEL
	RHEL 8 Beta - Java 1.8	RHEL 8 Beta Java stack with OpenJDK 1.8 and Maven 3.5	Maven, OpenJDK, RHEL
	Spring Boot	Spring Boot stack	Maven, OpenJDK, RHEL
	Vert.x	Vert.x stack	Maven, OpenJDK, RHEL
	Wildfly Swarm	Wildfly Swarm stack	Maven, OpenJDK, RHEL

Stack: A stack is a configuration template / recipe

CODEREADY WORKSPACES: A WORKSPACE

Workspace: An instance of a stack for your projects

A **Workspace** is made of

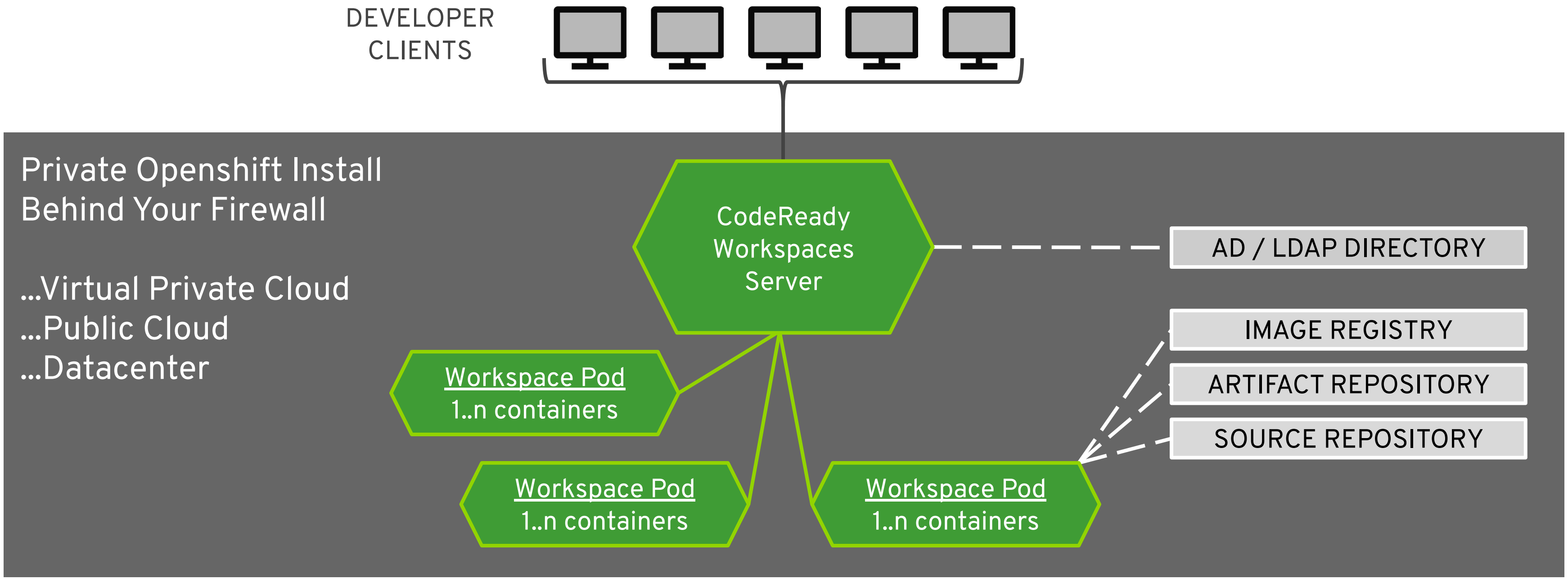
- The **Runtime** from the Stack
- The **Source code** of your Project
- The **IDE** and its configuration

User can **share** the workspace for **collaboration** or **easy development**

EASY & CENTRALIZED CONFIGURATION

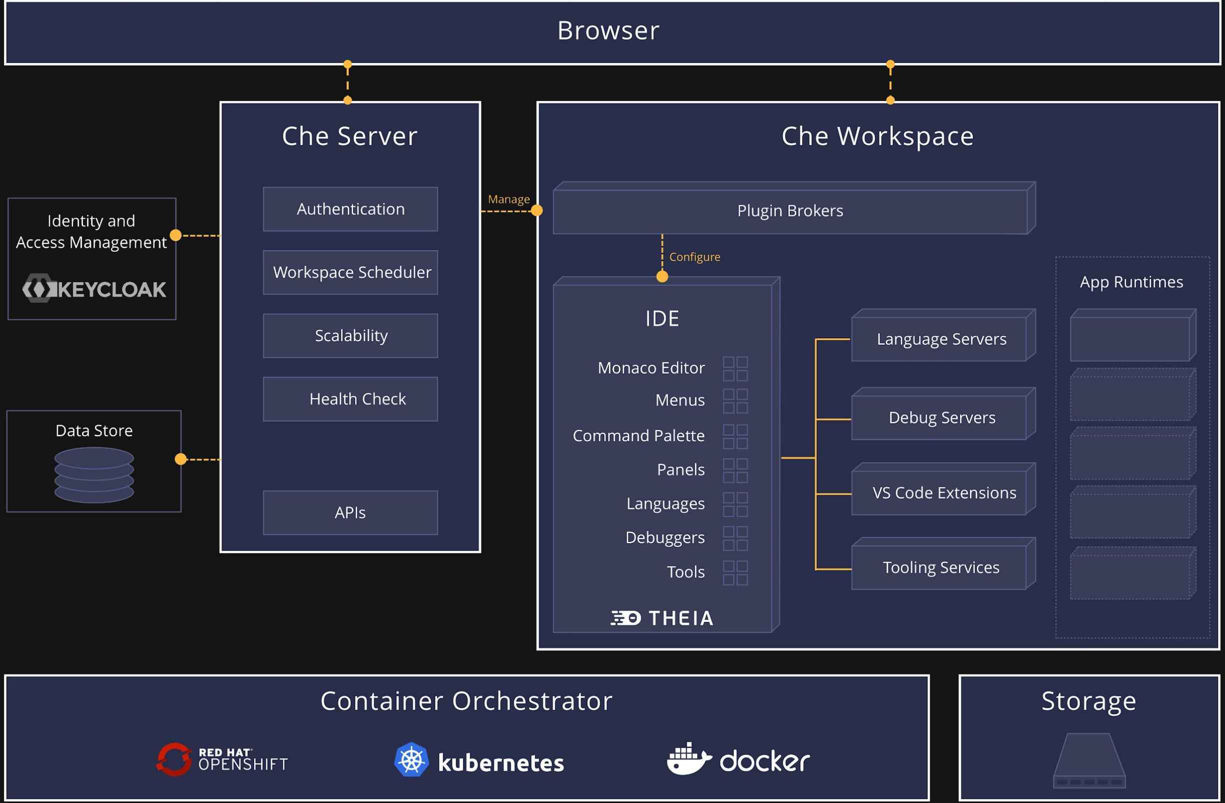
- Centralized Configuration using Stack & Workspaces
  - Controlled development environment and runtimes
  - Runtimes based on RHEL
- Share the Workspaces as a Factory
  - Simple file that your store with your project
- Share the Workspace as a URL for your teams

CODEREADY WORKSPACES ARCHITECTURE



ECLIPSE CHE CONCEPTUAL ARCHITECTURE

- Users can have more than one workspace (admin setting)
- One Che server per Kubernetes cluster
- Che server manages the lifecycle and authentication of all workspaces in the cluster
- Traffic to browser is minimal
- Majority of calls are inside the Che workspace, giving a "localhost" experience



DIFFERENT NEEDS, DIFFERENT GOALS

Developers / QA / Docs

- Bulk of the team
- Work off issues in a backlog
- Need guidance from leads/experts

Success defined by steady progress: effective and efficient.

