

OPENSIFT – The Art Of Development

Or “Why Development must be a labour of love”

Ian Lawson
Specialist Solution Architect
October 2017



OpenShift is the Developer's Dream

WRITING SOFTWARE IS AN ART FORM

- Developers have passion
- Developers are difficult beasts
- Developers are demanding (because their bosses are demanding!)
- Developers are destructive (because they are constructive!)

BUT WE PAY THEM TO PRODUCE!

- Developers often have difficult timescales
- Developers often don't have solid requirements
- Trendy technologies interfere with consistent software production
- Ops are evil! (not really)
- Devs are evil (again, not really!)
- Why can't development be efficient?

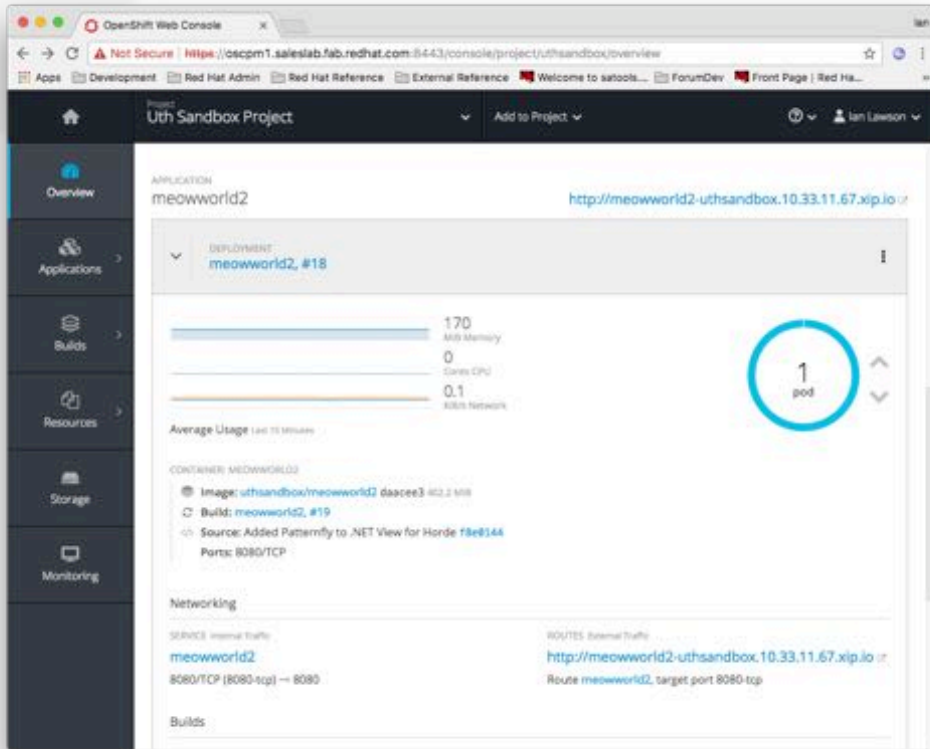


© Scott Adams



© UserFriendly.Org

Soooo, OpenShift



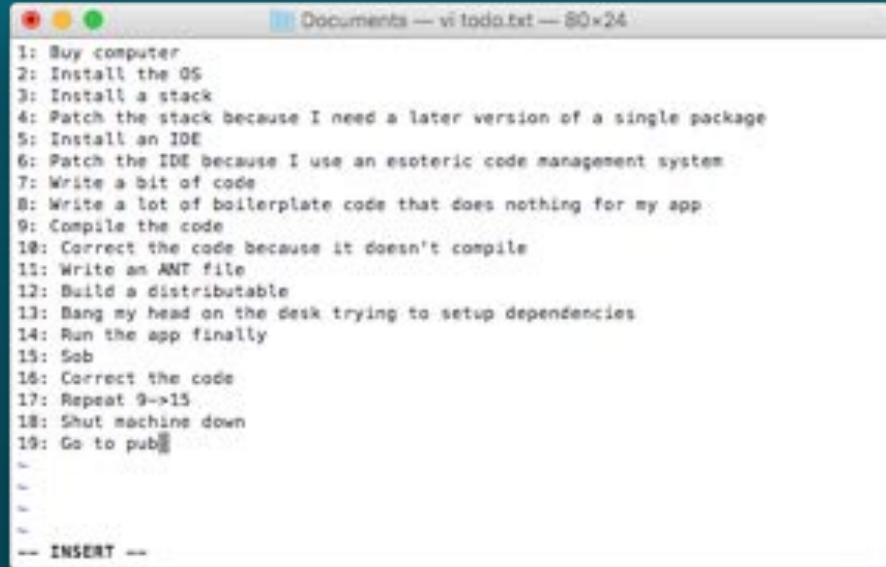
- Designed to ease the Developer's pain
- Let's the Developer *develop*
- Solves a lot of the core problems of day to day development
- Makes Ops life *much* easier
- Stops the inefficient waste of developer's valuable time

