Expert Days Fall 2019
SUSE Application Delivery Solutions

Accelerate delivery of modern applications
SUSE Application Delivery Solutions

Accelerate development and deployment of containerized applications to speed application delivery and increase business agility.

- **SUSE CaaS Platform**
  - Container Management

- **SUSE Cloud Application Platform**
  - Platform as a Service

**Kubernetes for the Agile Enterprise**
- Best Kubernetes operator experience

**Advanced Cloud Native Developer Experience**
- Proven productivity for Kubernetes
Transformation is Multifaceted and Multi-cloud
What is a Microservice?

Variant of SOA architectural style that structures an application as a collection of loosely coupled services

Services should be fine-grained and use lightweight protocols

Designed to sufficiently decompose the application in order to facilitate agile application development and deployment
Old (monolithic) Way
Instead of Large, Monolithic Applications…

Interdependent components
Application centric
Functionally organized teams
12-18 month release cycles
A Better Way…
Build and Deliver Cloud Native Applications

Increase agility with Microservices

Smaller Codebase + Specialized Teams + Deconstructed Services = Agile IT

= Opportunity!
Build and Deliver Cloud Native Applications

Containers enable Microservices model

Fast start/stop

Densification

In-place upgrades (we destroy and re-deploy)
Re-deploy to Virtual or Cloud Infrastructure
Developers Use Containers to Package Code

Convenient to use
- Small
- Fast to start up
- Easy for developers to build and deploy themselves

Operate consistently in any environment
- Include everything needed to run
- Enhance productivity and facilitate collaboration
- Eliminate ‘works on my machine’
DevOps Workflows for Kubernetes
A spectrum of automation options

Custom-made | Assembled | Modified | Factory-built

Greater flexibility and versatility

Faster time-to-deliver
Lower cost per unit
Higher re-use (more efficient)
More scalable production

SUSE Application Delivery Solutions
Demo
ISVs Use Containers to Deliver Software TODAY!
SUSE CaaS Platform Partners
Automation Enables Deployment at Cloud Scale

Orchestration
• Scheduling
• Service discovery

Performance and availability
• Scaling
• Load balancing
• Self-healing
• Monitoring

Maintenance
• Rollout
• Rollback
Kubernetes is an Ideal Multi-cloud Foundation
Requires advanced multi-cluster and workload management
Application Delivery Transformation Journey

**Containerize**
- Containerize & Orchestrate
  - Streamline application development and delivery

**Modernize**
- Microservices & DevOps
  - Refactor existing applications
  - Build and deliver new cloud native applications

**Formalize**
- Best practices, patterns, processes
  - Automate DevOps workflow

**Standardize**
- Factory-scale production
  - Scale with efficiency and control
Kubernetes Underpins Modern Application Delivery

**Containerize**
- Containerize & Orchestrate

**Modernize**
- Refactor existing applications
- Microservices & DevOps

**Formalize**
- Automate DevOps workflow
- Best practices, patterns, processes

**Standardize**
- Scale with efficiency and control
- Factory-scale production

- Streamline application development and delivery
- Build and deliver new cloud native applications

Kubernetes: Container Management
SUSE CaaS Platform Opportunities

**Containerize**
- Containerize & Orchestrate
- Streamline application development and delivery

**Modernize**
- Microservices & DevOps
- Build and deliver new cloud native applications

**Formalize**
- Best practices, patterns, processes
- Automate DevOps workflow

**Standardize**
- Factory-scale production
- Scale with efficiency and control

SUSE CaaS Platform (on-prem Kubernetes)

Kubernetes Container Management
SUSE Cloud Application Platform Opportunities

**Containerize**
- Containerize & Orchestrate
- Streamline application development and delivery

**Modernize**
- Microservices & DevOps
- Build and deliver new cloud native applications

**Formalize**
- Best practices, patterns, processes
- Refactor existing applications

**Standardize**
- Factory-scale production
- Automate DevOps workflow

SUSE Cloud Application Platform

Multi-cloud Kubernetes
- SUSE CaaS Platform
- AKS, EKS, GKE

Kubernetes Container Management

Scale with efficiency and control
SUSE in the CNCF Landscape
SUSE CaaS Platform 4
Kubernetes for the agile enterprise