“Help me burn down the backlog faster.”

Experts / Leads / Architects

- Small number in a team
- Likely “float” or run complex tasks
- Provide guidance and coaching

Success defined by leaps in progress: innovation and outperformance.

“Help the team run faster, with fewer stumbles.”

DIFFERENT NEEDS, DIFFERENT GOALS

Developers / QA / Docs

Uses CodeReady Workspaces

"Help me burn down the backlog faster."

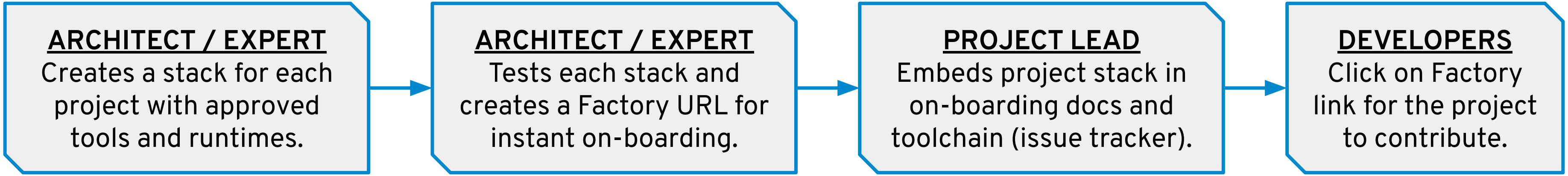
Experts / Architects

Builds CodeReady Workspace Stacks and Factories

"Help the team run faster, with fewer stumbles."

WORKFLOW AND HANDOFFS

New Project Start



ARCHITECT / EXPERT
Creates a stack for each project with approved tools and runtimes.

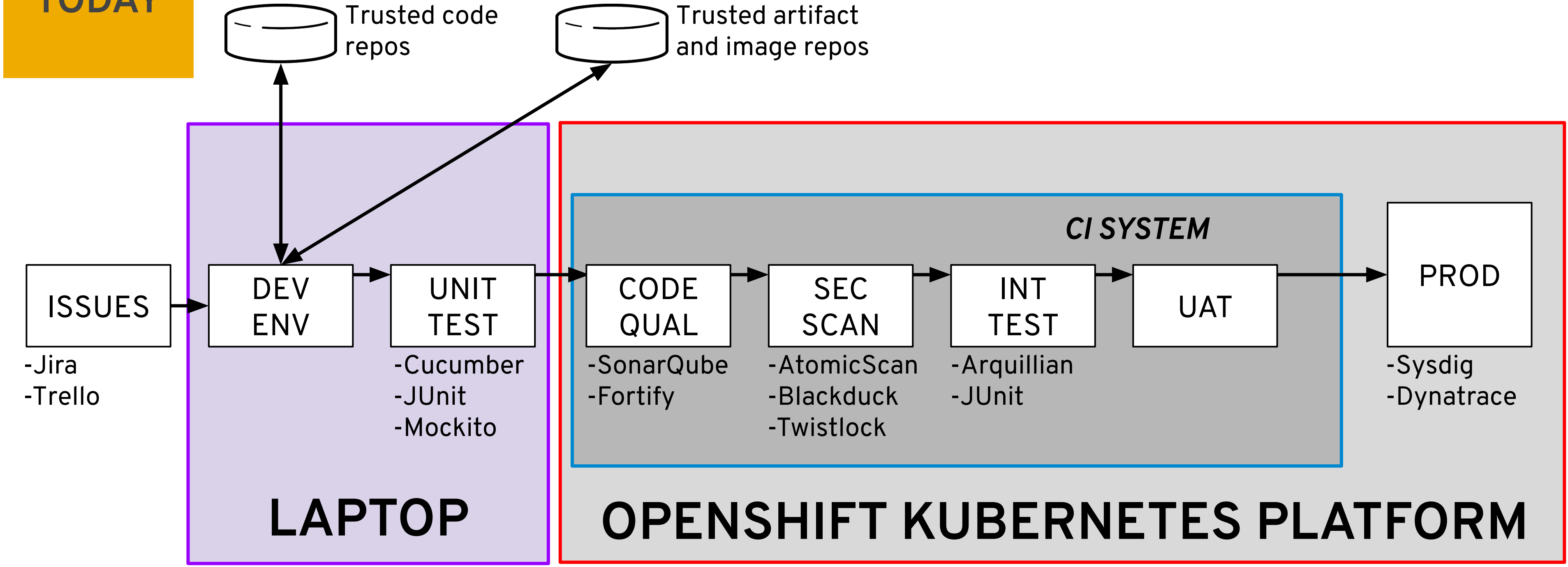
ARCHITECT / EXPERT
Tests each stack and creates a Factory URL for instant on-boarding.

PROJECT LEAD
Embeds project stack in on-boarding docs and toolchain (issue tracker).

DEVELOPERS
Click on Factory link for the project to contribute.

