SUSEcon digital '20

# SUP-1093 SUSE Enterprise Storage From Design To Implementation

Martin Weiss
Senior Architect Infrastructure Solutions
Martin.Weiss@SUSE.com

SUSE

# Agenda

1. Quick Introduction
2. Environment Analysis
3. Requirements
4. Architecture / Planning and Sizing
5. Deployment Best Practices
6. Testing
7. Upgrade

# Quick Introduction

# What is SUSE Enterprise Storage?

- Open Source Software Defined Storage based on Ceph
- Scale Out Architecture
- Multiprotocol Access
  - NFS, S3, Swift, iSCSI, SMB, RBD, RADOS, CephFS
- Redundancy
- Replication and Erasure Coding
- Self Healing, Self Managing
- Including Management and Monitoring
- Simple and Fast Deployment
- Runs on all (SLES Certified) Hardware
- Highly Flexible

➡ BYOS – Build Your Own Storage

# Setting Expectations

# Environment Analysis

# Environment Analysis

- Locations / Users / Servers
- Data Centers / Storage Systems
- WAN / LAN Infrastructure
- Addressing, Name Resolution, Time Synchronization
- Virtualization / Server Hardware
- Server Installation / Configuration Solutions
- Identity Stores

# Requirements

# General Requirements For The Solution

- Hardware
  - IHV, partners such as SuperMicro, HPE, Fujitsu, Lenovo, Dell …
  - SLES / SES Certified!
- Software
  - SES Subscriptions (SLES and SLE-HA)
- Knowledge via Sales, Pre-/Post-Sales Consulting, Partner Assistance
  - For architecture and to buy the right hardware
  - For the initial implementation and upgrades
  - For operational assistance
  - Enablement → Know How Transfer / Training / Certification
- Support
  - 24/7 in case of issues
- Maintenance and proactive support (SUSE Select)
  - Scale, Upgrade, Review and Fix