Accelerate modern application delivery with Kubernetes, today’s leading container management platform

Simplify Kubernetes administration with an exceptional platform operator experience

Maximize return on investment with a flexible, no lock-in solution
Accelerate Application Delivery

Streamline application development and deployment

Build and deliver new cloud native applications

Modernize legacy applications
SUSE Container as a Service Platform 4.0

Now built on SUSE Linux Enterprise Server 15 Service Pack 1

- Latest release of Kubernetes: v1.15 at launch
- Simplified Deployment Model – Bare Metal, Virtual Infrastructure, Public Cloud
- Kubernetes update schedule aligned with upstream community (+90 days)
- Uptime! Non-disruptive cluster upgrades
- Enhanced network security policies via Cilium

First Customer Shipment: September 2019
Containerization and registry

- Most of SUSE CaaS Platform is now delivered as containers
  - Only kubelet, kubectl, kubeadm, CRI-O installed into container host OS
    - As pattern SUSE-CaaSP-Node
  - Containers are delivered from registry.suse.com – not wrapped as RPMs
  - Architecture change makes it easier to update nodes without rebooting
    - CRI-O can be updated and restarted – it will reattach to running containers
## Release plan - quarterly updates

Sync with upstream Kubernetes (k8s) releases

<table>
<thead>
<tr>
<th>CaaS Platform</th>
<th>3.0</th>
<th>4.x</th>
</tr>
</thead>
<tbody>
<tr>
<td>K8s version</td>
<td>1.9</td>
<td>1.10</td>
</tr>
</tbody>
</table>

Incremental updates - cannot skip an update.
All updates (Kubernetes and OS) have to be non-disruptive.
Where does CAASP platform run…?

- Anywhere SLES15 SP1 x86_64 is available
- Bare Metal, Virtual Machines
- Public Cloud (BYOS)
- OpenStack Deployment
- VMware Infrastructure Deployment
Container Engine optimized for Kubernetes

- https://cri-o.io
CRI-O Impact on docker users

- **No** need to change container images
- **No** need to change the way to distribute images (pull from docker registries)
- **No** need to change Kubernetes manifest files
- The Container Runtime Interface is completely transparent to end-user
- However, debugging on a node is a bit different
Why CRI-O? – Less Complexity!

- Tech preview in v3, supported container engine with v4
- Deliver a component that just does the job - and nothing more
- docker open-source engine is not optimized for Kubernetes:
  \[\text{kubelet} \rightarrow \text{docker-shim} \rightarrow \text{dockerd} \rightarrow \text{containerd} \rightarrow \text{runC}\]
- Designed with Kubernetes in mind:
  \[\text{kubelet} \rightarrow \text{crio} \rightarrow \text{runC}\]
- Lightweight: offers better performance
- Easier to maintain and to debug
- Open Container Initiative (OCI) compliant: uses runC
SUSE CaaS Platform 4 Benefits

- CRI-O enables Kubernetes upgrades without cluster restart
  - Non-disruptive upgrades if using CRI-O and SUSE Linux Enterprise
- Prompt update to new SLES service packs
- High re-use of existing deployment and patch infrastructure – SUSE Manager
- Cluster node deployment via kubeadm will give:
  - Quicker release of upstream components
  - More flexible approach
Strengthen Application Security at Scale

Enable network security policies and more complex network topology

https://cilium.io/blog/2018/09/19/kubernetes-network-policies/
Cilium Label-Aware Scalability

Create This ONE Time and Scale Out

```yaml
endpointSelector:
  matchLabels:
    role = "backend"

ingress:
  matchLabels:
    role = "frontend"
```