TODAY

POWERED BY WORKSPACES ON OPENSHIFT



Trusted code repos

Trusted artifact and image repos

ISSUES
-Jira
-Trello

DEV ENV

UNIT TEST
-Cucumber
-JUnit
-Mockito

LAPTOP

CI SYSTEM

CODE QUAL
-SonarQube
-Fortify

SEC SCAN
-AtomicScan
-Blackduck
-Twistlock

INT TEST
-Arquillian
-JUnit

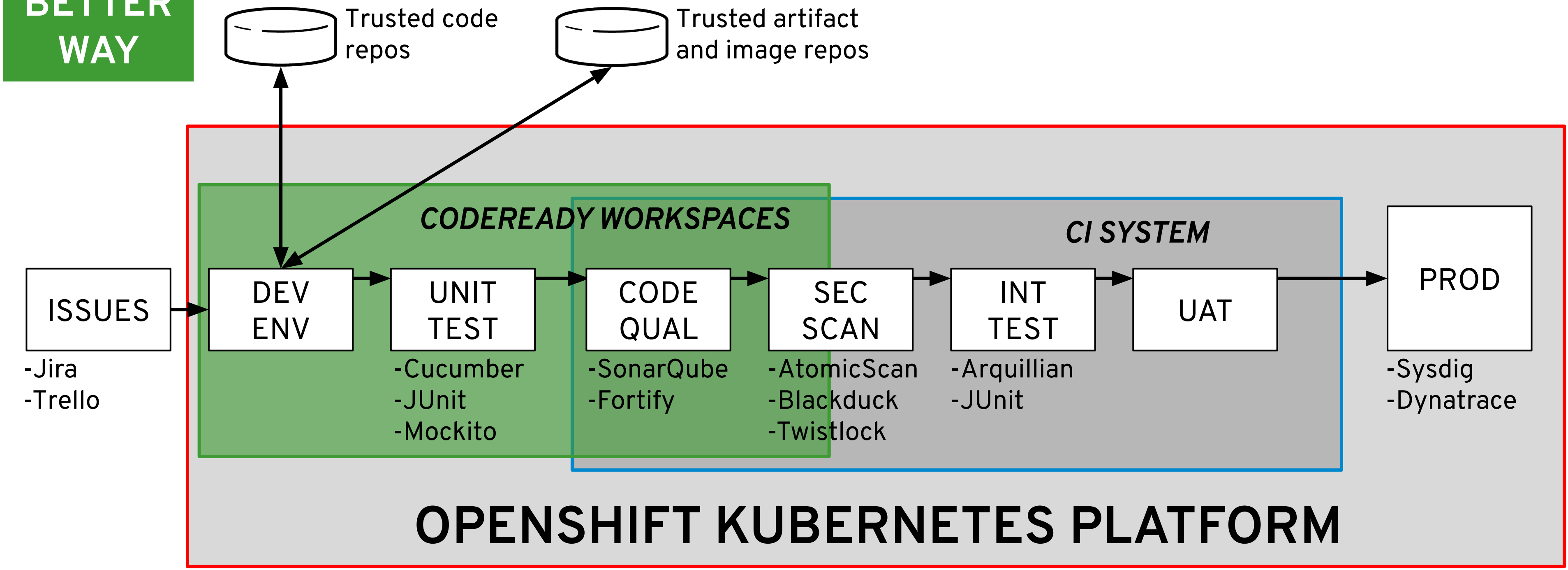
UAT

PROD
-Sysdig
-Dynatrace

OPENSHIFT KUBERNETES PLATFORM

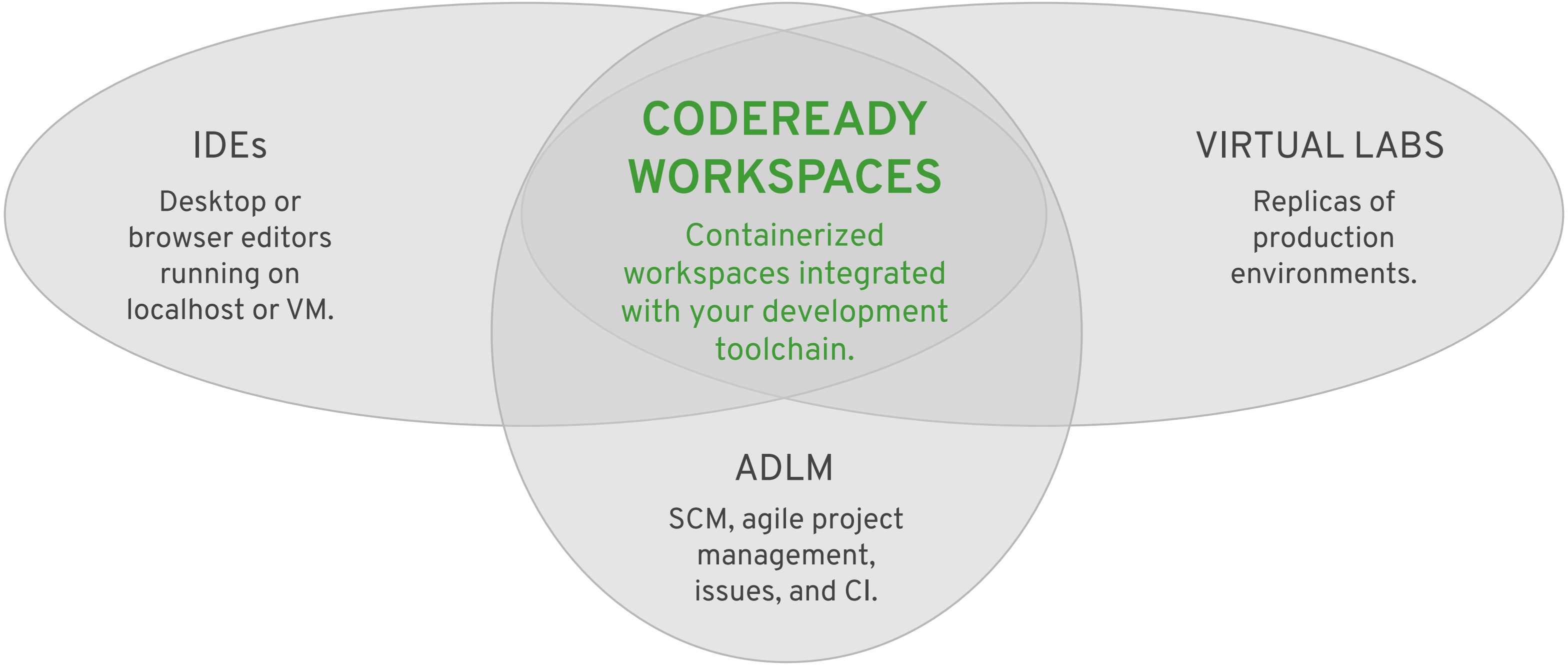
THE BETTER WAY

POWERED BY WORKSPACES ON OPENSHIFT



CODEREADY WORKSPACES IS UNIQUE

Combines aspects of an IDE, a virtual lab environment and lifecycle management.
All hosted on a private OpenShift Kubernetes cluster.



IDEs

Desktop or browser editors running on localhost or VM.

CODEREADY WORKSPACES

Containerized workspaces integrated with your development toolchain.

VIRTUAL LABS

Replicas of production environments.

ADLM

SCM, agile project management, issues, and CI.

CODEREADY WORKSPACES AND ECLIPSE CHE

- CodeReady Workspaces is a supported subscription for the open Eclipse Che project.
- It is **not** a fork.
- CodeReady Workspaces and Eclipse Che have nearly identical functionality, the only differences are:
  - Workspaces is supported by Red Hat's global support organization
  - Workspaces includes only Red Hat Enterprise Linux based secure stacks
  - Workspaces is released quarterly (Che is released every 3 weeks)
  - Workspaces releases are tested more thoroughly due to the slower cadence
  - Any bugs fixed are contributed to the upstream Eclipse Che sources for inclusion into a future release (no proprietary code)
- The CodeReady Workspaces development team includes several Che committers, making it easier and faster for us to contribute bug fixes back to the community

CODEREADY WORKSPACES DEMO

HOW TO GET CODEREADY WORKSPACES?

What is the price for Red Hat CodeReady Workspaces?