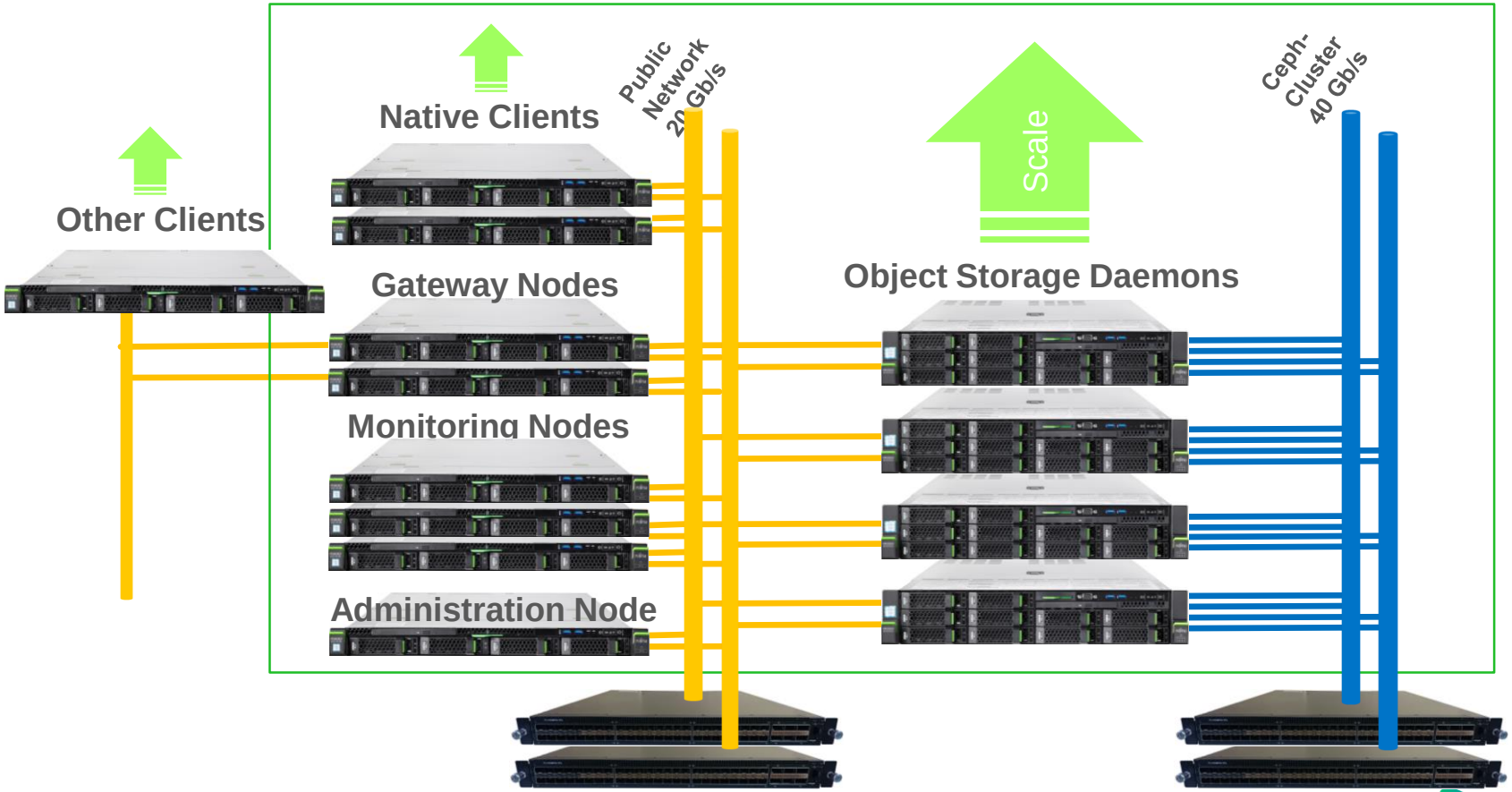
# Use Case Specific Requirements

- Capacity / Data Growth
- Access Protocols: RBD, S3/Swift, iSCSI, CephFS, NFS, SMB
- Availability: Replication Size, Data-Centers
- I/O Workload: Bandwidth, Latency, IOPS, Read vs Write
- Budget
- Integration / Interoperability
- Who are you talking to / working with?

Security, Politics, Religion, Philosophy, Processes ;-)

# Architecture / Planning and Sizing

# Planning and Sizing – Architecture Overview



# Planning and Sizing – Storage Devices

- BlueStore ~~vs. Filestore~~
  - Replication vs. Erasure Coding
- Number of disks = Capacity Requirement * Replication Size + 20% / Size of Disk
  - i.e., 1 PB with 8 TB HDDs and Replication Size 3 = 3,6 PB / 8 TB = 450 HDDs
  - i.e., 200 TB with 8 TB HDDs and Replication Size 3 = 720 TB / 8 TB = 90 HDDs
- Bandwidth Expectations (check white papers)
  - HDD (~150 MB/s, high latency)
  - SSD (~300 MB/s, medium / low latency)
  - NVMe (~2000 MB/s, lowest latency)
- For lower latency (small I/O), use SSD, NVMe for WAL / RocksDB
  - Ratio NVMe vs HDD = 1:12, SSD vs. HDD = 1:4
  - Size of NVMe → 64 GB for DB (no WAL) per OSD

# Planning and Sizing – Network

- Network Technology
  - 10 Gbit/s = ~1 GB/s
  - 25 Gbit/s = ~2,5 GB/s (lower latency)
- Network Bandwidth
  - Cluster Network = 2 * Public Network Bandwidth
  - Due to Size=3 and due to Self-Healing
- Use Bonding (LACP, Layer 3+4)
- Balance number of disks in a server vs. network bandwidth
  - Example :
    20 * 150 MB/s = 3 GB/s total disk bandwidth in a server
    Using replication = 3 with 10 Gbit/s network
    1 GB/s over Public and 2 GB/s over Cluster network
- Switches / VLANs
  - Two switches, not many hops
  - Cluster Network, Public Network, Admin Network, IPMI

# Planning and Sizing – Server

## Admin Server

- Administration via DeepSea (Salt), Monitoring via Grafana and Prometheus
- Optional: Load Balancer to Ceph Dashboard
- Test client for basic performance testing?
- Possibly a VM?