OR...modify and grow this each time a pod is added

```bash
# Allow Apache to access Vitess/Mysql in pods [1,2]-2
iptables -i eth0 -p tcp -dport 3306 -s ${IP1-1} -d ${IP1-2}

iptables -i eth0 -p tcp -dport 3306 -s ${IP2-1} -d ${IP1-2}

iptables -i eth0 -p tcp -dport 3306 -s ${IP1-1} -d ${IP2-2}

iptables -i eth0 -p tcp -dport 3306 -s ${IP2-1} -d ${IP2-2}
```

Cloud Native Computing Foundation
Cilium as default network plugin

**Service Identity**
Visibility + filtering based on Kubernetes service labels, DNS-names, etc., not IP addresses.

**API-Aware Security**
Goes beyond TCP/UDP ports, natively understanding HTTP, gRPC, Kafka, DNS, & more.

**Performance & Scale**
BPF datapath and control plane optimized for highly dynamic, large scale envs with high throughput.

**Multi-Cluster Routing**
Provides simple, efficient, and secure connectivity between multiple Kubernetes clusters.

**Universal Encryption**
Adds encryption to all traffic between Cilium endpoints with no application/pod changes.

**Transparent to Apps**
By running in the kernel, BPF + Cilium require no changes by or coordinate with app teams.
SUSE CaaS Platform 4 Architecture (cont’d)

- Deliver container images via SUSE Registry
  - No more “containers as RPMs”
- Release and support Prometheus and Grafana containers
  - Industry standard monitoring and visualization
  - Consistent delivery and experience across CaaSP, SES, SOC, SUMA
  - Allow integration into customer logging strategies
    - rsyslog, Elastic stack, Splunk, etc.
## SUSE CaaS Platform

**Based on SUSE Linux Enterprise 12 SP3 and Kubernetes 1.9**
- LDAP integration
- Airgapped install
- Nginx ingress controller
- Documentation update
- Kubernetes 1.10 update

### New Architecture:
- Kubeadm based
- Command line first
- More flexible

### Container Host OS:
- Codebase: SUSE Linux Enterprise 15 SP1
- CRI-O as container engine

### Orchestration:
- Kubernetes 1.15
- Network options (Cilium as first plugin)

**Planned for 4x Releases**
- Kubernetes version updates
- Customer certificates for external endpoints
- GPU enablement
- Azure/AWS/GCE integration
- Prometheus metrics/Grafana dashboards
- Stratos operations GUI
- CSI certification for third-party storage
- CNI: Kuryr as network plugin
- Sandboxed containers
- Support for transactional updates
- Service mesh

**Planned for 5 Release**
- Based on SUSE Linux Enterprise 15 SP2
- Kubernetes version updates
- IPv6 support
- ARM64 port

### Overall Themes
- Continue to make Kubernetes easy to install, update, operate, and secure
- Multi-cluster, Multi-cloud
- Integration into customer environments (storage, networking)

*Information is forward looking and subject to change at any time.*
SUSE Cloud Application Platform

A modern application delivery platform that brings an advanced cloud native developer experience to Kubernetes. SUSE Cloud Application Platform increases business agility by helping enterprises to:

- Boost developer productivity
- Reduce complexity and increase IT efficiency
- Maximize ROI
Modular Kubernetes Solution Architecture
Flexible packaging to match your needs

Kubernetes user experiences

- SUSE CaaS Platform
  Kubernetes for the Agile Enterprise

DIY DevOps Automation

- SUSE Cloud Application Platform
  Advanced Cloud Native Developer Experience

Kubernetes operator experiences

- Amazon EKS
- Microsoft AKS
- Google GKE
Boost Developer Productivity

With easy one step deployment of cloud native applications using the language and framework most appropriate for the task.

- One step application deployment
- Develop and deploy software solutions faster than ever before
- Leverage your in-house skills
Reduce Complexity & Increase IT Efficiency

With a single, lean, platform that brings together proven open source technologies for rapid application delivery at scale.

