Cloud Native Application Delivery

Simplify, Modernize & Accelerate Application Delivery

SUSE Global System Integrators Alliance Team
Rahul Krishna Gupta
Senior Solution Architect
Global System Integrators Alliance Team
SUSE

- Rahulkrishna.gupta@suse.com
- LinkedIn: rkgupta76
1. Modern Application Delivery Market
2. SUSE Application Delivery Solutions
3. SIMPLIFY
4. MODERNIZE
5. ACCELERATE
6. Packaging & Sizing
7. The SUSE Advantage
SUSE at a Glance

- Open Source Community
- Customers & Partners

- 10K+ Certified Applications
- 5000+ Global Partners
- 150+ Support Engineers
- 20K+ Certified Systems
- 27 Years of Linux Engineering Experience
- 1st Enterprise Linux Distribution
- 2/3+ of the Fortune Global 100 use SUSE Linux Enterprise
- 19% Revenue Growth

The Largest Independent Open Source Company
SUSE: Underpinning Digital Transformation

Application Delivery

Platform as a Service
SUSE Cloud Application Platform

Software-Defined Infrastructure

Container Management
SUSE CaaS Platform

Compute
Virtual Machine & Container

Storage
SUSE Enterprise Storage

Networking
SDN and NFV

Multimodal Operating System
SUSE Linux Enterprise Server

Physical Infrastructure: Multi-platform Servers, Switches, Storage

Open, Secure, Proven
Working With Our Family

Customers

Partners

Communities

- FIDUCIA GAD
- Daimler
- SAP
- London Stock Exchange
- Office Depot
- ADP
- Mizuho
- MERCK
- rackspace
- SHOpPERS DRUG MART
- UNiSYS
- Walgreens Boots Alliance
- teradata
- Ericsson
- Deutsche Bank
- HSBC
- Fannie Mae

- Alibaba Cloud
- OPEN FOUNDRY
- IBM
- HCL
- Micro Focus
- Lenovo
- Wipro
- TATA
- Intel
- Microsoft
- AWS
- NVIDIA
- Open Mainframe Project
- VMware
- IBM
- Hybrid Cloud

- openATTIC
- Ceph
- Linux
- CLOUD FOUNDARY
- DPDK
- Kubernetes
- Cloud Native Computing Foundation
- Linux
- openstack
- Open Source
- Xen Project
- OpenPOWER
- DLF Networking
Modern Application Delivery Market
Modern Application Delivery market

**Application Container market**
- Container Management & Orchestration
- Monitoring & Logging
- DevOps, CI/CD
- Data Management & Services
- Security
- Networking
- Storage

**Management Services**

**PaaS**

Source: 451 research, Gartner
SUSE's Current Focus

Modern Application Delivery market

Application Container market

- Container Management & Orchestration
- Monitoring & Logging
- DevOps, CI/CD
- Data Management & Services
- Security
- Networking
- Storage

PaaS

However, all vendors in this market offer functionalities to cater to adjacent market segments

Source: 451 research, Gartner
Application Development & Deployment Flow

Developer

Pushes Code

Triggers Build

Build Pipeline

Download Source
Build
Unit Testing
Security Analysis
Dependency check
Build container image
Push image to Registry

Container Registry (Dev)
Application Development & Deployment Flow

Developer

Pushes Code

Triggers Build

Download Source
Build
Unit Testing
Security Analysis
Dependency check
Build container image
Push image to Registry

Build Pipeline

Container Registry (Dev)

Build Pipeline

Triggers Deployment

Release Pipeline

Container Registry (Prod)

Dev → QA → Staging → Prod

Application Environment
Flow encapsulated into Stages

Communication, Collaboration and Security

Dev

Build

Code

Plan

Release

Deploy

Operate

Monitor

Ops

Continuous Integration (CI)

Continuous Deployment (CD)

Continuous Delivery (CD)

Continuous Feedback (CF)
Flow encapsulated into Stages

- Source Code management
- Database management
- Image build
- Repository/Artefact Management

- Code Development & Review
- Container Orchestration
  - Infrastructure management
  - Deployment