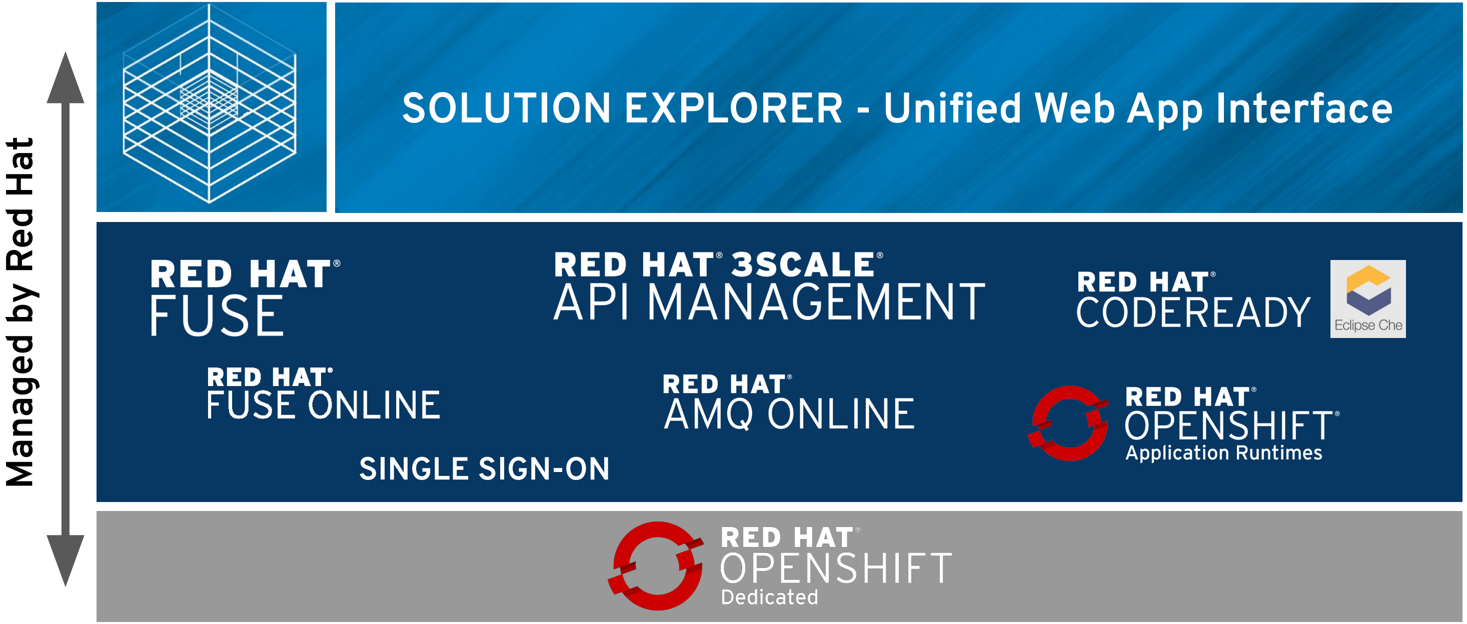
- Red Hat CodeReady Workspaces is free to any customers with a subscription for OpenShift Container Platform, OpenShift Dedicated or OpenShift Online. It is not available with OpenShift Engine.
  - Red Hat CodeReady Workspaces is compatible with OpenShift Container Platform (OCP) and OpenShift Dedicated (OSD) 3.11+.

Can I try Che or Red Hat CodeReady Workspaces today?

YES

https://access.redhat.com/documentation/en-us/red_hat_codeready_workspaces_for_openshift/

NEW: RedHat Managed Integration



Managed by Red Hat

SOLUTION EXPLORER - Unified Web App Interface

RED HAT® FUSE

RED HAT® 3SCALE® API MANAGEMENT

RED HAT® CODEREADY

Eclipse Che

RED HAT® FUSE ONLINE

RED HAT® AMQ ONLINE

RED HAT® OPENSHIFT® Application Runtimes

SINGLE SIGN-ON

RED HAT® OPENSHIFT Dedicated

Front End App

Hosted, pre-configured & pre-installed products (Integration bundle)