• Multiple cloud deployment models
• Increased efficiency with a lean and fast platform
• Easy to install with Helm charts and managed in Kubernetes
Maximize Return on Investment

With industry leading open source technologies that leverage your existing investments.

- Lower risk by aligning with industry momentum
- Gain the benefits and value of a thriving ecosystem of contributors
- Protect your investment with a rock solid open source core
LEADING THE WAY TO CLOUD FOUNDRY AND KUBERNETES INTEGRATION

SUSE Cloud Application Platform

- Cloud Foundry in Kubernetes
- Containerized Cloud Foundry Application Runtime
- Enterprise Ready
- Kubernetes-native scheduler
SUSE Application Delivery Solutions
Community Involvement
SUSE is Committed to Cloud Foundry

As a long-time supporter of Cloud Foundry, SUSE contributes to the project and its Foundation in many ways:

- Platinum member of Cloud Foundry Foundation

- SUSE Cloud Application Platform is a certified distribution

- Originator and Project Lead for CF Containerization

- Originator and Project Lead for Stratos

- Early adopter and influencer of Project Eirini
CF Containerization

Packages Cloud Foundry Application Runtime (CFAR) as containers instead of virtual machines, allowing CFAR to be deployed to Kubernetes

- Originated at SUSE; SUSE is project lead
- Eliminates requirement for BOSH
- Allows your organization to standardize on Kubernetes as your hosting platform
- Much smaller CFAR footprint than VM based distribution. (Min 32GB vs 128GB.)
- Easy to start small and scale up without breaking the bank
Stratos

Web-based UI for managing Cloud Foundry (and now, Kubernetes). It allows users and administrators to both manage applications running in the Cloud Foundry cluster and perform cluster management tasks.

- Originated at SUSE; SUSE is project lead
- Single pane of glass dashboard and metrics
- Aggregate multiple CF instances: SUSE-CAP, PCF, open source, or any other
- View of underlying Kubernetes infrastructure
- Customizable branding and styling
- Extensible for adding additional integrations
Project Eirini

Enables pluggable scheduling for CFAR (allows operators to choose whether CFAR should use Diego or Kubernetes to orchestrate application container instances)

• SUSE announced forthcoming support for Eirini at CF Summit EU in October 2018

• SUSE-CAP 1.4 is the first Cloud Foundry software distribution to support Eirini

• Run Cloud Foundry applications natively in Kubernetes

• Same Cloud Foundry developer experience

• Familiar Kubernetes operator experience
Cloud Foundry Application Runtime

A code-centric platform that simplifies the life of developers. It takes your code, written in any language or framework, and runs it on any cloud.

• One step command to containerize, deploy, and manage an application

• Automatically identifies and pulls in language libraries and frameworks via buildpacks

• Open source service brokers automatically create and bind services to applications

• Automates application lifecycle management by assigning appropriate resources, managing routing, load balancing, scaling, and more
CF Build Packs – Application runtime environment
A haiku...

”Here is my source code
run it on the cloud for me
I do not care how”

-ONSI FAKHOURI
With cloud native, time-to-market is accelerated

Before

Download Base Image

Patch Image

Add artifact
Install extra packages

Database connections

Network

Tests

After

> cf push
SUSE CAP - Public Cloud Deployment
Azure Open Service Broker / Cloud Provider Integration
Demo
SUSE Cloud Application Platform 1.5

Modern Application Delivery Platform for Kubernetes

- SUSE Cloud Foundry has been updated to version 2.18.0
- One-step application deployment model – (e.g. cf push)
- Policy Driven, Fault Tolerant, Self Healing Platform-as-a-Service Infrastructure
- Reference Architectures available for deployment on Public Cloud
- Optionally includes with SUSE Container as a Service Platform and SUSE Enterprise Storage for on-premise deployments

First Customer Shipment: September 2019
Multi-cloud Management
Centralized management of clusters and workloads
*New* SUSE Cloud Application Platform on EKS

