

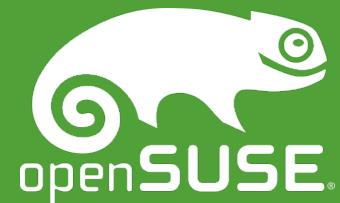
# openSUSE Kubic

The Container Platform by openSUSE

<https://kubic.opensuse.org>

Sarah Julia Kriesch

sarah.kriesch@opensuse.org



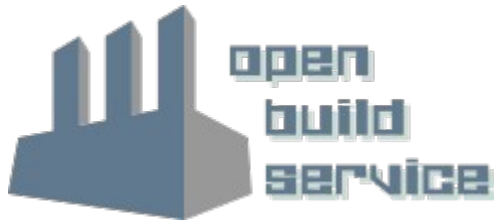
# Agenda

- About me
- The openSUSE project
- openSUSE Kubic
- Features
- Transactional Updates
- Velum as a Dashboard
- openSUSE Leap and Kubic
- Podman
- How to contribute



# About me

- Student in Computer Science (B.Sc.), TH Nürnberg
- Student Representative in Senate/ University Council
- Student Representative in AG Study Plan, AG Laboratories
- Founder AG Open Source (Educator + Trainer), TH Nürnberg
- Global Translation Coordinator, QA, Wiki, Advocate (openSUSE) at openSUSE
- Chairman at One Week Experience e.V. (One Week Student)



# openSUSE Kubic project

- Container as a Service Platform
- MicroOS, based on openSUSE Tumbleweed
- Velum, Cluster Bootstrapper & Cluster Dashboard
- Based on SUSE Container as a Service Platform (CaaSP)



# Features

- Transactional Updates
- Latest packages with openSUSE Tumbleweed
- Based on Podman
- Created with Docker base(part of Atomic Project)



# Transactional Updates

- Is Atomic:
  - Fully applied or not at all
  - Update does not influence the running system
- Can be rolled back:
  - A failed or incompatible update can be discarded and the previous system can be restored



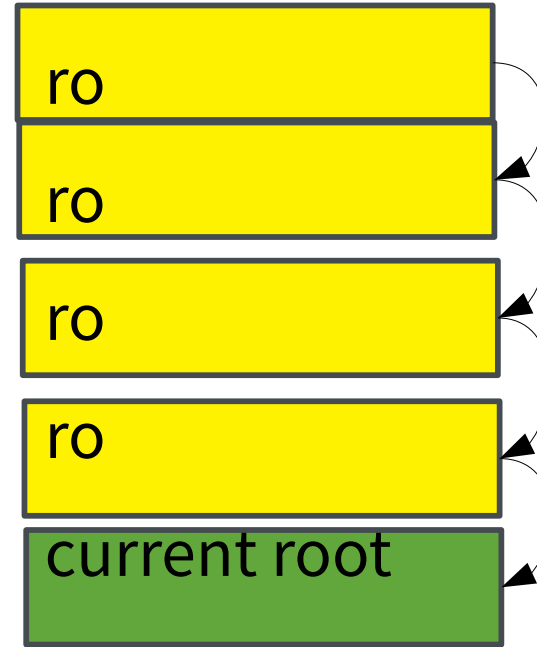
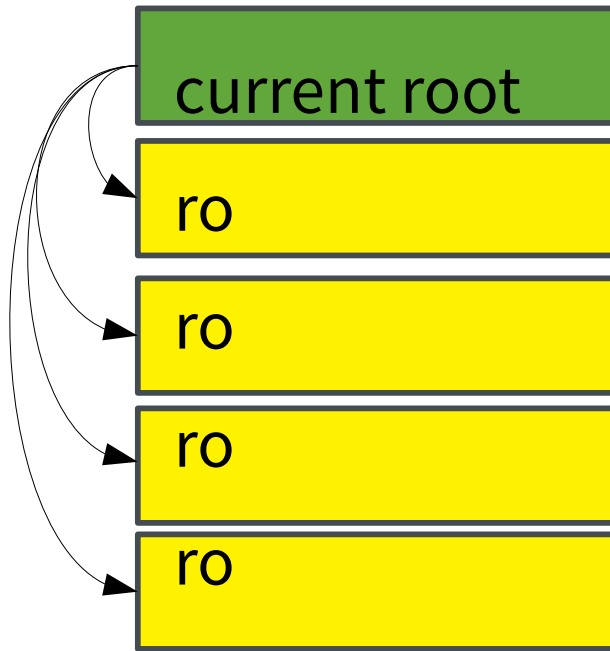
# Requirements