## Object Storage Daemon (OSD) Servers

- YES Certified
- CPU (~1.5 GHz per disk for replication, more for EC)
- Memory for OS plus
  Filestore: 1-2 GB RAM per TB
  BlueStore (1): 2 + 1 GB (HDD), 2 + 3 GB (SSD) per OSD, more for read cache
  BlueStore (2): 2 + 4 GB per OSD, more for read cache
- SSD for OS (RAID 1)
- Fault Tolerance (loosing disks or servers reduces capacity)
- JBOD/HBA and no RAID Controller for OSDs

# Planning and Sizing – Other Services

- Co-Located or Stand-Alone?
- Monitor, Manager, Metadata Server
  - CPU, Memory (Cache), Disk (Monitor)
  - Network (Public)
- RGW, iSCSI, NFS, SMB
  - Additional Network for these Clients
- Load Balancer
  - RGW Scale and Fault Tolerance, SSL Endpoint?
- SLE-HA
  - NFS (failover)
  - SMB (failover and scale)

# Deployment Best Practices

# Deployment – Infrastructure Preparation

- Documentation
- Review the Design
  - Depending on the requirements, adjust before implementation
- Hardware Installation
  - Ensure that hardware installation and cabling is correct
  - Update Firmware
  - Adjust Firmware / BIOS settings
- Disable everything not required (i.e., serial ports, network boot, power saving)
- Configure HW date/time
- Preparation of Time Synchronization
  - Have a fault-tolerant time provider group
- Name Resolution
  - Ensure that all server addresses have different names
  - Add all addresses to DNS with forward and reverse lookup
  - Ensure that DNS is fault tolerant
  - /etc/HOSTNAME must be the name in the public network

# Deployment – Server Installation

- Software Staging
  - Subscription Management Toolkit, SUSE Manager, RMT (limited)
  - Ensure staging of patches to guarantee the same patch level on existing servers and newly installed servers
- General
  - Use BTRFS for the OS
  - Disable Firewall / AppArmor / IPv6
  - Adjust CPU governor to performance
  - CPU mitigations?
- AutoYaST
  - Ensure that all servers are installed 100% identical
  - Consulting solution available (see https://github.com/Martin-Weiss/cif )
- Configuration Management
  - Templates
  - Salt

# Deployment – Infrastructure Verification

- Verify Time Synchronization
- Verify Name Resolution
- Test all Storage Devices
  - HDDs, SSDs, NVMes
  - Bandwidth
  - Latency
- Test all Network Connections
  - Public and Cluster Network
  - Bandwidth
  - Latency

# Deployment – DeepSea

- Configure Salt and Install DeepSea; set deepsea grain
- Adjust reboot, patch and timesync settings (global.yml)
- Execute stage.0 (prepare)
- Execute stage.1 (discovery)
- Create policy.cfg and adjust drive group settings for storage
- Verify and adjust cluster (cluster.yml)
- Adjust gateway configuration (S3 gateway, ports, SSL)
- Execute stage.2 (configure)
- Execute stage.3 (deploy cluster and OSDs)
- Execute stage.4 (deploy gateways)
- Execute stage.5 (optional: delete)

# Deployment – Ceph

- Adjust Crushmap (cluster hierarchy)
- Adjust Crushmap (placement rules)
- Adjust existing pools (rules, number of placement groups)
- Create required additional pools
- Adjust gateway settings
- Verify functionality (Ceph, Dashboard, Prometheus, Grafana, Gateways)

# Testing

# Testing – Preparation

- Create a test plan (based on requirement assessment)
- For every test describe:
  - Starting point (cluster status, cluster usage)
  - Test details
  - Expected result
- When executing the test:
  - Prepare and verify the starting point
  - Execute the test
  - Document the test execution
  - Document the test results
  - Compare the test results with expectations
  - Repeat the test several times

# Testing – Fault Tolerance

- Ensure all fault tolerance tests are done with load on the system
- Network failure (OSD, MON, Gateway)
  - Single NIC / Multiple NIC
  - Single Switch / Multiple Switches
  - Cluster Network / Public Network
- Disk / Server failure
  - Single Disk / Multiple Disks
  - Single Server / Multiple Server / Rack
  - Data-Center
  - Kill one / two MONs
  - Kill one / two Gateways