- Code
- Build
- Test
- Testing

- Plan
- Release

- Deploy
- Operate
- Monitor
- Monitoring
- Logging
- Tracing
Common tools for these stages
Scope of Container Management Platform

Source: Gartner research
Scope of PaaS

PaaS enables Application Delivery
The market players

Local GEO specific competitors exist
SUSE Application Delivery Solutions

SIMPLIFY… MODERNIZE… ACCELERATE…
Application Delivery Transformation Journey

Containerize
Containerize & 
Orchestrate
Streamline application development and delivery

Modernize
Microservices & 
DevOps
Refactor existing applications

Formalize
Best practices, 
patterns, processes
Automate DevOps workflow

Standardize
Factory-scale production
Scale with efficiency and control

Build and deliver new cloud native applications
Application Delivery Transformation Roadmap

**Containerize**
- **Containerize & Orchestrate**

**Dev**
- Containerize existing apps

**Application Ops**
- Run containerized ISV apps
- Deploy existing internally developed apps/updates more frequently

**Platform Ops**
- Provide modern Kubernetes infra to internal Dev & Ops teams
- Move existing apps to new or different platforms

**Modernize**
- **Microservices & DevOps**

**Dev**
- Build new web apps using 12-factor methodology
- Evaluate existing apps and re-implement using microservices where appropriate

**Application Ops**
- Rollout/rollback new and updated services, including canary deployment
- Implement service mesh
- Monitor and adjust deployment or provide feedback to dev as appropriate

**Platform Ops**
- Secure and maintain container registries and images?

**Formalize**
- **Best practices, patterns, processes**

**Dev**
- Build a library of re-usable microservices & associated connectivity

**Application Ops**
- Create default deployment artifacts
- Automate deployment pipeline
- Integrate into CI/CD pipeline
- Create operators for complex apps

**Platform Ops**
- Implement security policies
- Standardize service mesh policies

**Standardize**
- **Factory-scale production**

**Dev**
- Focuses on development
- Uses company-wide standards and processes

**Application Ops**
- Standardize deployment processes and tooling across teams

**Platform Ops**
- Manage multiple clusters
- Manage multiple tenants
- Manage across multiple clouds
Kubernetes Underpins Modern Application Delivery

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

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

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

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

Kubernetes
Container Management
Cloud Foundry Accelerates Application Delivery

**Containerize**
Containerize & Orchestrate

**Modernize**
Microservices & DevOps

**Formalize**
Best practices, patterns, processes

**Standardize**
Factory-scale production

Streamline application development and delivery

Refactor existing applications

Build and deliver new cloud native applications

Cloud Foundry

Automate DevOps workflow

Scale with efficiency and control

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

Kubernetes
Container Management
SUSE's approach
Modular Kubernetes Solution Architecture
Flexible packaging to match your needs

Kubernetes user experiences

SUSE CaaS Platform
Kubernetes for the Agile Enterprise

SUSE Cloud Application Platform
Cloud Native Productivity at Scale

DIY DevOps Automation

Kubernetes operator experiences

Amazon EKS
Microsoft AKS
Google GKE
SIMPLIFY
Consolidate Operations at Scale*
with unified management and service deployment across complex multi-cluster, multi-cloud, and multi-platform environments

• Manage all of your Kubernetes clusters from a single point of control
• Converge operations of modern application platforms
• Deploy services across managed systems more quickly and easily

* Currently available as Tech Preview
Supported End-points with Stratos
Support for multiple end-points

- Cloud Foundry
- Azure AKS
- Amazon EKS
- GKE
- SUSE CaaS
- Prometheus
- Helm Repository

Kubernetes
Simplify Cluster and Application Management*
with a web-based console that makes it easier to deploy, monitor, scale, and troubleshoot applications, and to manage cluster resources too

• Grow your Kubernetes management team faster
• Quickly grasp situational status or your cluster and running applications

* Available end of January as Tech Preview
SUSE CaaS Platform
Kubernetes for the Agile enterprise

**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

SUSE CaaS Platform
Kubernetes Container Management
SUSE CaaS Platform
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 CaaS Platform 4.1