- Btrfs root filesystem with snapper (snapshots)/  
rollback enabled
- zypper
- btrfsprogs





# Snapshots vs. Transactional Updates



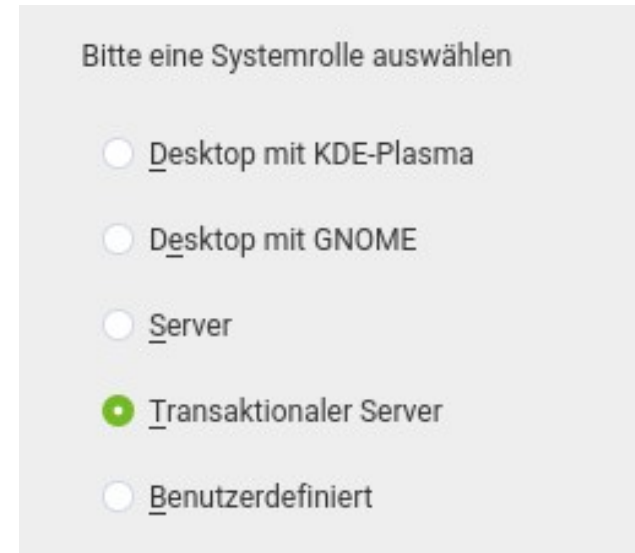
# Transactional Update

- Clone of current root to ro
- Change rw
- zypper up
- Change to ro
- “Rollforward” with btrfs subvol set-default
- After reboot change / update executed
- Quickly rollback



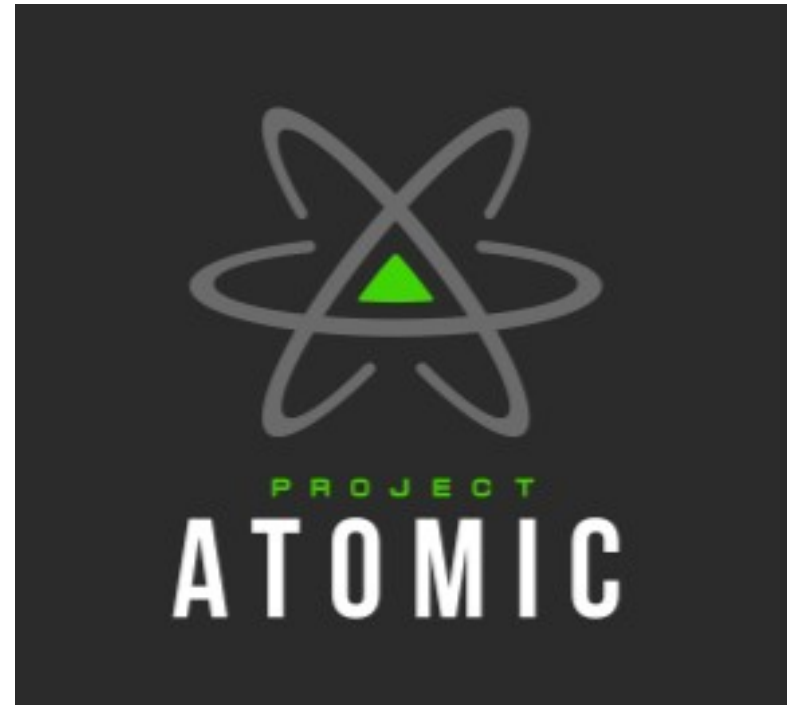
# Executing Transactional Updates

- Update/Upgrade:  
transactional-update  
[up|dup|patch]
- Installation/ Remove /  
Update 1 package:  
transactional-update pkg  
[install|remove|update]  
pkg
- Rollback:  
transactional-update  
rollback



# Project Atomic

- Container Tools  
(incl. Compose)
- Integrated docker  
commands
- Easy to use with  
container experience



# Velum as a Dashboard

The screenshot displays the SUSE CaaS Platform dashboard. At the top, there is a navigation bar with 'SUSE® CaaS Platform', 'Home', 'Settings', and a 'Logout' button. A yellow warning banner states: 'A supported deployment of SUSE CaaS Platform requires a minimum of three nodes.' Below this is the 'Cluster Status' section, which includes a 'Summary' table and a 'Nodes' table.