Single tenant

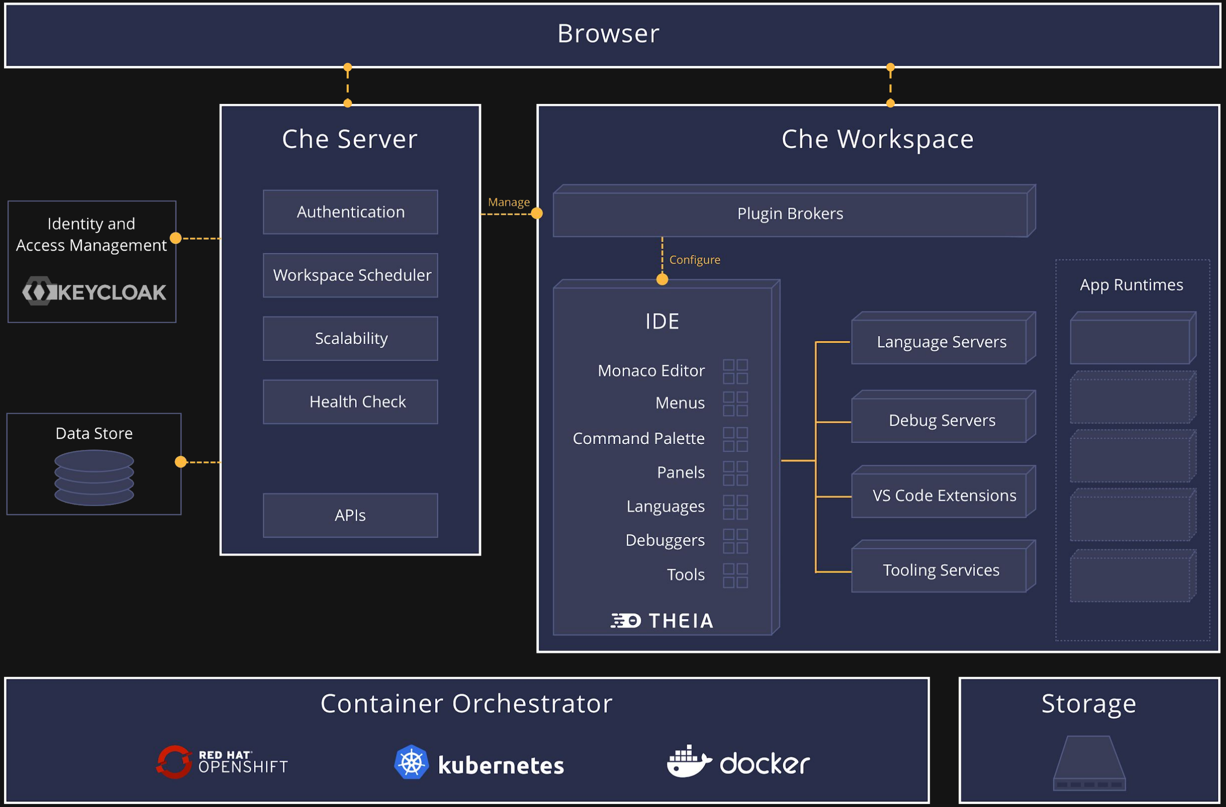
BACKUP SLIDES

+

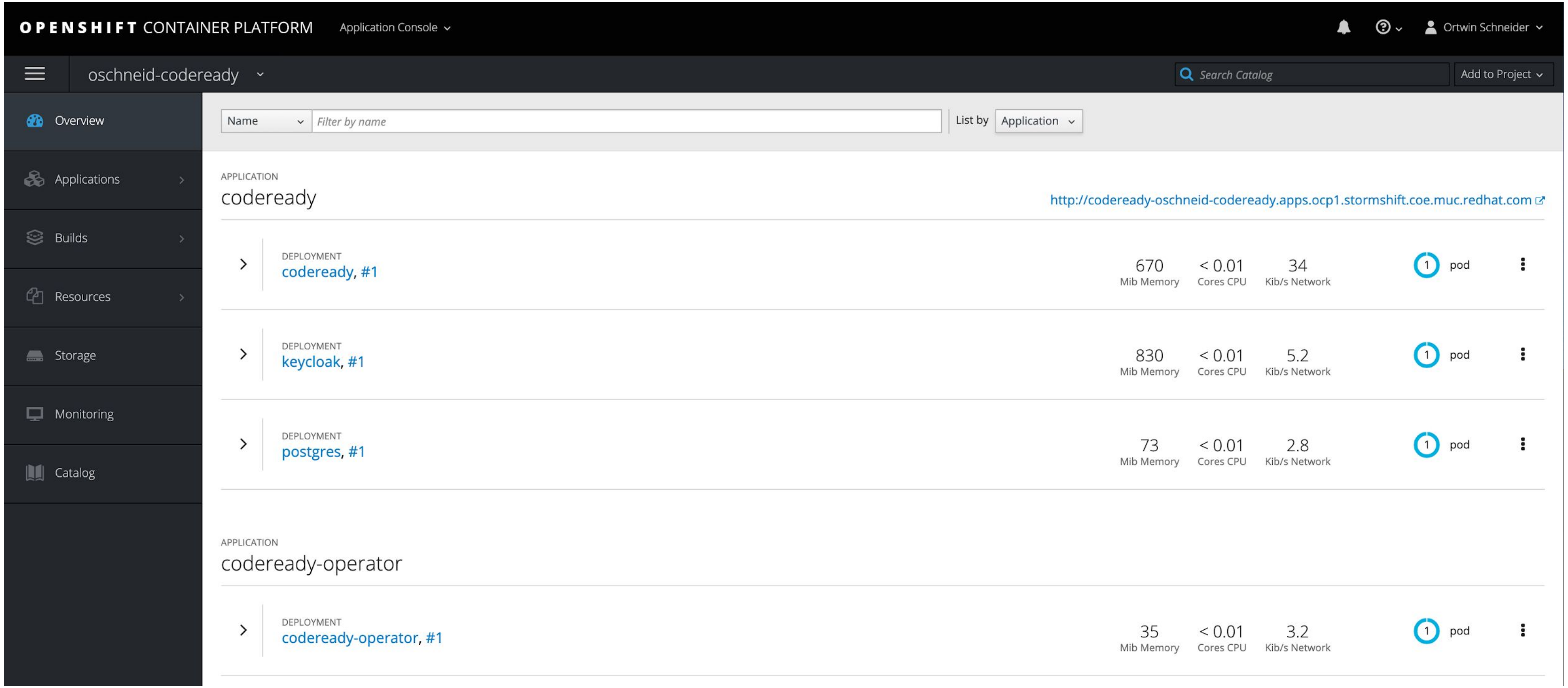
SCREENSHOTS

ECLIPSE CHE CONCEPTUAL ARCHITECTURE

- Users can have more than one workspace (admin setting)
- One Che server per Kubernetes cluster
- Che server manages the lifecycle and authentication of all workspaces in the cluster
- Traffic to browser is minimal
- Majority of calls are inside the Che workspace, giving a “localhost” experience



CodeReady Workspaces OpenShift Resources



APPLICATION

codeready

http://codeready-oschneid-codeready.apps.ocp1.stormshift.coe.muc.redhat.com

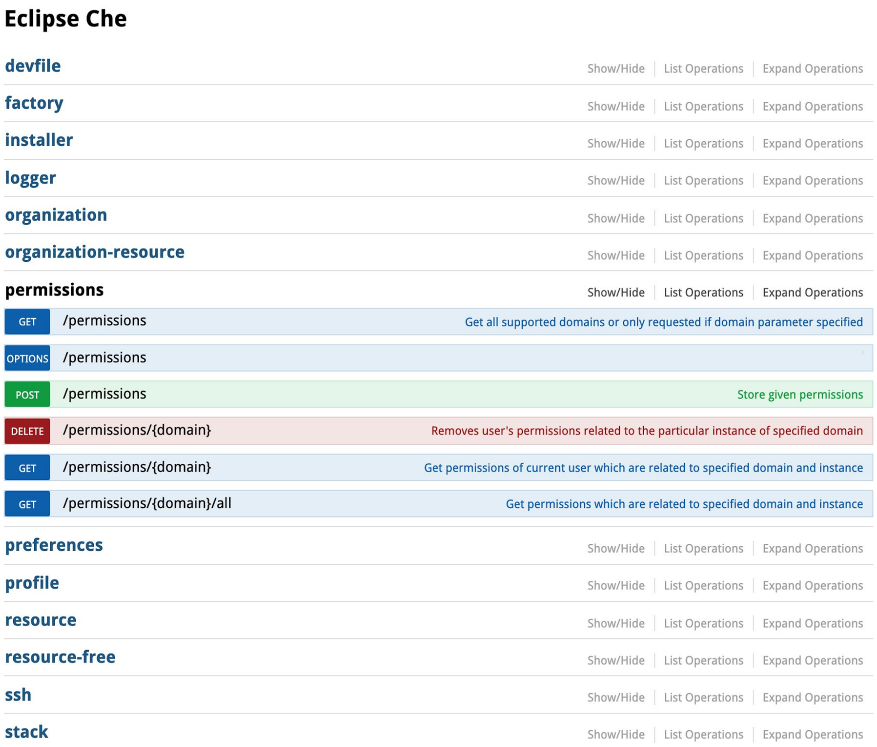
DEPLOYMENT	Mib Memory	Cores CPU	Kib/s Network	
codeready, #1	670	< 0.01	34	1 pod
keycloak, #1	830	< 0.01	5.2	1 pod
postgres, #1	73	< 0.01	2.8	1 pod

APPLICATION

codeready-operator

DEPLOYMENT	Mib Memory	Cores CPU	Kib/s Network	
codeready-operator, #1	35	< 0.01	3.2	1 pod

CodeReady Workspaces REST API (Swagger)