But why?

(Kudos to anyone who spots this is .NET....)

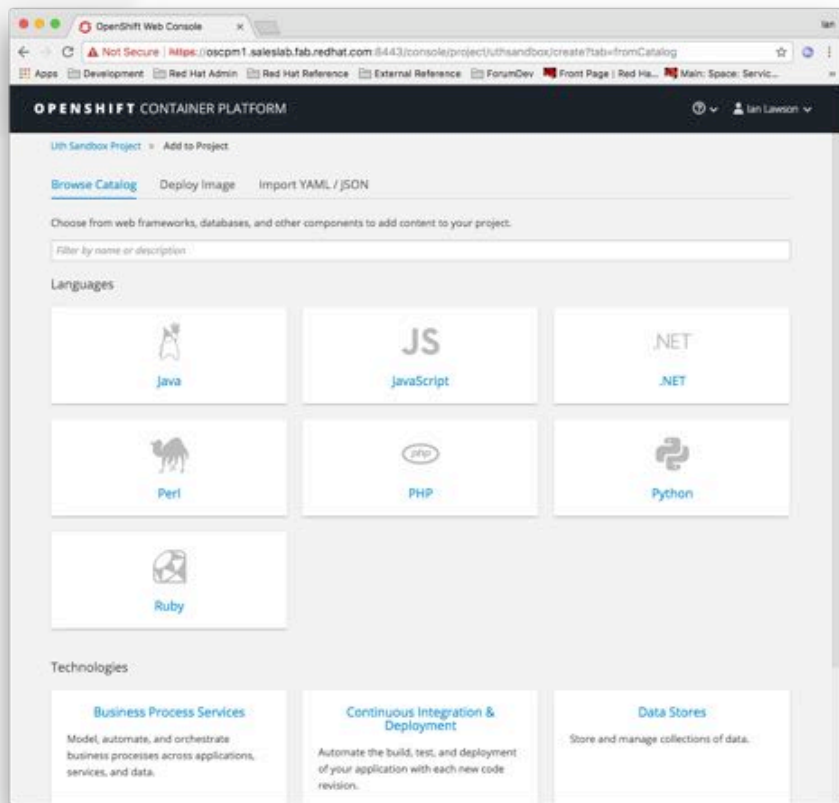
CORE OLD-SCHOOL DEV PROBLEMS

- ENVIRONMENTS!
- INCONSISTENCY!
- BOILERPLATEism!
- EVIL OPSism!
- COMPLEX BUILDism

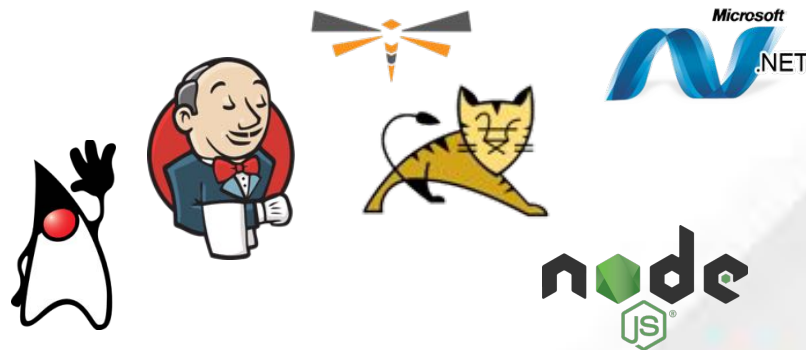


```
Documents — vi toda.txt — 80x24
1: Buy computer
2: Install the OS
3: Install a stack
4: Patch the stack because I need a later version of a single package
5: Install an IDE
6: Patch the IDE because I use an esoteric code management system
7: Write a bit of code
8: Write a lot of boilerplate code that does nothing for my app
