

# PaaS Einführung: „Re-Platform“ oder „Re-Factor“

Wie können existierende JEE Applikationen „Containerisiert“ werden?

Carsten Lichy-Bittendorf

Cloud Success Architect

27. November 2018

# AGENDA

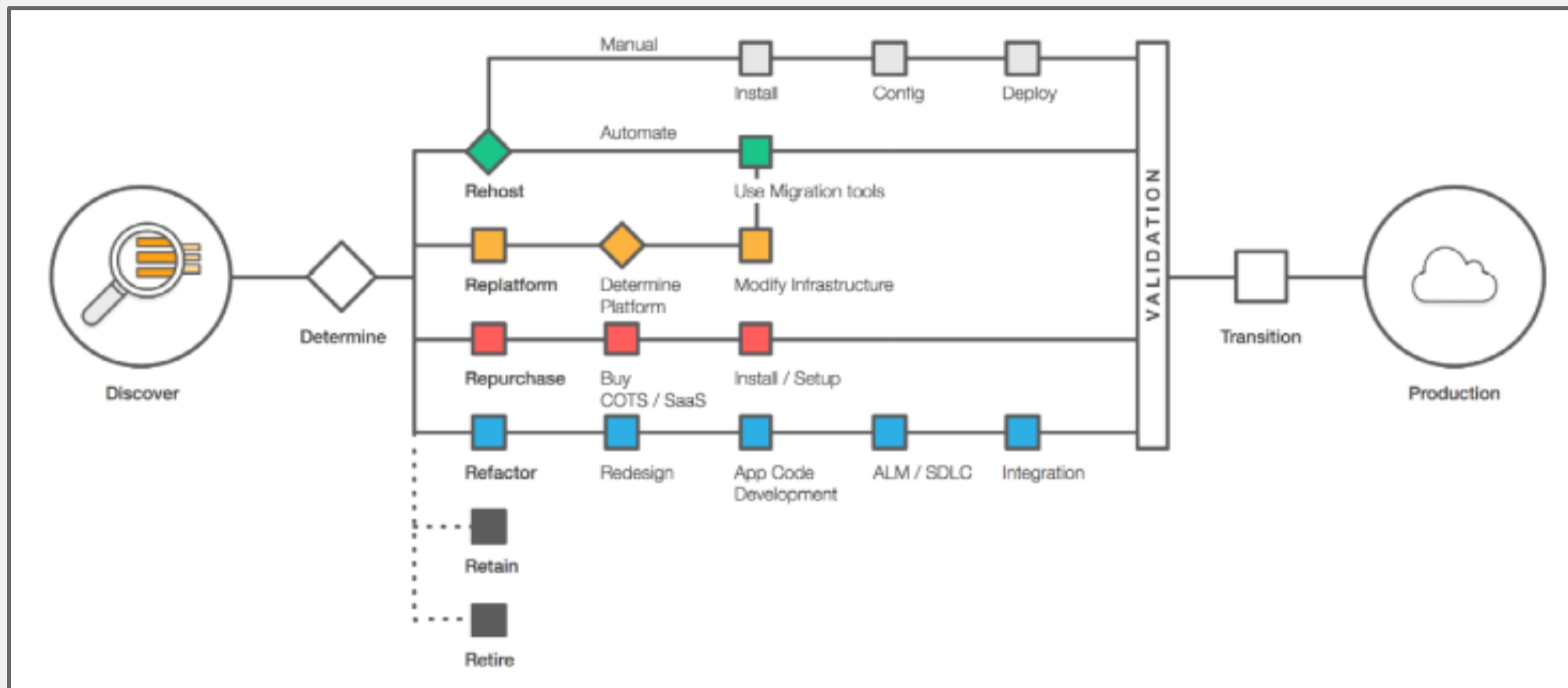
Time to practice ...