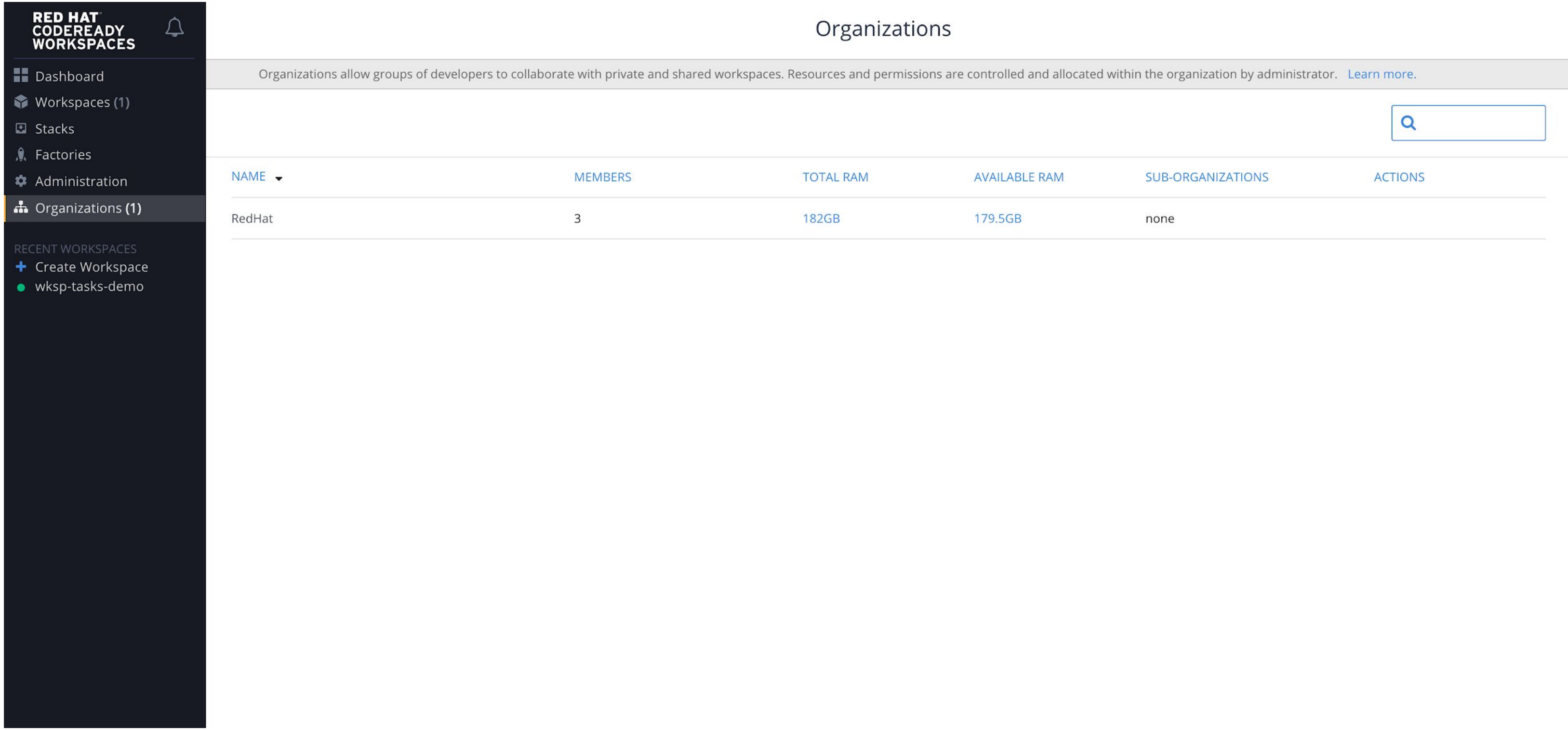
Eclipse Che

- devfile — Show/Hide | List Operations | Expand Operations
- factory — Show/Hide | List Operations | Expand Operations
- installer — Show/Hide | List Operations | Expand Operations
- logger — Show/Hide | List Operations | Expand Operations
- organization — Show/Hide | List Operations | Expand Operations
- organization-resource — Show/Hide | List Operations | Expand Operations
- permissions — Show/Hide | List Operations | Expand Operations

GET	/permissions	Get all supported domains or only requested if domain parameter specified
OPTIONS	/permissions	
POST	/permissions	Store given permissions
DELETE	/permissions/{domain}	Removes user's permissions related to the particular instance of specified domain
GET	/permissions/{domain}	Get permissions of current user which are related to specified domain and instance
GET	/permissions/{domain}/all	Get permissions which are related to specified domain and instance

- preferences — Show/Hide | List Operations | Expand Operations
- profile — Show/Hide | List Operations | Expand Operations
- resource — Show/Hide | List Operations | Expand Operations
- resource-free — Show/Hide | List Operations | Expand Operations
- ssh — Show/Hide | List Operations | Expand Operations
- stack — Show/Hide | List Operations | Expand Operations

Organizations

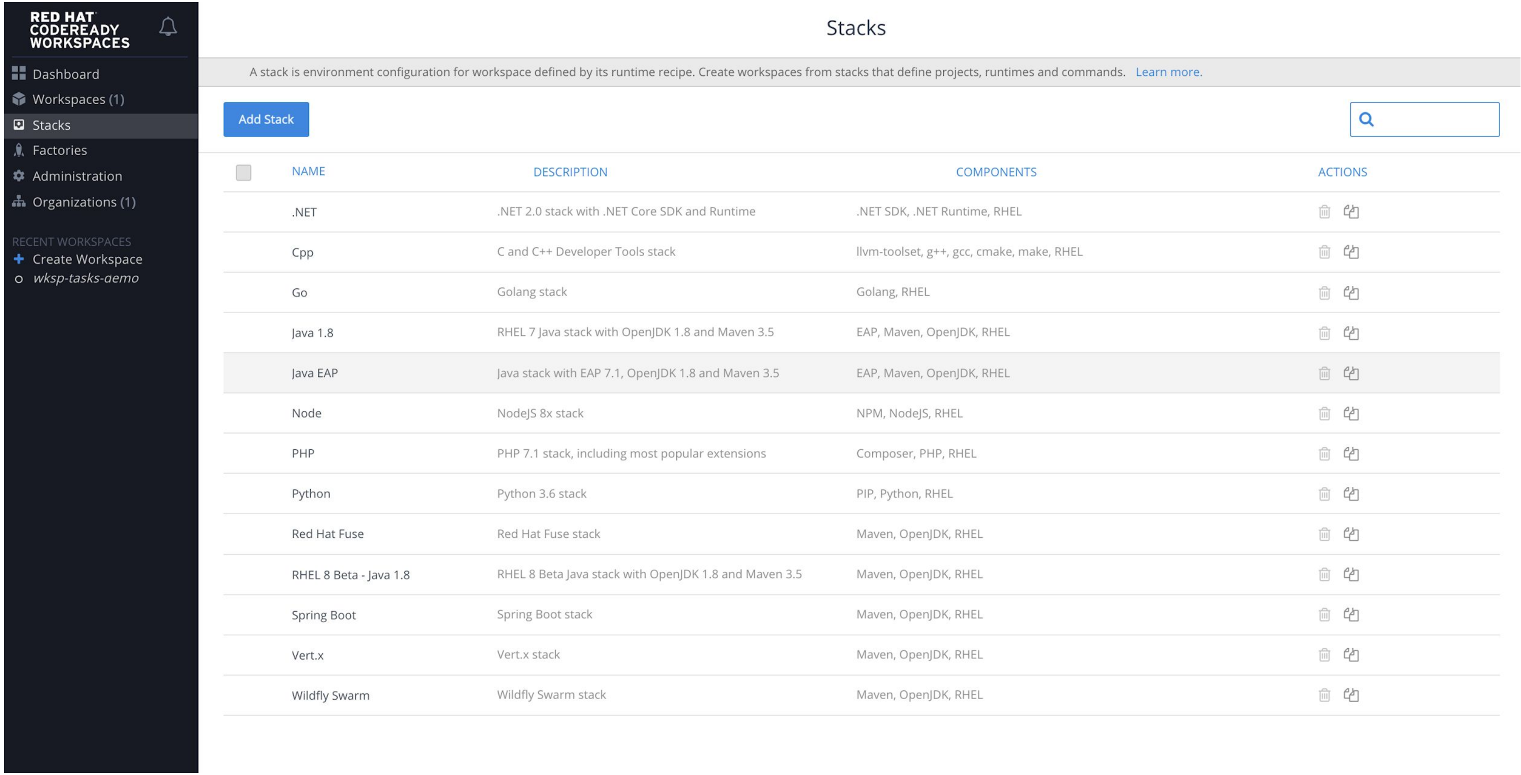


Organizations

Organizations allow groups of developers to collaborate with private and shared workspaces. Resources and permissions are controlled and allocated within the organization by administrator. Learn more.