9: Compile the code
10: Correct the code because it doesn't compile
11: Write an ANT file
12: Build a distributable
13: Bang my head on the desk trying to setup dependencies
14: Run the app finally
15: Sob
16: Correct the code
17: Repeat 9->15
18: Shut machine down
19: Go to pub
-- INSERT --
```

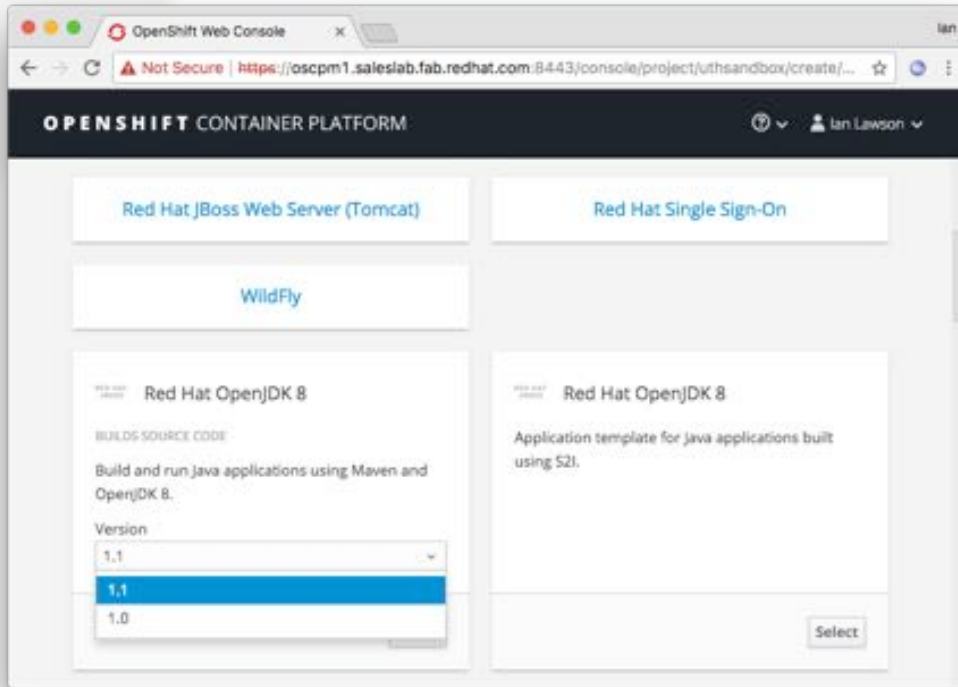
ENVIRONMENTS OPENSIFT!



- Containers flip it on it's head
- Developers just pick the base technology and start developing
- Complete abstraction from the underlying infrastructure
- Total Ops control over what the Devs can consume