**Summary**

Total nodes	2	Updates	Manual
Master nodes	1	# of nodes w/ outdated software	0
New nodes	0		

**Nodes**

Online	Applied	ID	Hostname	Role	Actions
		4f16ea66e6bc485dbc030b2b5aa9f905	master-0.devenv.caasp.suse.net	<input checked="" type="checkbox"/> master	
		02534239728c40a6919b21ffb678b597	worker-0.devenv.caasp.suse.net	<input type="checkbox"/> worker	

**Pending Nodes**

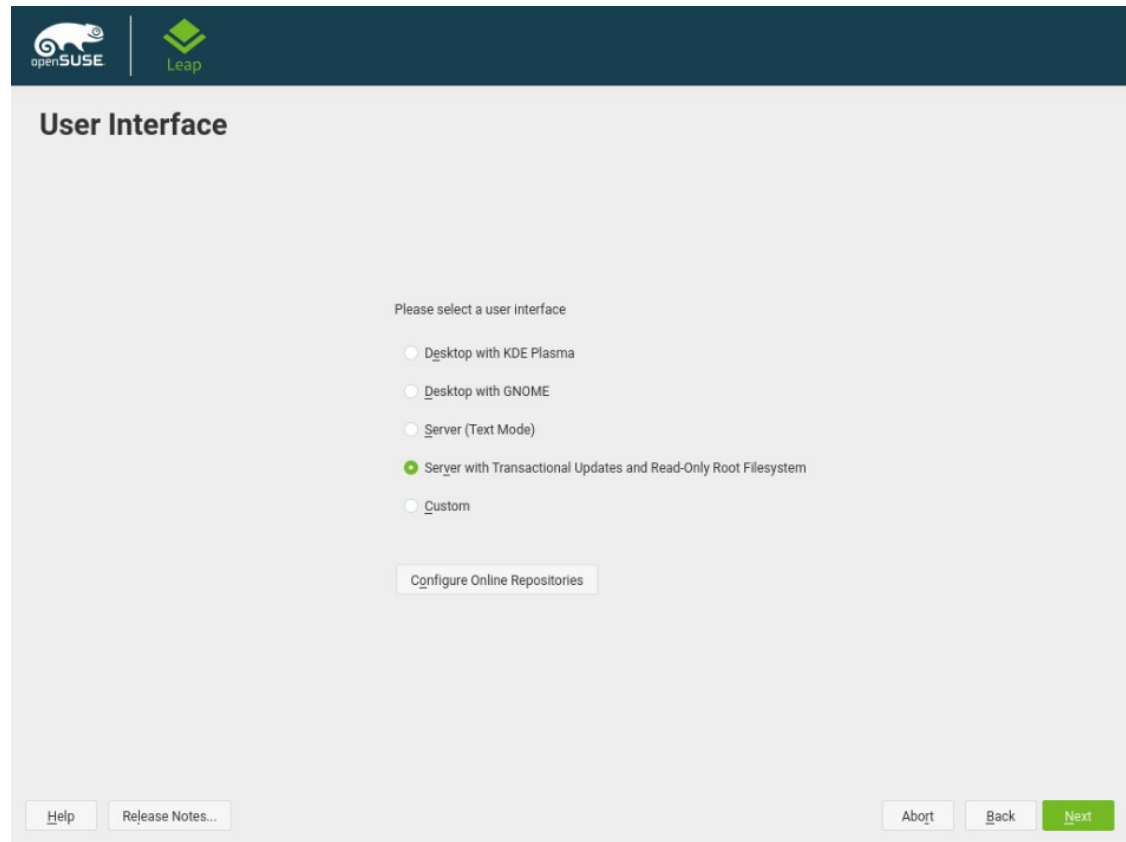
You currently have no nodes to be accepted for bootstrapping.

SUSE® CaaS Platform 3.0.0 | © SUSE Linux 2018



# How does Leap profit from Kubic?

- Transactional Updates available in Tumbleweed and Leap



# kubeadm

- toolkit produced by Kubernetes
- Creation of kubeadm nodes for a cluster
- Initialization of the network and joining a node:
- *kubeadm init --pod-network-cidr=10.244.0.0/16*
- *kubeadm join*

## openSUSE MicroOS

- System designed for running containers and optimised for large deployments.
- Includes Docker Open Source Container Runtime by default

## kubeadm Node

- MicroOS plus kubeadm kubernetes bootstrapping tool
- Create a cluster with `kubeadm init`
- Join a cluster with `kubeadm join`

# kubicctl

- Adding Worker Nodes:
  - *kubicctl node add node1,node2,...*
- Verifying the cluster:
  - *kubectl get nodes*