NAME	MEMBERS	TOTAL RAM	AVAILABLE RAM	SUB-ORGANIZATIONS	ACTIONS
RedHat	3	182GB	179.5GB	none	

Stacks

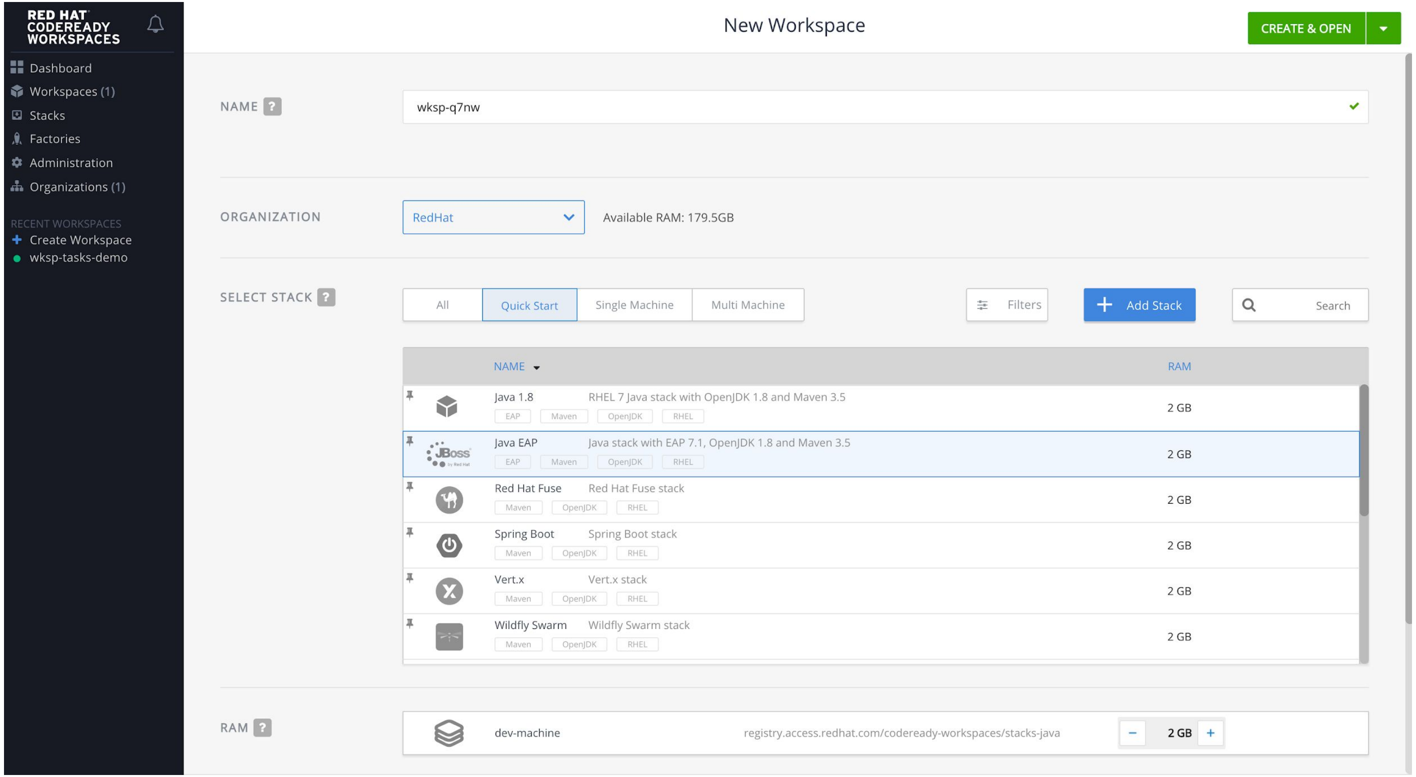


Stacks

A stack is environment configuration for workspace defined by its runtime recipe. Create workspaces from stacks that define projects, runtimes and commands. Learn more.

NAME	DESCRIPTION	COMPONENTS	ACTIONS
.NET	.NET 2.0 stack with .NET Core SDK and Runtime	.NET SDK, .NET Runtime, RHEL	
Cpp	C and C++ Developer Tools stack	llvm-toolset, g++, gcc, cmake, make, RHEL	
Go	Golang stack	Golang, RHEL	
Java 1.8	RHEL 7 Java stack with OpenJDK 1.8 and Maven 3.5	EAP, Maven, OpenJDK, RHEL	
Java EAP	Java stack with EAP 7.1, OpenJDK 1.8 and Maven 3.5	EAP, Maven, OpenJDK, RHEL	
Node	NodeJS 8x stack	NPM, NodeJS, RHEL	
PHP	PHP 7.1 stack, including most popular extensions	Composer, PHP, RHEL	
Python	Python 3.6 stack	PIP, Python, RHEL	
Red Hat Fuse	Red Hat Fuse stack	Maven, OpenJDK, RHEL	
RHEL 8 Beta - Java 1.8	RHEL 8 Beta Java stack with OpenJDK 1.8 and Maven 3.5	Maven, OpenJDK, RHEL	
Spring Boot	Spring Boot stack	Maven, OpenJDK, RHEL	
Vert.x	Vert.x stack	Maven, OpenJDK, RHEL	
Wildfly Swarm	Wildfly Swarm stack	Maven, OpenJDK, RHEL	