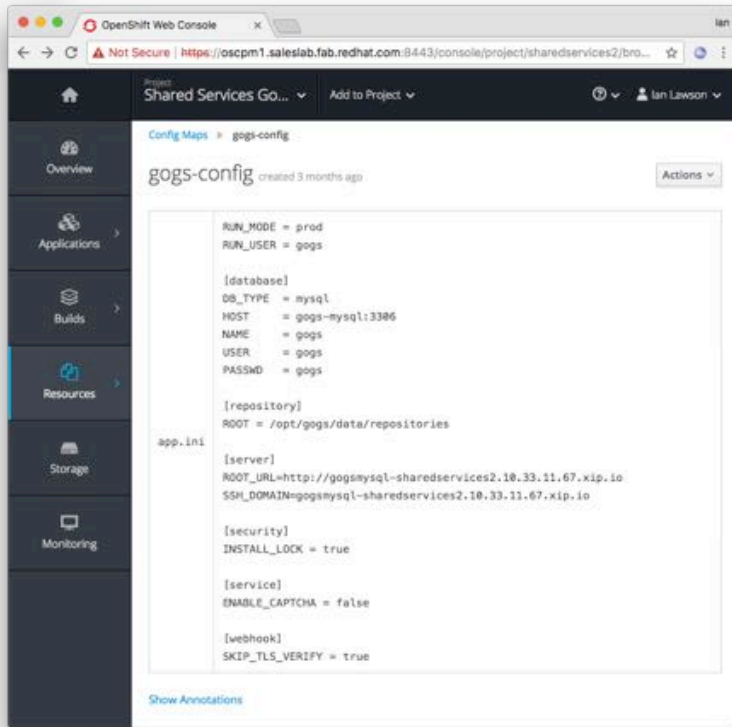
~~INCONSISTENCY~~ OPENSIFT!



- Developers like new and shiny
- But some like sticking with older versions
- *New and Shiny* is often incompatible with *Old and Reliable*
- OpenShift negates the old ‘end of Sprint, everyone has used different versions’ problem
- Ops can control exactly what versions can and should be used
- Definitive versioning but totally configurable by Ops



~~BOILERPLATE~~ ISM OPENSIFT!



The screenshot shows the OpenShift Web Console interface. The browser address bar indicates the URL: `https://oscpm1.saleslab.fab.redhat.com:8443/console/project/sharedservices2/bro...`. The console header shows the project name "Shared Services Go..." and the user "lan Lawson". The main content area displays the configuration for a ConfigMap named "gogs-config", which was created 3 months ago. The configuration is shown in a code editor with the following content:

```
app.ini
[database]
DB_TYPE = mysql
HOST = gogs-mysql:3306
NAME = gogs
USER = gogs
PASSWD = gogs

[repository]
ROOT = /opt/gogs/data/repositories

[server]
ROOT_URL=http://gogsmysql-sharedservices2.10.33.11.67.xip.io
SSH_DOMAIN=gogsmysql-sharedservices2.10.33.11.67.xip.io

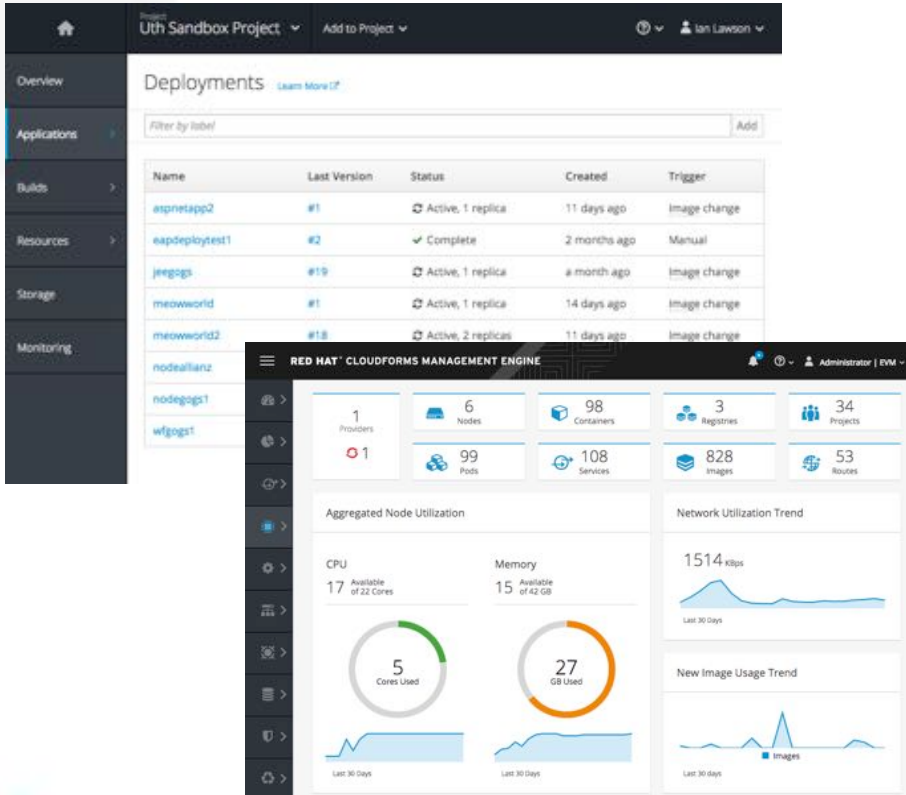
[security]
INSTALL_LOCK = true

[service]
ENABLE_CAPTCHA = false

[webhook]
SKIP_TLS_VERIFY = true
```