- Why this workshop?
- Switch to the console - sorry, slides only this time
- Takeaways



# OPTIONS FOR APPLICATION MODERNIZATION

Common strategies: “The 6 R’s” from Amazon & Gartner



# What is available today

to see what can get reused

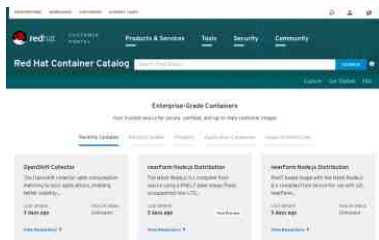
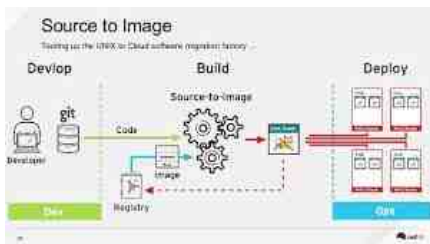


What is a common setup today:

- Java / JEE application
- Maven build
- revisioned in an enterprise repo
- deployed on JBoss EAP || TomCat

# Main Pillars of the Solution

It's all about orchestration



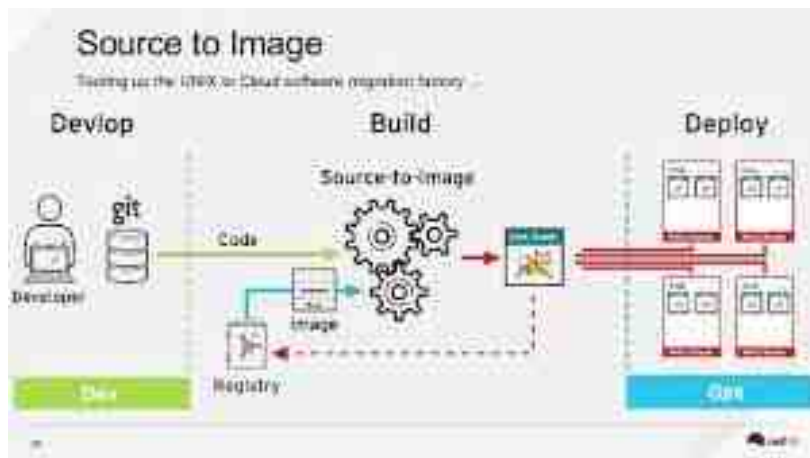
Red Hat's product for PaaS is OpenShift Container Platform. It is much more than Docker + Kubernetes as it is complete a Ecosystem.

What is needed to containerize an existing JEE application:

- JBoss EAP base image
  - preferred form Red Hat Container Catalog
- Source-to-Image [S2I] build

# Source-to-Image [S2I] build

The application build orchestration for OpenShift



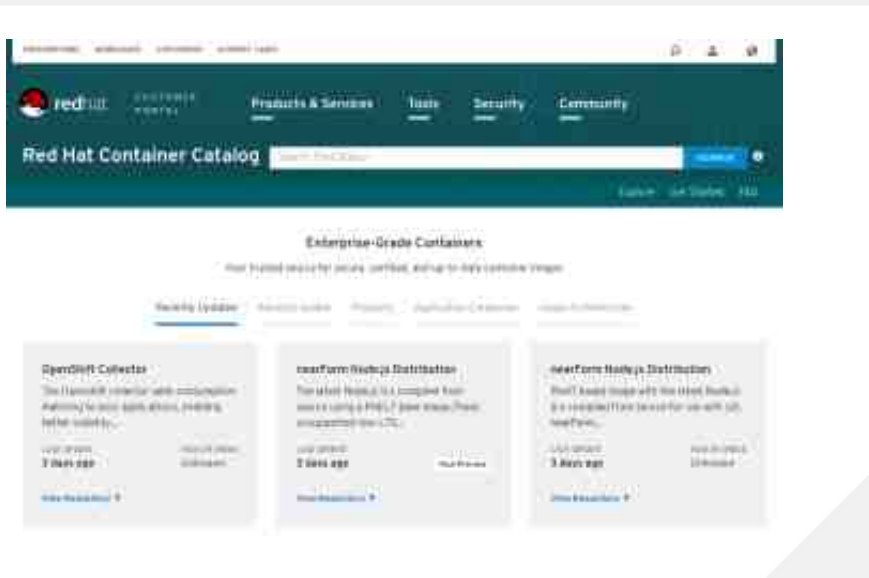
Source to Image is a build factory as part of OpenShift Container Platform. It capable to build ready to deploy container images from application sources using a base image.