Quick Starts

- Deploys a highly available VPC architecture in under an hour
- Jointly engineered following the AWS Well-Architected Framework
- Email [aws@suse.com](mailto:aws@suse.com) for more info

## SUSE Cloud Application Platform

### 1
- Containerized Cloud Foundry on SUSE Linux Enterprise
- Stratos-UI 1.0 web console
- HA pod configuration
- Volume Service (NFS)
- MySQL service broker

### 1.1
- Cloud Foundry 2018 certification
- Supported on Azure AKS
- Backup/restore cf plugin
- PostgreSQL service broker
- CF updates - every point release

### 1.2
- Supported on Amazon EKS
- App-AutoScaler
- OSBAPI Broker for Helm: MariaDB, PostgreSQL, Redis, and MongoDB
- Stratos UI v2: service management, Angular updates
- Support for Azure Load Balancer

### 1.3
- CredHub: credential management
- Brokers: Minibroker (Helm), AWS, Azure OSBA
- Stratos UI: Kubernetes endpoint plugin and metrics reporting

### 1.4
- Supported on Google GKE
- Eirini: Kubernetes-native container scheduling (technology preview)
- Support policy for BYO KBS
- CF version updates and 2019 recertification
- CF v3 API features
- DRBD persistent store support

### 1.5
- Eirini: Kubernetes-native application scheduling (Beta feature)
- Stratos: New Helm UI for browsing repositories and deploying charts
- Usage metering via Cloud Foundry Prometheus Exporter
- Deployment automation for CSPs (AKS, EKS, GKE)

### 2 Beta
- All features and updates in 1.5 plus
- CF Quarks: a Kubernetes Operator for Cloud Foundry providing dynamic configuration via BOSH manifests and better lifecycle management for day two operations

### 2.0
- Serverless: Function-as-a-Service in Cloud Foundry and Kubernetes (e.g. via knative)
- Istio & Envoy in CF (pending upstream readiness)
- Integrated CI/CD (Concourse, Jenkins, other)
- Git hosting: Github Enterprise, Gitlab, or Gitea
- Online IDE (e.g. Eclipse Che)
- CF App SSO via Oauth2/UAA: Route service for limiting access to CF-hosted applications.
- System log normalization, filtering and aggregation
- Better SNI/certificate management

### 2.1
- Supported on Google GKE
- Eirini: Kubernetes-native container scheduling (Beta feature)
- Stratos: New Helm UI for browsing repositories and deploying charts
- Usage metering via Cloud Foundry Prometheus Exporter
- Deployment automation for CSPs (AKS, EKS, GKE)

### 2.2
- CredHub: credential management
- Brokers: Minibroker (Helm), AWS, Azure OSBA
- Stratos UI: Kubernetes endpoint plugin and metrics reporting

### 2.3
- All features and updates in 1.5 plus
- CF Quarks: a Kubernetes Operator for Cloud Foundry providing dynamic configuration via BOSH manifests and better lifecycle management for day two operations

### 2.4
- Supported on Amazon EKS
- App-AutoScaler
- OSBAPI Broker for Helm: MariaDB, PostgreSQL, Redis, and MongoDB
- Stratos UI v2: service management, Angular updates
- Support for Azure Load Balancer

### 2.5
- Eirini: Kubernetes-native application scheduling (Beta feature)
- Stratos: New Helm UI for browsing repositories and deploying charts
- Usage metering via Cloud Foundry Prometheus Exporter
- Deployment automation for CSPs (AKS, EKS, GKE)

* Information is forward looking and subject to change at any time.
Transform Application Delivery with SUSE

Increase business agility and speed innovation
- Enhance productivity
- Streamline application lifecycle management
- Build rapidly evolvable applications

Lower costs
- Save time
- Reduce errors and their impact
- Increase efficiency

Maximize return on investment
- Speed time-to-value
- Avoid vendor lock-in
- Leverage and extend existing investments
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.