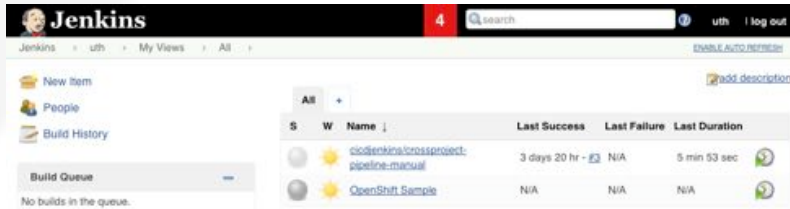
- J2EE and home/remote/local interfaces.....
- Too much time is spent coding the same boilerplate which has 0% actual functionality in it
- OpenShift counters this by providing templates, pre-defined Containers and dependency injection via Env, File or ConfigMap
- Developers can focus on *pure* functionality
- More efficient Developers = happier management
- You can define your own templates that pre-define as much of the applications space as you need

~~EVILOPS~~ism OPENSIFT!



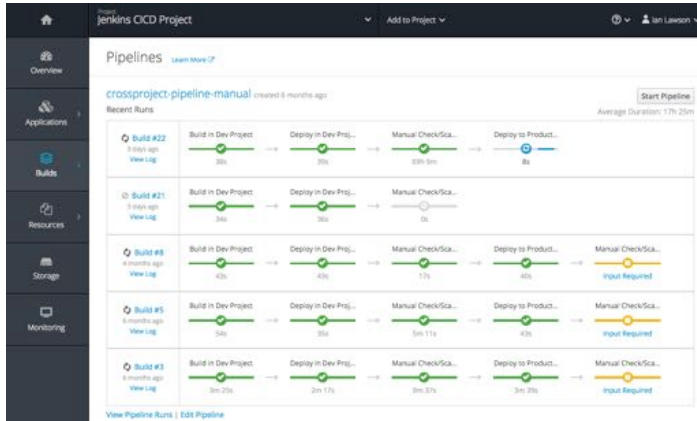
- Ops and Dev need to be the ***same*** team
- Far too easy in the old days for Dev to blame Ops
- And Ops to blame Dev
- When the problem was over complex app requirements on platform and....
- Too long cycles for platform provision
- OSCP abstracts the platform
- OSCP contains the application
- OSCP neatly divides responsibility
- Dev builds the functionality
- Ops decides the deployment

COMPLEX BUILDism OPENSIFT!



The screenshot shows the Jenkins dashboard. At the top, there's a header with the Jenkins logo, a search bar, and user information. Below the header, there's a sidebar with navigation options like 'New Item', 'People', 'Build History', and 'Build Queue'. The main area displays a table of recent builds.

S	W	Name	Last Success	Last Failure	Last Duration
		cicdenkins/crossproject-pipeline-manual	3 days 20 hr - 📄	N/A	5 min 53 sec
		OpenShift Sample	N/A	N/A	N/A



The screenshot shows the Jenkins Pipelines view for a project named 'crossproject-pipeline-manual'. It displays a sequence of build stages for several builds. The stages are: 'Build in Dev Project', 'Deploy in Dev Proj...', 'Manual Check/Sc...', and 'Deploy to Product...'. The progress of each stage is indicated by a green bar and a percentage.