New Workspace



NAME: wksp-q7nw

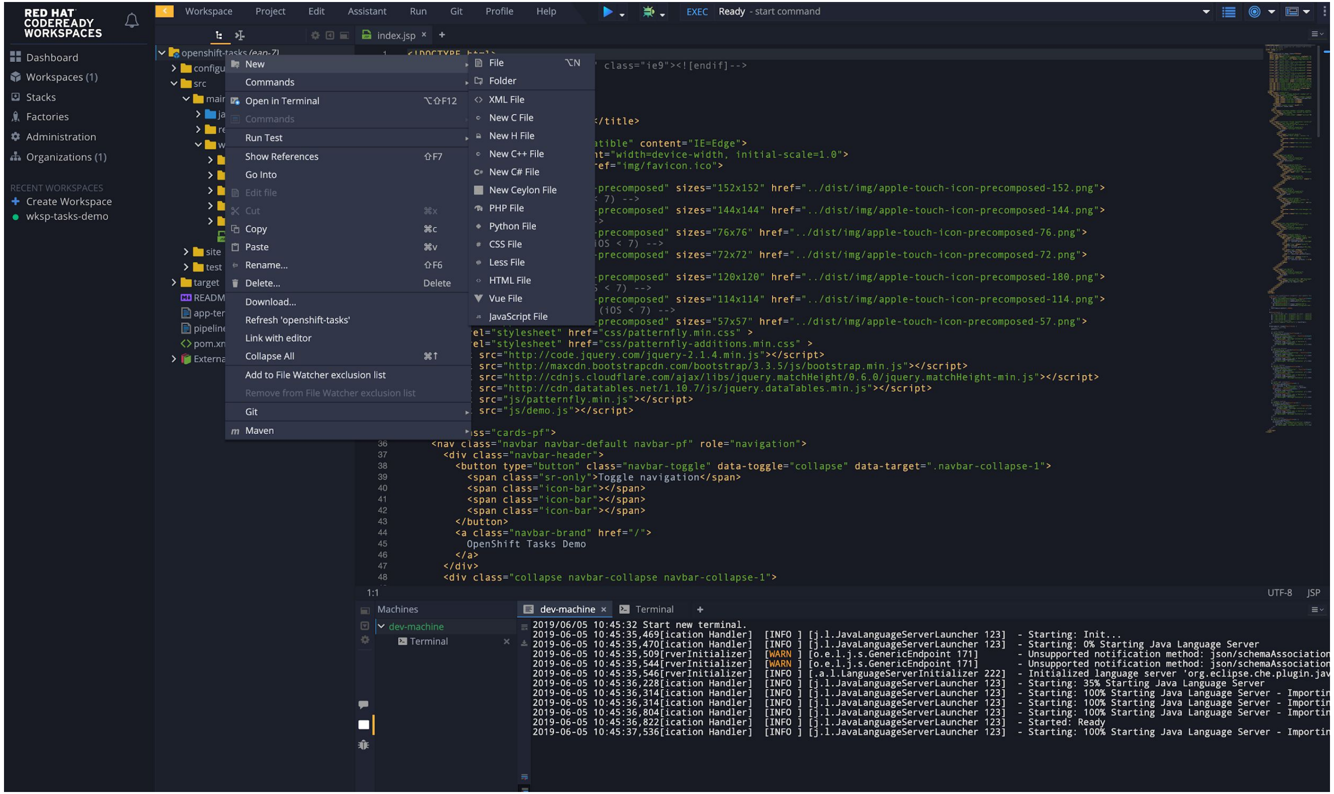
ORGANIZATION: RedHat   Available RAM: 179.5GB

SELECT STACK: All | Quick Start | Single Machine | Multi Machine   Filters   Add Stack   Search

NAME		RAM
Java 1.8 EAP Maven OpenJDK RHEL	RHEL 7 Java stack with OpenJDK 1.8 and Maven 3.5	2 GB
Java EAP EAP Maven OpenJDK RHEL	Java stack with EAP 7.1, OpenJDK 1.8 and Maven 3.5	2 GB
Red Hat Fuse Maven OpenJDK RHEL	Red Hat Fuse stack	2 GB
Spring Boot Maven OpenJDK RHEL	Spring Boot stack	2 GB
Vert.x Maven OpenJDK RHEL	Vert.x stack	2 GB
Wildfly Swarm Maven OpenJDK RHEL	Wildfly Swarm stack	2 GB

RAM: dev-machine   registry.access.redhat.com/codeready-workspaces/stacks-java   − 2 GB +

Web IDE



```
<!DOCTYPE html>
' class="ie9"><![endif]-->
</title>
atible" content="IE=Edge">
nt="width=device-width, initial-scale=1.0">
ref="img/favicon.ico">
-precomposed" sizes="152x152" href="../dist/img/apple-touch-icon-precomposed-152.png">
< 7) -->
-precomposed" sizes="144x144" href="../dist/img/apple-touch-icon-precomposed-144.png">
->
-precomposed" sizes="76x76" href="../dist/img/apple-touch-icon-precomposed-76.png">
iOS < 7) -->
-precomposed" sizes="72x72" href="../dist/img/apple-touch-icon-precomposed-72.png">
-precomposed" sizes="120x120" href="../dist/img/apple-touch-icon-precomposed-180.png">
5 < 7) -->
-precomposed" sizes="114x114" href="../dist/img/apple-touch-icon-precomposed-114.png">
(iOS < 7) -->
-precomposed" sizes="57x57" href="../dist/img/apple-touch-icon-precomposed-57.png">
el="stylesheet" href="css/patternfly.min.css" >
el="stylesheet" href="css/patternfly-additions.min.css" >
 src="http://code.jquery.com/jquery-2.1.4.min.js"></script>
 src="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.5/js/bootstrap.min.js"></script>
 src="http://cdnjs.cloudflare.com/ajax/libs/jquery.matchHeight/0.6.0/jquery.matchHeight-min.js"></script>
 src="http://cdn.datatables.net/1.10.7/js/jquery.dataTables.min.js"></script>
 src="js/patternfly.min.js"></script>
 src="js/demo.js"></script>

ss="cards-pf">
    <nav class="navbar navbar-default navbar-pf" role="navigation">
      <div class="navbar-header">
        <button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse-1">
          <span class="sr-only">Toggle navigation</span>
          <span class="icon-bar"></span>
          <span class="icon-bar"></span>
          <span class="icon-bar"></span>
        </button>
        <a class="navbar-brand" href="/">
          OpenShift Tasks Demo
        </a>
      </div>
      <div class="collapse navbar-collapse navbar-collapse-1">
```

```
2019/06/05 10:45:32 Start new terminal.
2019-06-05 10:45:35,469[ication Handler]  [INFO ] [j.l.JavaLanguageServerLauncher 123]  - Starting: Init...
2019-06-05 10:45:35,470[ication Handler]  [INFO ] [j.l.JavaLanguageServerLauncher 123]  - Starting: 0% Starting Java Language Server
2019-06-05 10:45:35,509[rverInitializer]  [WARN ] [o.e.l.j.s.GenericEndpoint 171]       - Unsupported notification method: json/schemaAssociation
2019-06-05 10:45:35,544[rverInitializer]  [WARN ] [o.e.l.j.s.GenericEndpoint 171]       - Unsupported notification method: json/schemaAssociation
2019-06-05 10:45:35,546[rverInitializer]  [INFO ] [.a.l.LanguageServerInitializer 222]  - Initialized language server 'org.eclipse.che.plugin.jav
2019-06-05 10:45:36,228[ication Handler]  [INFO ] [j.l.JavaLanguageServerLauncher 123]  - Starting: 35% Starting Java Language Server
2019-06-05 10:45:36,314[ication Handler]  [INFO ] [j.l.JavaLanguageServerLauncher 123]  - Starting: 100% Starting Java Language Server - Importin
2019-06-05 10:45:36,314[ication Handler]  [INFO ] [j.l.JavaLanguageServerLauncher 123]  - Starting: 100% Starting Java Language Server - Importin
2019-06-05 10:45:36,804[ication Handler]  [INFO ] [j.l.JavaLanguageServerLauncher 123]  - Starting: 100% Starting Java Language Server - Importin
2019-06-05 10:45:36,822[ication Handler]  [INFO ] [j.l.JavaLanguageServerLauncher 123]  - Started: Ready
2019-06-05 10:45:37,536[ication Handler]  [INFO ] [j.l.JavaLanguageServerLauncher 123]  - Starting: 100% Starting Java Language Server - Importin
```