Source to Image is available for:

- Java - Maven based builds
- Node.js
- Perl
- PHP
- Python
- Ruby
- .NET Core

# Red Hat Container Catalog

Base your application on images from a reliable source



There are plenty of images on public repos in the internet. Please consider using the one Red Hat provides in the Red Hat Container Catalog.

Simply as Red Hat has a defined life-cycle on all images we ship. So you can be sure that we'll maintain them and apply patches, especially security patches. We even offer you a health index on every image release.

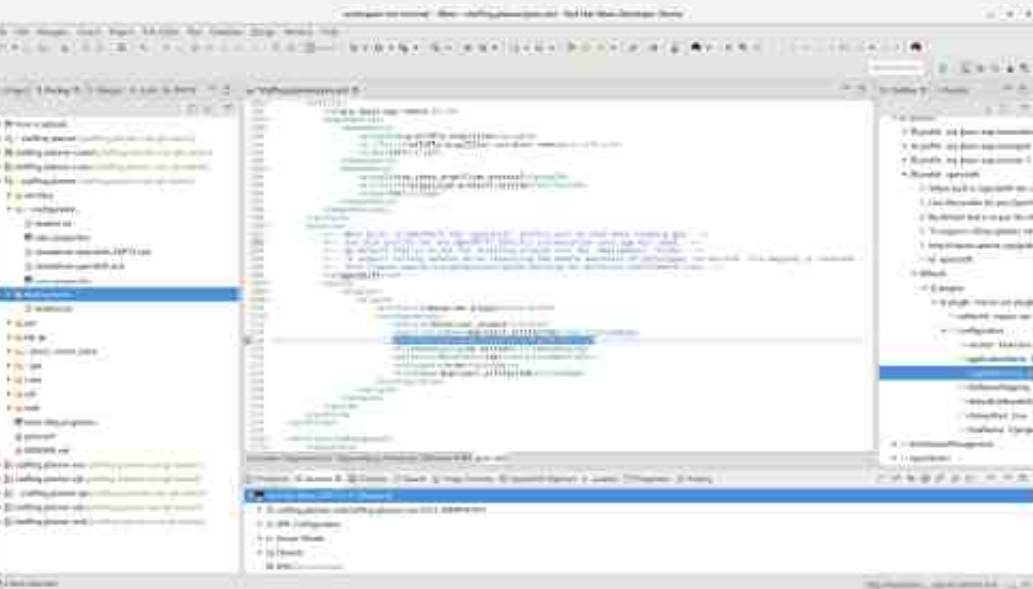
Furthermore, the xPaaS images are prepared for S2I builds.



Let's get it all assembled



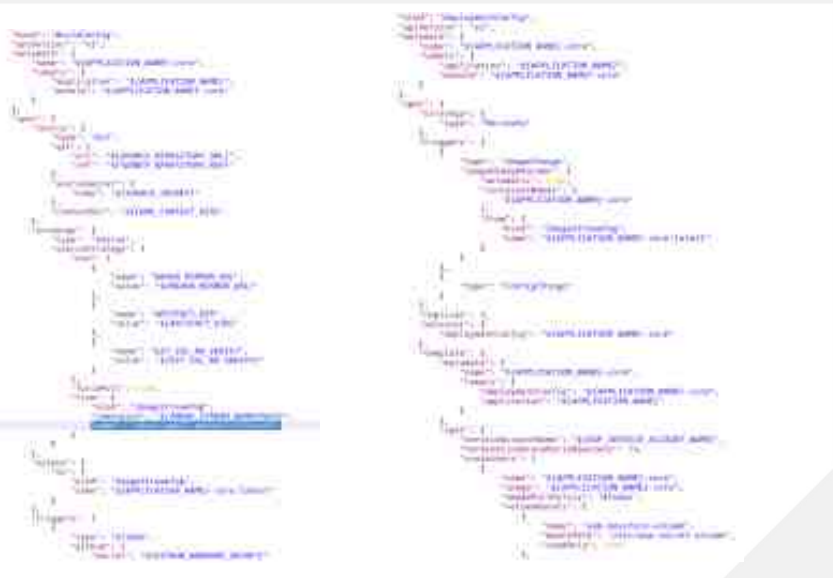
# Prepare the Application Build for S2I



Steps to prepare application build:

