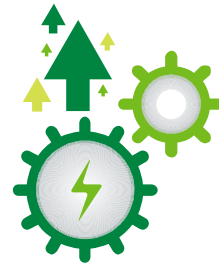




Just Enough Operating System to kick start creativity

Simona Arsene

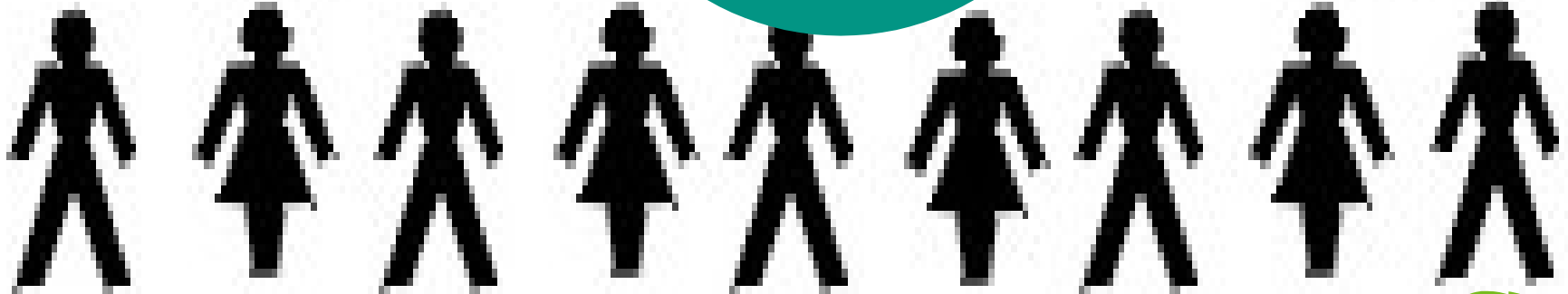
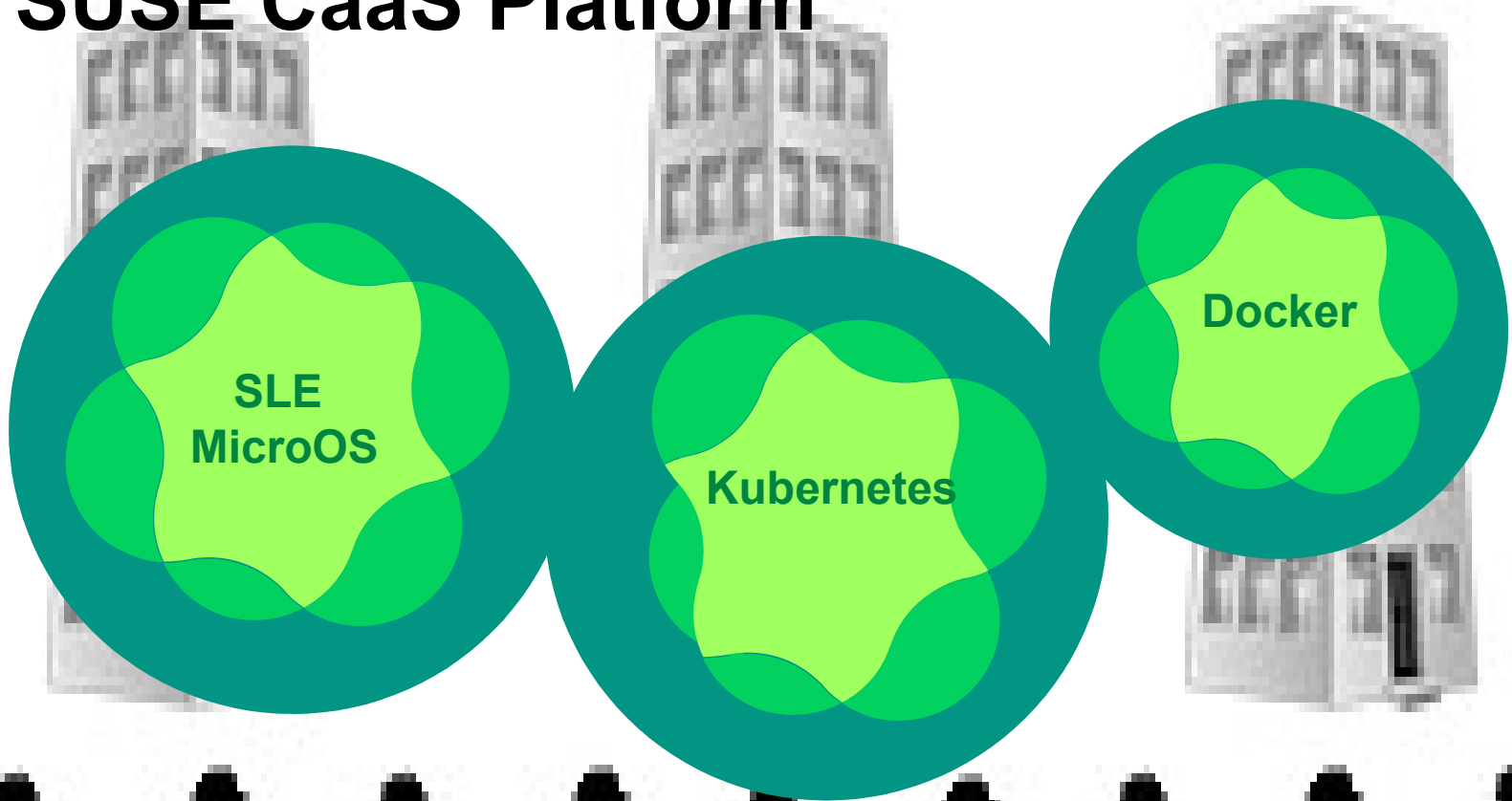
SUSE® Linux Enterprise Server JeOS speeds up virtual image deployment