Now built on SUSE Linux Enterprise Server 15 Service Pack 1

- Latest release of Kubernetes: v1.16
- Simplified Deployment Model – Bare Metal, Virtual Infrastructure, Public Cloud, OpenStack
- Kubernetes update schedule aligned with upstream community (+90 days)
- Uptime! Non-disruptive cluster upgrades
- Enhanced network security policies via Cilium
Container Engine optimized for Kubernetes

cri-o

LIGHTWEIGHT CONTAINER RUNTIME FOR KUBERNETES

DESIGNED
Optimized for Kubernetes

STABLE
Committed to passing Kubernetes tests

ANY IMAGE, ANY REGISTRY
Pull from any compliant registry; run any OCI-compliant container

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
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/
A (Tiny) Cloud Native Web Farm – Standard Kubernetes
Security Requires Pod IPs – Standard Kubernetes
iptables Rules to Enforce App/DB Security

# Allow http protocol access to Apache running in pods [1,2]-1
iptables -i eth0 -p tcp -dport 80 -s ${clientIPRange} -d ${IP1-1}
iptables -i eth0 -p tcp -dport 80 -s ${clientIPRange} -d ${IP2-1}

# Allow Apache to access Vitess/Mysql in pods [1,2]-2

<table>
<thead>
<tr>
<th>iptables</th>
<th>-i eth0 -p tcp -dport 3306 -s ${IP1-1} -d ${IP1-2}</th>
</tr>
</thead>
<tbody>
<tr>
<td>iptables</td>
<td>-i eth0 -p tcp -dport 3306 -s ${IP1-1} -d ${IP1-2}</td>
</tr>
<tr>
<td>iptables</td>
<td>-i eth0 -p tcp -dport 3306 -s ${IP1-1} -d ${IP2-2}</td>
</tr>
<tr>
<td>iptables</td>
<td>-i eth0 -p tcp -dport 3306 -s ${IP2-1} -d ${IP2-2}</td>
</tr>
</tbody>
</table>

# Allow related packets on established or related connections
iptables -m state --state ESTABLISHED,RELATED -j ACCEPT

# Drop all other packets

<table>
<thead>
<tr>
<th>iptables</th>
<th>-i eth0 -j DROP # or -j REJECT</th>
</tr>
</thead>
<tbody>
<tr>
<td>iptables</td>
<td>-o eth0 -j DROP # or -j REJECT</td>
</tr>
</tbody>
</table>
Better!!!...Cilium Label-Aware Security & Visibility

Label-based Security Policy:

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

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

Label-based Security Visibility Logs:

```
23:15:01:   allow: role=frontend \rightarrow role=backend
23:16:34:   deny: role=other \rightarrow role=backend
```

Cloud Native Computing Foundation
Create This ONE Time and Scale Out

endpointSelector:
  matchLabels:
    role = "backend"

ingress:
  matchLabels:
    role = "frontend"

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

```
# Allow Apache to access Vitess/Mysql in pods [1,2]-2

<table>
<thead>
<tr>
<th>iptables</th>
<th>-</th>
<th>eth 0</th>
<th>-</th>
<th>tc p</th>
<th>-dport</th>
<th>330 6</th>
<th>-</th>
<th>${IP1-1}</th>
<th>-</th>
<th>${IP1-2}</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>i</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>s</td>
<td></td>
<td>d</td>
<td></td>
</tr>
<tr>
<td></td>
<td>i</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>s</td>
<td></td>
<td>d</td>
<td></td>
</tr>
<tr>
<td></td>
<td>i</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>s</td>
<td></td>
<td>d</td>
<td></td>
</tr>
<tr>
<td></td>
<td>i</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>s</td>
<td></td>
<td>d</td>
<td></td>
</tr>
</tbody>
</table>
```

Create This ONE Time and Scale Out OR...modify and grow this each time a pod is added
Optimize System Performance
with insights gained from comprehensive, real-time monitoring and powerful data visualization tools

- Replace guesswork with data-driven decisions
- Understand system behavior more quickly and deeply
- Leverage existing skills
Why **SUSE CaaS Platform** is awesome!