# Verification of the cluster

```
master:~ # kubectl get nodes
NAME        STATUS    ROLES    AGE      VERSION
master     Ready    master   8m       v1.11.1
node1      Ready    <none>   37s     v1.11.1
master:~ #
```



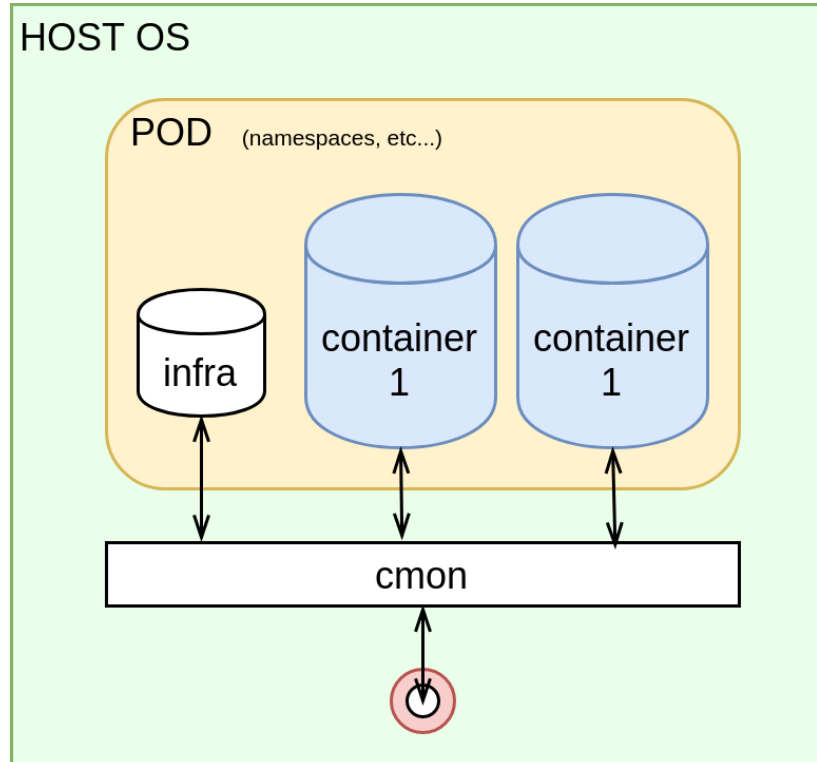
# Kubic Node Roles

- Kubic Admin Node:
  - kubecd  
(daemon which communicates via gRPC with clients and setup of Kubernetes network)
  - salt-master (for node management via Dashboard)
  - Kubernetes Master Node
- Kubic Loadbalancer Node:
  - MicroOS without container runtime
  - haproxy installed
- Additional Kubic Node:
  - Additional Master for HA
  - Worker Node

# Podman

- Alternative container engine instead of docker
- Same commands as docker
- Daemonless and rootless containers possible
- Container Manager as docker-compose
- Sharing of one network namespace
- Compatible with multiple image formats including the OCI and Docker image formats

# Pods



- 1 Kubernetes pod in 1 Linux namespace and the same cgroup
- Flexible combinations of services possible (webserver, github pull, database, ...)

# Daemonless

- Docker daemon is (mostly) executed by root
- Podman is working without central daemon
- Additional process *common* is monitoring and gives correct start parameters for the container runtime

# Rootless

- Start of a container without root privileges
  - Mapping of the UID in the namespace
  - Using root in a container possible (isolating root)
  - User in the container has default access on host system
- 
- Docker offers that since version 19.03

# Example of a rootless podman process

```
$ podman run --name nginx -d -p 8080:80 -v ~/public_html:/usr/share/nginx/html:ro nginx
```

Podman process (euid=\$UID)

Join existing user+mount namespace, or create one

Podman process (euid=0)

Setup the network namespace with slirp4netns

Setup the storage and the fuse-overlays mount

common

OCI runtime

container process

# Podman commands – as docker

- podman run
- podman exec
- podman info
- podman logs
- podman ps
- podman cp
- podman rm
- podman poll
- podman push
- podman build
- podman image
- podman images
- podman commit
- podman history
- podman stop
- podman start
- podman volume



# Podman (only) commands

- podman generate kube
- podman container cleanup
- podman container exists
- podman container restore
- podman image exists
- podman pod create
- podman pod kill
- podman pod ps
- podman pod pause
- podman pod restart

“Do you wanna become a Contributor?”