SUSE Containers as a Service Platform



SUSE CaaS Platform

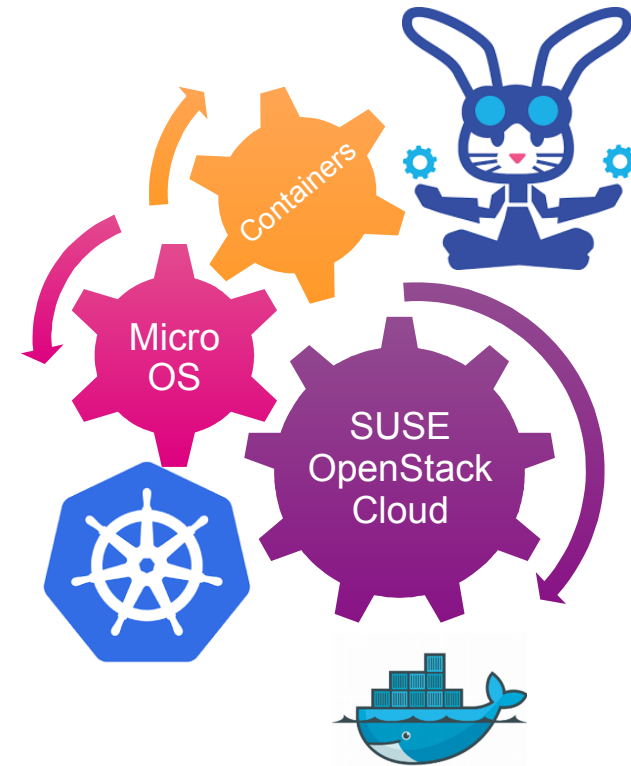


What is SLE MicroOS?

SLE MicroOS:

Modern Linux Operating System, designed for **containers** and optimized for **large deployments**.

MicroOS inherits the SLE knowledge while redefining the operating system into a small, efficient and reliable distribution.



MicroOS: What needs to be in?

Priority 1

- Running workloads in a multi-cloud vendor environment (Setup and manage a private cloud environment; Container-native Orchestration)
- Running a Container Cluster on premise or in a public cloud

Priority 2

- Optimizing resources within a Private/Public Cloud
- Running a PaaS
- Storage

MicroOS: What problems are we trying to solve?

SLE today:

- Covers multiple use cases: physical installation, virtual host/guest, container host
- Solves multiple problems in different use cases
- Is focused on one-off upgrade, manual upgrade, auto-update

Customers using containers need:

- A small and easy to manage/upgrade OS
- A fast way to setup a cluster and to manage multiple nodes
- An always up-to-date OS
- Transactional Updates

MicroOS: What are critical elements of our solution?