Accelerate delivery of modern apps with Kubernetes container management

Simplified Kubernetes administration – exceptional operator experience

Vanilla Kubernetes experience enhances flexibility

Strengthen application security at scale – 1st to support Cilium network plugin
ACCELERATE
SUSE Cloud Application Platform
Cloud Native Developer Experience in Kubernetes

**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

**Standardize**
- Factory-scale production
  - Automate DevOps workflow
  - Scale with efficiency and control

SUSE CaaS Platform
Kubernetes Container Management

SUSE Cloud Application Platform
SUSE Cloud Application Platform
Cloud Native Developer Experience in Kubernetes

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
Boost Developer Productivity

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

- Boost developer productivity
- Developers can serve themselves while staying in compliance with IT guidelines
- Flexibility to use the most appropriate language, framework, and services
Reduce Complexity & Increase IT Efficiency

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

- Easily update language libraries and frameworks
- 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
SUSE Cloud Application Platform

- Cloud Foundry in Kubernetes
- Containerized Cloud Foundry Application Runtime
- Enterprise Ready
- Kubernetes-native scheduler
- Small memory footprint
- 100% open source
- Platinum member of Cloud Foundry Foundation
- SUSE Cloud Application Platform is a certified distribution
- Originator and Project Lead for CF Containerization & Stratos
- Early adopter and influencer of Project Eirini

- Packages Cloud Foundry Application Runtime (CFAR) as containers instead of virtual machines, enabling easy deployment to Kubernetes
- Much smaller CFAR footprint than VM based distribution. (Min 32GB vs 128GB.)

- Web-based UI for managing Cloud Foundry.
- Allows users and administrators to:
  - Manage applications running in the Cloud Foundry cluster
  - Perform cluster management tasks.

- Enables pluggable scheduling for CFAR
- Allows operators to choose whether CFAR should use Diego or Kubernetes to orchestrate application container instances
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 → DB connections → Tests → Network

After

> cf push
SUSE CAP - Public Cloud Deployment

Azure Open Service Broker / Cloud Provider Integration

SUSE Cloud Application Platform

Azure Kubernetes Services

Azure Resources

On Demand Azure Services
- Databases
- Storage
- Security, Vault...
- Messaging, Queuing
- IA, BigData...

Your Application #1
- App Routing
- App Auth
- PaaS Store

Your Application #2

Azure Services used by SUSE CAP
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
- Apply for AWS credits on qualified pilot projects: http://bit.ly/2uxgV2G
- Email aws@suse.com for more info

https://aws.amazon.com/quickstart/architecture/suse-cloud-application-platform/
SUSE Cloud Application Platform Opportunities

Target Use Cases

**Wants to build cloud-native applications**
- May already be using K8s or Cloud Foundry

**Prefers to Buy rather than Build platforms**
- Wants to become productive (reduce time to value)
- Doesn’t have expertise to build and maintain platforms
- Wants to focus expertise elsewhere

**Multi-cloud environments**

**Delivering many cloud native applications**
- Wants factory-scale efficiency
- Values governance and control
Why SUSE Cloud Application Platform is awesome!

$100,000 average cost saving per application development cycle*

10 weeks faster time-to-market per application on average*

Bring your own Kubernetes for maximum flexibility (SUSE CaaSP, Amazon EKS, Azure AKS, Google GKE, 3rd party K8s)

Stratos UI for mult-cloud management

Boost developer productivity

100% open

Fully open source solution

Source: *Cloud Foundry Foundation Application Runtime User Survey, Oct 2018
DISCLAIMER: Some of the features demonstrated are currently in development - they may not make it into product as shown.
Packaging & Sizing
Modular Kubernetes Solution Architecture
Flexible packaging to match your needs