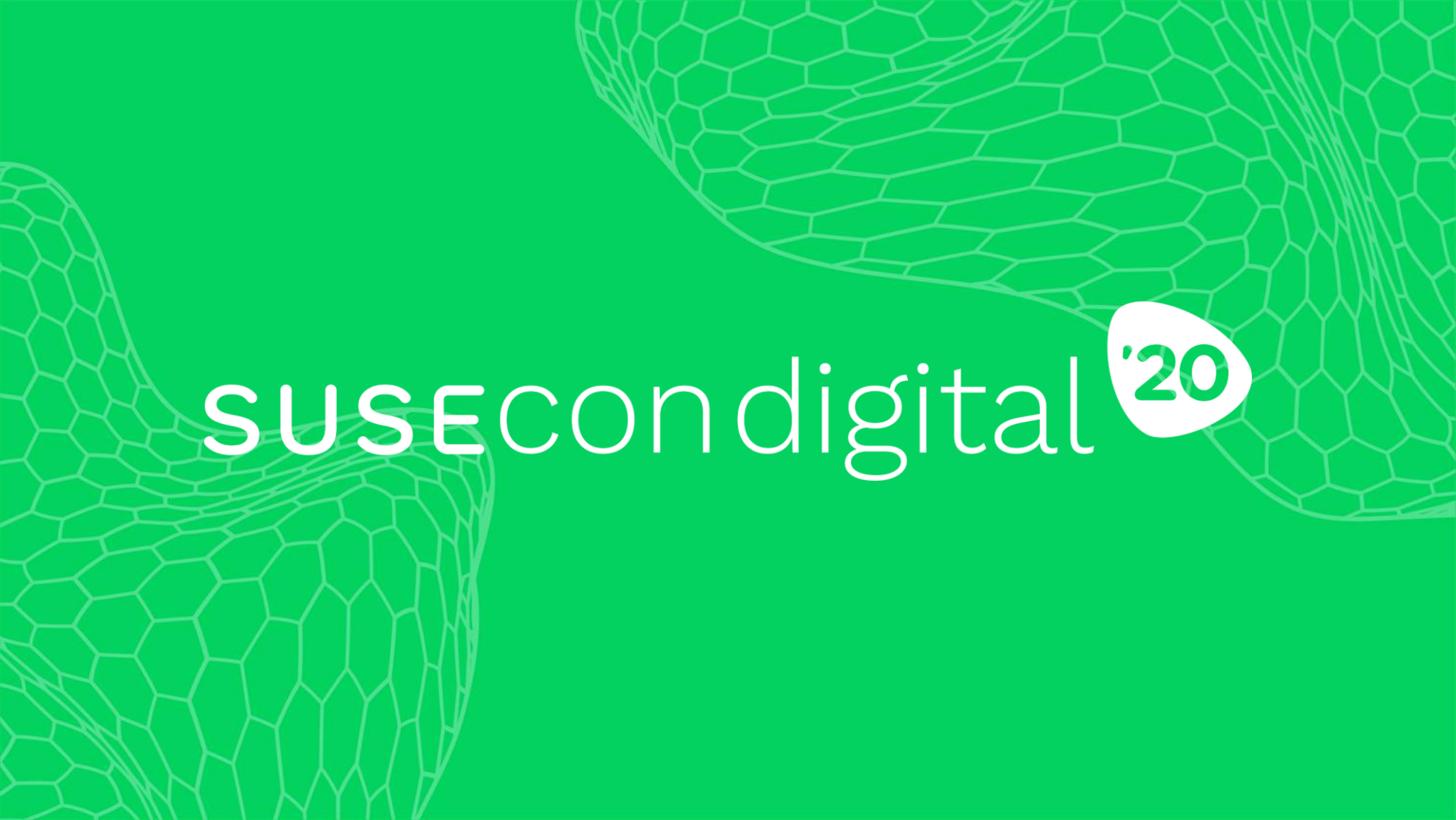
# Testing – Performance

- Create a Baseline
- Bottom Up
- Disk Bandwidth (dd / fio)
- Disk Latency (dd / fio)
- Network Bandwidth (iperf)
- Network Latency (iperf, ping, standard packet size, large packet size)
- Filesystem Layer (optional with filestore)
- OSD Layer (ceph tell osd.* bench)
- OSD layer (ceph osd perf)
- RADOS layer write and read (rados bench write –no-cleanup)
- RBD (map, dd / fio)
- iSCSI (connect, dd / fio)
- CephFS (connect, dd / fio)
- S3 / Swift
- Application


SUSE

## 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 SUSE, LLC, Inc. in the United States and other countries. All third-party trademarks are the property of their respective owners.


SUSEcon digital '20