Installation:

- Minimal Installer for bare metal/Golden Images
- Cloud and virtualization: Provide ready to run images like with JeOS
- Optional: provide Driver Update Disks (DUDs)

Configuration

- System should be pre-configured as far as possible
- Cloud-init to configure the Base OS and minimal Kubernetes

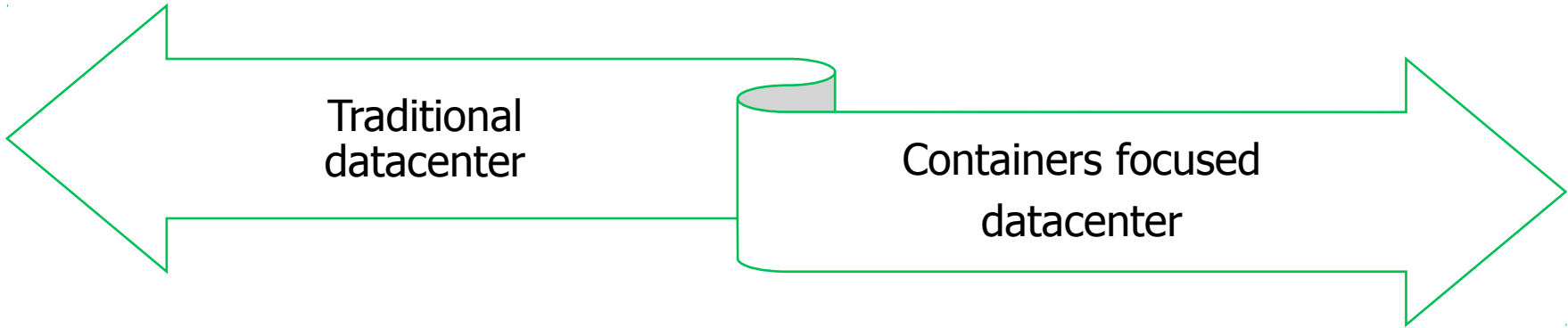
Scalability

- >25 nodes

Updates:

- Transactional updates
- Maintenance window
- Automatic updates

MicroOS vs JeOS



JeOS

- General purpose OS
- Minimal images
- Easy to extend to full SLES
- Availability of all extensions
- Full control of the installed packages, updates, upgrades

MicroOS

- OS focused only on containers
- Minimal images
- Transactional Update
- Designed for Cloud, Kubernetes, PaaS
- Focused on large deployments
- Reduced end-user interactions

MicroOS: What differentiates us from existing solutions?

Technology: Btrfs as default file system (e.g. leverage copy-on-write)

RPMs for simplified upgrades (tbd)

SUSE Build Infrastructure

Security Certifications

**Partners
Eco System** Experience in the Linux/Open Source market

Enterprise customer base

Partner eco-system

Focus on partnership vs vendor lock-in

MicroOS: How are we gonna make money from it?

**Same
business
model as
for
classical
SLES**

Standard and Priority Subscriptions

Additional support offerings (PSE/DSE, etc)

Leverage OEM partnerships

New partners for new type of business (use cases where SLES is currently not considered)

Public Cloud Providers might show interest in a small OS

What is Kubernetes?

Kubernetes is an open-source platform for automating **deployment, scaling, and operations of application** containers across clusters of hosts, providing container-centric infrastructure.

What is Kubernetes?

With **Kubernetes**, customers are able to quickly and efficiently respond to customer demand:

Deploy your applications quickly and predictably.

Scale your applications on the fly.

Seamlessly **roll out** new features.

Optimize use of your hardware by using only the resources you need.

Customer Benefits

Running
Anywhere

DevOps

Accelerate
business
innovation

Enterprise-
grade
security and
scalability



Schedule



Container Infrastructure Platform(CIP) Milestones

- SUSECon – Let's talk about the initiative
- Mar 30, 2017 Public Beta
- May 24, 2017 RC
- Jun 15, 2017 GMC
- Jun 22, 2017 GM
- July 19 2017 FCS



**SUSE
Containers
as a Service
Platform**

**Coming
soon**



Unpublished Work of SUSE LLC. All Rights Reserved.

This work is an unpublished work and contains confidential, proprietary and trade secret information of SUSE LLC.

Access to this work is restricted to SUSE employees who have a need to know to perform tasks within the scope of their assignments. No part of this work may be practiced, performed, copied, distributed, revised, modified, translated, abridged, condensed, expanded, collected, or adapted without the prior written consent of SUSE.

Any use or exploitation of this work without authorization could subject the perpetrator to criminal and civil liability.

General Disclaimer

This document is not to be construed as a promise by any participating company 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. SUSE 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 SUSE products remains at the sole discretion of SUSE. Further, SUSE 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 SUSE marks referenced in this presentation are trademarks or registered trademarks of Novell, Inc. in the United States and other countries. All third-party trademarks are the property of their respective owners.