Kubernetes user experiences

DIY DevOps Automation

SUSE Cloud Application Platform
Cloud Native Productivity at Scale

SUSE CaaS Platform
Kubernetes for the Agile Enterprise

Kubernetes operator experiences

Amazon EKS
Microsoft AKS
Google GKE
# How we package our products

<table>
<thead>
<tr>
<th>Available Subscription</th>
<th>Metric used</th>
<th>Entitlements</th>
<th>Target Environment</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SUSE Cloud Application Platform</strong></td>
<td>1 core or 1-2 vCPUs</td>
<td>CAP</td>
<td>For use with 3rd Party Kubernetes</td>
</tr>
<tr>
<td>Cloud Application Platform</td>
<td>1-2 sockets or 1 VM</td>
<td>CaaS, SLES</td>
<td>Physical Low density virtual or Cloud environments</td>
</tr>
<tr>
<td>SUSE CaaS Platform</td>
<td>1-2 sockets with unlimited VMs</td>
<td>CaaS, SLES</td>
<td>High density virtual or Cloud environments</td>
</tr>
<tr>
<td>Migration offering with special conditions</td>
<td>1-2 sockets or 1 VM/with unlimited VMs</td>
<td>CaaS</td>
<td>Migration from existing SLES subscriptions to CaaS</td>
</tr>
<tr>
<td><strong>SUSE CaaS Platform + SUSE Cloud Application Platform</strong></td>
<td>1 core or 1-2 vCPUs</td>
<td>CAP, CaaS</td>
<td>Primarily On-premise</td>
</tr>
</tbody>
</table>
Operating Environments & Units of Measure

The Unit of Measure for SUSE CAP is per 1 Core or 2 vCPUs.

- Each Physical Server or Virtual Machine on which SUSE CAP Container Workloads are deployed, installed, used or executed must have a current SUSE CAP Subscription Offering.
- SUSE CAP Subscription Offerings are Stackable to match or exceed the number of Cores or vCPUs (rounding up to an even number if counting vCPUs).

For Physical Servers, Cores are the unit of measure.
- A Physical Server with 16 Cores needs 16 Subscription Offerings for ‘1 Core or 2 vCPUs’.

For VM & Cloud deployments, vCPU pairs are the unit of measure.
Where the number of Physical Cores cannot be counted directly, vCPU pairs as reported by the hypervisor or cloud provider are counted instead.
- An AWS m4.4xlarge instance type with 16 vCPUs requires 8 Subscription Offerings of ‘1 Core or 2 vCPUs’.
### Available Subscriptions

<table>
<thead>
<tr>
<th>Available Subscription</th>
<th>Metric used</th>
<th>Entitlements</th>
<th>Target Environment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cloud Application Platform (with infrastructure)</td>
<td>1 core or 1-2 vCPUs</td>
<td>CAP, CaaSP</td>
<td>Primarily On-premise</td>
</tr>
<tr>
<td>Cloud Application Platform</td>
<td>1 core or 1-2 vCPUs</td>
<td>CAP</td>
<td>For use with 3rd Party Kubernetes</td>
</tr>
</tbody>
</table>

SUSE Cloud Application Platform Subscription Offerings are available with or without supporting infrastructure entitlements. A "SUSE Cloud Application Platform (with Infrastructure)" Subscription Offering entitles You to use:

(i) SUSE CaaS Platform Kubernetes Worker Nodes, which are counted towards the SUSE CAP Units of Measure.

(ii) SUSE CaaS Platform Kubernetes Master Nodes solely to deploy and execute SUSE CAP Container Workloads, which are not counted towards the SUSE CAP Units of Measure.
# Sizing Considerations

<table>
<thead>
<tr>
<th>Elements of sizing</th>
<th>Key questions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Identify standard VM or hardware cores and memory</td>
<td>• If deployed on bare metal – Number of cores on the node</td>
</tr>
<tr>
<td></td>
<td>• If deployed on VMs – Number of vCPUs for the VMs</td>
</tr>
<tr>
<td></td>
<td>• Memory capacity of the VMs</td>
</tr>
<tr>
<td>Number of Application Instances needed</td>
<td>• Number of application instances to be deployed</td>
</tr>
<tr>
<td></td>
<td>• Types of applications to be deployed</td>
</tr>
<tr>
<td>Determine node utilization</td>
<td>• Space to be reserved to increased demand</td>
</tr>
<tr>
<td>Calculate total memory footprint</td>
<td>• Based on average memory footprint per app</td>
</tr>
</tbody>
</table>

*This is for illustration purposes only. Exact sizing may vary depending on customer specific requirements*
Sizing Example

**Assumptions**

- PaaS in a virtualized environment on-prem
- Using VMs with 16GB of memory and 4 vCPUs
- Number of app instances – 400
- 80% node utilization
- 2GB memory used per application instance