- Add directory “deployments” as this content get deployed into EAP image
- Add Maven-profile “openshift” as this profile is the default for a S2I-Maven build.
  - Configure <outputDirectory> to directory “deployments”

# Chose or Create your own Template



- OpenShift comes with a collection of templates
- These can not cover all combinations and are good for
  - Demonstration and
  - Quick testing
- It is common practice to build your own template describing your system in all details
- Different stages might lead to variants of the template

# Upload your Template



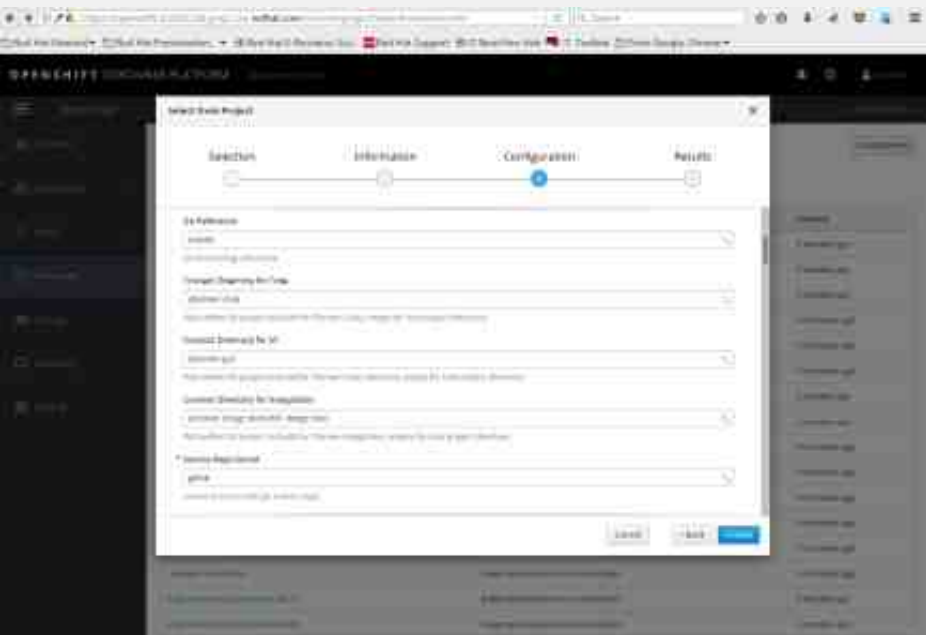
Your template can be uploaded using

- the Web-UI or
- the command-line

It can get uploaded into

- a single project,
- made available across multiple projects or
- OpenShift cluster wide