Build	Build in Dev Project	Deploy in Dev Proj...	Manual Check/Sc...	Deploy to Product...	Manual Check/Sc...
Build #22	38%	38%	83% 5m	8%	
Build #21	34%	34%	8%		
Build #8	43%	43%	17%	43%	Input Required
Build #5	54%	32%	34% 11s	43%	Input Required
Build #3	34% 25s	2m 17s	34% 27s	3m 33s	Input Required

- A build process is NOT the app!
- A build process should ease the work, not add to it
- Too much time and too many brain-cycles used on the Build process as opposed to the functionality
- The Build process should be the time the Dev goes off for a coffee/pint/e-cig
- OSCP has embedded the Jenkins Pipeline functionality
- OSCP allows *all* stages of the Build/Test/Deploy to be fully automated and controlled
- But most importantly, OSCP does NOT impose a mode of operation on you

YES, YES, BUT WHY SHOULD I BE EXCITED?

- Micro Services!
- Micro Services!
- Micro Services!
- And Micro Services!
- But seriously.....

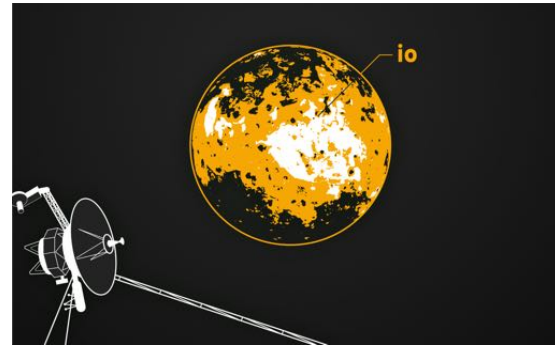


VERT.X



OPENSIFT FOR DEVELOPERS

- So, above and beyond the basics that OpenShift gives what exciting stuff is coming for Devs?
- RHOAR!
- ... or – Red Hat OpenShift Application Runtimes
- Openshift.io!
- ... which is specifically designed to ease Devs into the process of Micro services as a team



WHY SHOULD A DEV CARE ABOUT MSA?

- Because it's common sense for Application development
- Because it makes it easier to develop functionality with minimal impact
- Fine grain functionality = more efficient development
- Loose coupled behaviour = faster development