*This is for illustration purposes only. Exact sizing may vary depending on customer specific requirements*
## Sizing Example

<table>
<thead>
<tr>
<th>Elements of sizing</th>
<th>Key questions</th>
<th>Environment variables</th>
</tr>
</thead>
</table>
| Identify standard VM or hardware cores and memory | • If deployed on bare metal – Number of cores on the node  
• If deployed on VMs – Number of vCPUs for the VMs  
• Memory capacity of the VMs | • vCPUs – 4  
• Memory – 16 GB |
| Number of Application Instances needed | • Number of application instances to be deployed  
• Types of applications to be deployed | • 400  
• Java apps |
| Determine node utilization | • Space to be reserved to increased demand | • 80% |
| Calculate total memory footprint | • Based on average memory footprint per app | • Average per app – 2 GB |

*This is for illustration purposes only. Exact sizing may vary depending on customer specific requirements*
## Calculations

<table>
<thead>
<tr>
<th>Calculation</th>
<th>Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>Memory capacity per node (VM)</td>
<td>Memory capacity of the VMs * Node Utilization</td>
</tr>
<tr>
<td></td>
<td>16GB * 80% = <strong>12.8GB</strong></td>
</tr>
<tr>
<td>Total memory utilization</td>
<td>Application Instances * Average memory footprint per app</td>
</tr>
<tr>
<td></td>
<td>400 * 2GB = <strong>800GB</strong></td>
</tr>
<tr>
<td>Number of nodes (VM) required to cover utilization</td>
<td>Total memory utilization / Memory capacity per node</td>
</tr>
<tr>
<td></td>
<td>800GB / 12.8GB = <strong>63 nodes</strong></td>
</tr>
<tr>
<td>Total vCPUs required</td>
<td>Number of Nodes required * vCPUs per node (VM)</td>
</tr>
<tr>
<td></td>
<td>63 * 4 = <strong>252 vCPUs</strong></td>
</tr>
<tr>
<td>Number of CAP subscriptions</td>
<td>Total vCPUs required / 2</td>
</tr>
<tr>
<td></td>
<td>252 vCPUs / 2 = <strong>126 CAP Subscription Offerings</strong></td>
</tr>
<tr>
<td>Total price of deployment</td>
<td>Number of CAP Subscription Offerings * Price of Subscription Offering</td>
</tr>
</tbody>
</table>

*This is for illustration purposes only. Exact sizing may vary depending on customer specific requirements.
Why is SUSE's packaging better

Greater flexibility

Faster time to value

Simplified installation and deployment

Supports customers' Multi-cloud ambitions

Reduced operational expenses – use what you need
The SUSE Advantage
Kubernetes DevOps Solution Architecture

Flexible DevOps automation for custom workloads

Fast and easy DevOps automation for cloud native workloads

Do-It-Yourself

SUSE Cloud Application Platform

Kubernetes

SUSE CaaS Platform
Amazon EKS
Azure AKS
Google GKE

Kubernetes user experience

Kubernetes workflow automation tooling

Kubernetes multi-cloud infrastructure

Stratos (UI)
SUSE Products Deliver this Value

Focus of SUSE Cloud Application Platform

Focus of SUSE CaaS Platform

Code → Application → Linux Container → Container Orchestration → Deployed Containers

- Operate
- Monitor
- Secure

✓ Deploy code to production in minutes
✓ Improve operational efficiency
✓ Deliver applications at scale

Deployed Containers

Operate
Monitor
Secure
Where is SUSE winning? 

- Modular architecture
- Reduced lock-in
- Simpler, yet flexible developer experience
- Multi-vendor community
- Native Kubernetes experience
- Cost effective

- Stratos UI for multi-cloud management
- Better networking performance and security
- Lower learning curve
- Faster time to value
- Simplified management – Lower Opex
- Greater flexibility
Summary

- We have 2 great products
- We have highly differentiated positions
- Our competitors have done the initial legwork in this market
- Their customers are looking for alternatives
- Sell SUSE Application Delivery Solutions: The better alternative!
Questions
Watch on-demand at suse.com/events/webinars!
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.