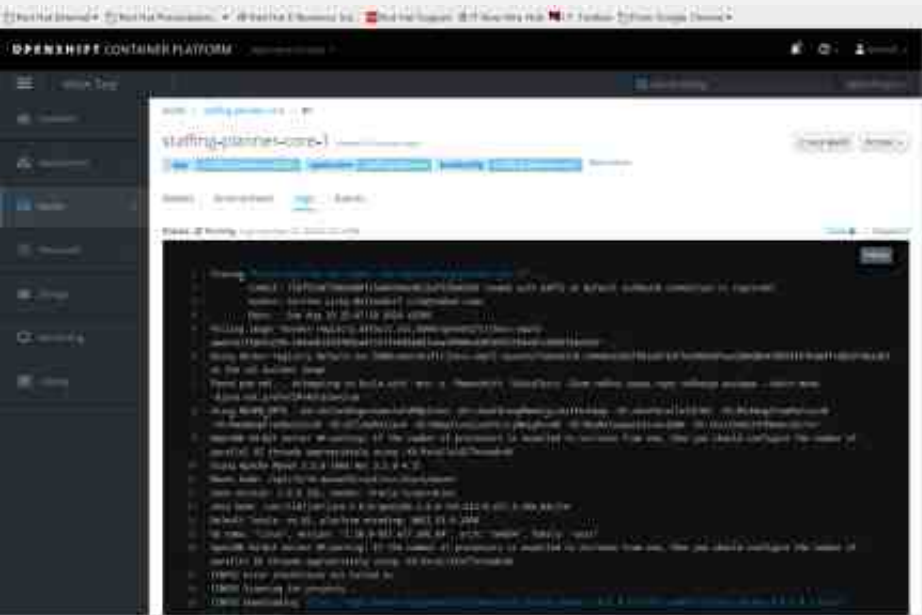
# Instantiate your Template



Instantiate the template to

- To launch included build-config to build the images
- To deploy images as specified in the DCs
- To create all required services and routes.

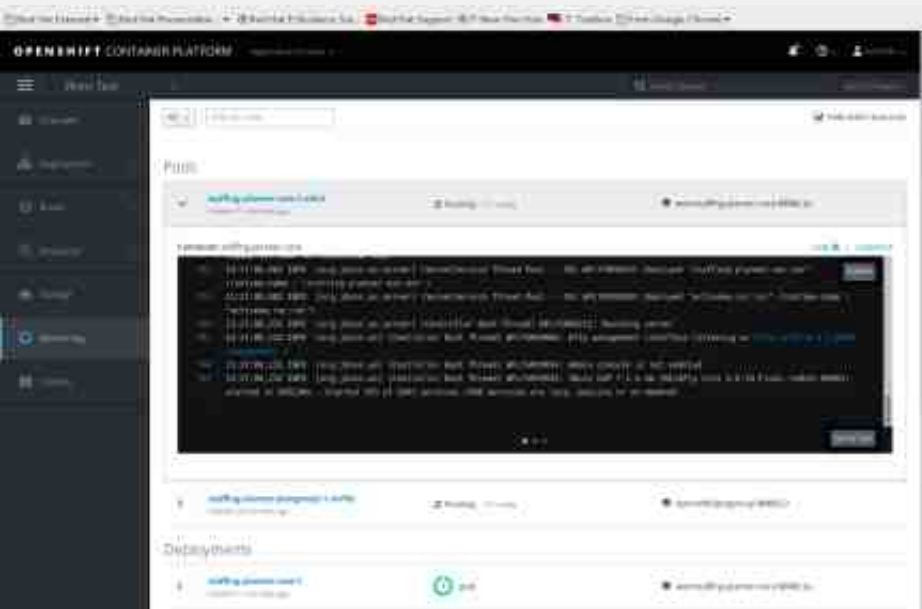
# See how it builds



Following the logs of the build pod you'll see:

1. The sources get pulled
2. The Maven Build launched to build the application
3. New Image-Layers get build
4. New Image-Layers get pushed to the registry

# Check the results



When the build was successful there will be:

- The resulting image in the registry
- An Image-Stream referencing the image

If a DC with an Image-Change trigger has been created the new image will also get deployed instantly

# Takeaways

What has been demonstrated

Red Hat provides a tooling around Java/JEE applications to containerize them

- Source-to-Image Build can get used to build Maven-build application into containers
- Red Hat Container Catalog contains all what Red Hat ships as JBoss middleware
  - Prefer to use those as Red Hat cares on patching these images

Using these tools

- existing Java/JEE applications can containerized without large efforts.
- mixed development modes (classic and containerized) are feasible.

Surly the application itself needs to respect the container limitations.

# Further Readings

For all who like to know more

If you want to learn more, evaluate, etc.

- [Red Hat Application Migration Toolkit](#) to analyse existing apps
- Documentation on [RH JBoss Middleware for OpenShift](#)
- Using [Jenkins Build-Pipelines with OpenShift](#)

The technology to address the staffing-planning challenge:

- [OptaPlanner](#) as a constraint solver, which is part of [Red Hat Decision Manager](#)



## Carsten Lichy-Bittendorf

Cloud Success Architect

Phone: +49 69 365 051 024

Mail: clichybi@redhat.com

# THANK YOU



[plus.google.com/+RedHat](https://plus.google.com/+RedHat)



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



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



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



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