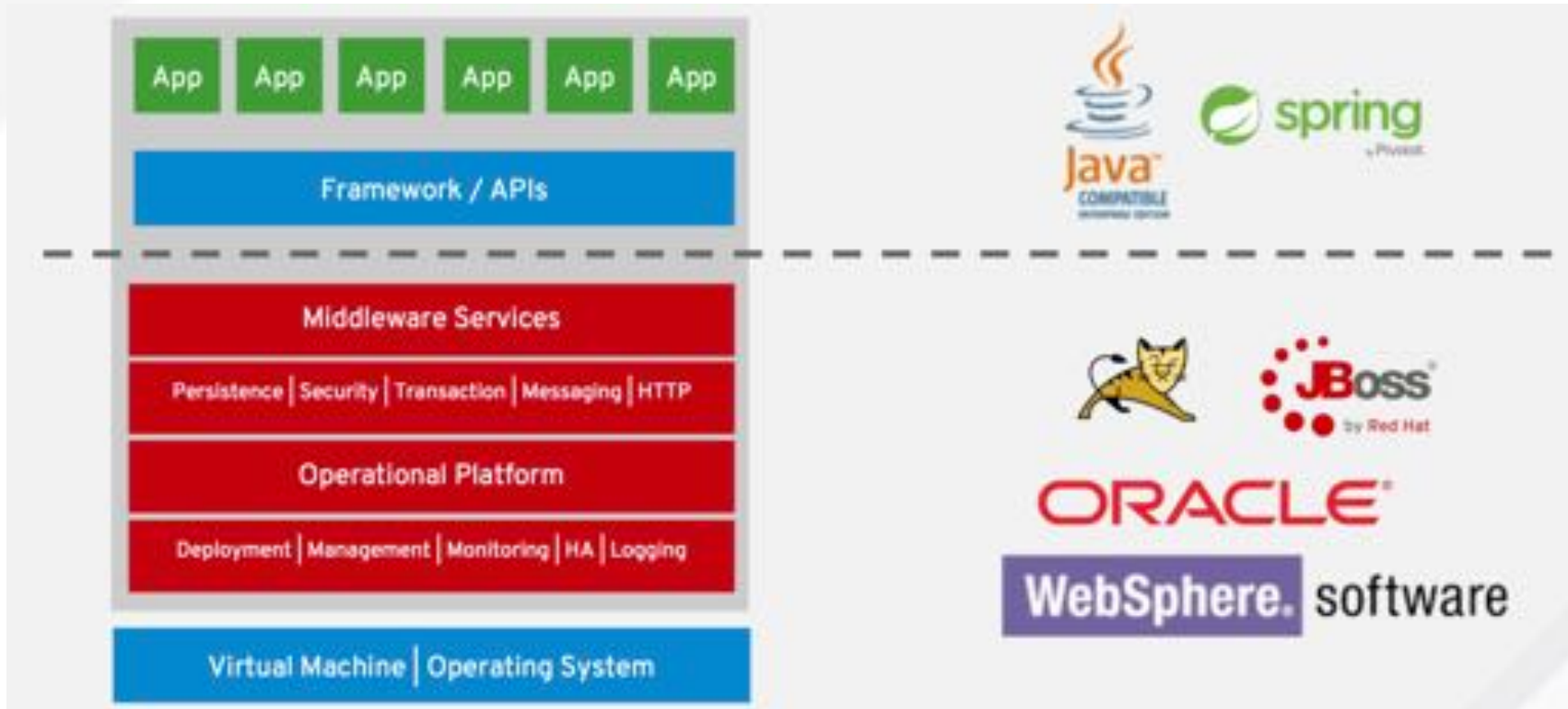


WHY SHOULD OPS CARE ABOUT MSA?

- Because you can intelligently deploy Micro services to take advantage of the infrastructure
- Because it gives you a fine grain level of control over Application behaviour
- Because it allows you to respond in an Agile way to end consumer requirements



THE 'GOOD OLD DAYS' – 2000 to 2014



THE NEXT GENERATION - 2017+



VERT.X

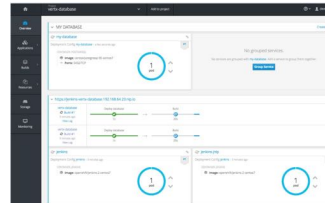
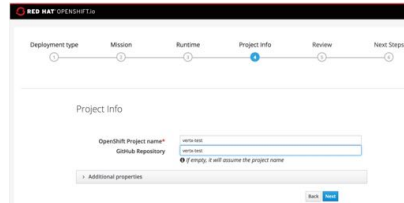
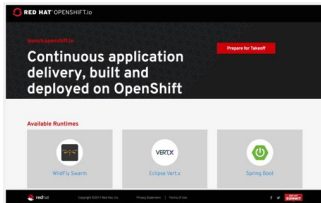


OPENSHIFT
by Red Hat



OPENSIFT.IO – A WHISTLESTOP TOUR

- Designed to make Micro service development *easy*
- Wizards and Quickstarts to rapidly enable Devs to get going
- Simplifies access into OpenShift, lets the Devs focus on Developing
- Enhanced Collaborative Tools for Team Working
- All the Dev tools needed available via a Browser (Eclipse Che, OSCP)
- Linked directly with RHOAR to offer varied mechanisms
- Coming very soon!
- <https://openshift.io> to get on the waiting list.....



BUT BACK TO THE PASSION

- Developers can be a misunderstood bunch
- We write software because we love to write software
- Take advantage of that – give Devs the tools that encourage efficiency
- OpenShift ticks all the boxes for letting Devs Dev, lettings Ops Ops
- But don't take my word for it – try it yourself.

<https://developers.redhat.com/products/cdk/hello-world/> - IT'S FREE!

Engage with Red Hat for a free Proof Of Concept around OSCP – IT'S FREE!



VISIT THE DEMO ROOM

ATTEND THE OTHER OPENSIFT PRESENTATIONS

Specifically - 'Speed Dating with OpenShift' and 'JEE Migration'

BUY ME A BEER AND PICK MY BRAINS



RED HAT

FORUM

Europe, Middle East & Africa