# Packaging

- Creation of packages with the Open Build Service
- Submit to openSUSE Factory
  - > openSUSE Tumbleweed
- <https://build.opensuse.org/project/show/devel:kubic>



# Development of openSUSE Kubic

- Github contributions:
  - <https://github.com/kubic-project>



# Quality Assurance

- Test it!
- Try it!
- Perform it!
- Report bugs!
- Bugzilla:  
<https://bugzilla.opensuse.org/buglist.cgi?component=Kubic&product=openSUSE%20Tumbleweed>



# References

- openSUSE Kubic:

<https://kubic.opensuse.org>

- <https://www.projectatomic.io>

- Podman:

<https://jaxenter.de/docker/podman-container-daemonless-rootless-89349>

- Podman Usage:

<https://github.com/containers/libpod/blob/master/transfer.md>





Questions?

The background features a large, dark green arrow pointing to the right, which is part of a larger geometric pattern of interlocking shapes in various shades of green and blue. The text is overlaid on the dark green arrow.

openSUSE Summit 2020

Dublin

May 27 – 28, 2020

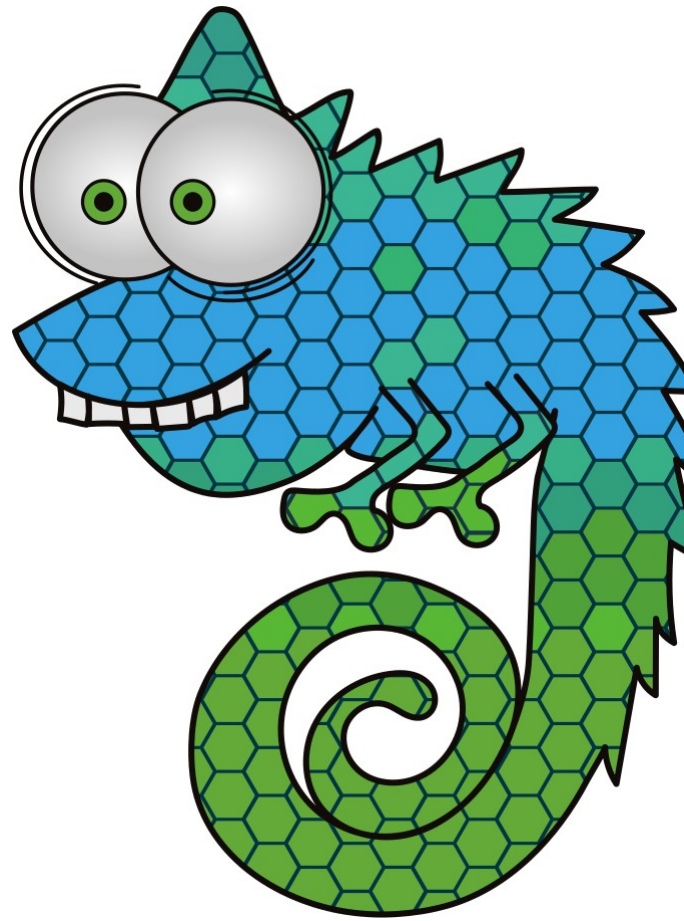
<https://events.opensuse.org>



Join our conference,  
contribute & have a lot of fun!  
[events.opensuse.org](https://events.opensuse.org)

**Thank you.**





**Have a Lot of Fun, and Join Us At:**

[www.opensuse.org](http://www.opensuse.org)



## License

This slide deck is licensed under the Creative Commons Attribution-ShareAlike 4.0 International license. It can be shared and adapted for any purpose (even commercially) as long as Attribution is given and any derivative work is distributed under the same license.

Details can be found at <https://creativecommons.org/licenses/by-sa/4.0/>

## General Disclaimer

This document is not to be construed as a promise by any participating organisation to develop, deliver, or market a product. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. openSUSE makes no representations or warranties with respect to the contents of this document, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. The development, release, and timing of features or functionality described for openSUSE products remains at the sole discretion of openSUSE. Further, openSUSE reserves the right to revise this document and to make changes to its content, at any time, without obligation to notify any person or entity of such revisions or changes. All openSUSE marks referenced in this presentation are trademarks or registered trademarks of SUSE LLC, in the United States and other countries. All third-party trademarks are the property of their respective owners.

## Credits

### Template

Richard Brown  
[rbrown@opensuse.org](mailto:rbrown@opensuse.org)

### Design & Inspiration

openSUSE Design Team  
<http://opensuse.github.io/branding-guidelines/>