



Selecting Workloads for Hyperconverged vs Hyperscale Software-Defined Storage

Eric Carter
Hedvig Inc.

- ◆ The material contained in this tutorial is copyrighted by the SNIA unless otherwise noted.
- ◆ Member companies and individual members may use this material in presentations and literature under the following conditions:
 - ◆ Any slide or slides used must be reproduced in their entirety without modification
 - ◆ The SNIA must be acknowledged as the source of any material used in the body of any document containing material from these presentations.
- ◆ This presentation is a project of the SNIA Education Committee.
- ◆ Neither the author nor the presenter is an attorney and nothing in this presentation is intended to be, or should be construed as legal advice or an opinion of counsel. If you need legal advice or a legal opinion please contact your attorney.
- ◆ The information presented herein represents the author's personal opinion and current understanding of the relevant issues involved. The author, the presenter, and the SNIA do not assume any responsibility or liability for damages arising out of any reliance on or use of this information.

NO WARRANTIES, EXPRESS OR IMPLIED. USE AT YOUR OWN RISK.

◆ Selecting workloads for hyperconverged vs hyperscale software-defined storage

- ◆ Debate is rising in the industry over the best method for deploying infrastructure in private and public cloud datacenters. The predictable performance, packaging, and capacity increments of hyperconverged systems have made it the latest model to gain traction. Hyperscale is also gaining momentum as a preferred architecture due to its independent scaling capabilities. Now, businesses are asking, which is the best deployment for software-defined storage? The answer: it depends.

Attendees will learn which deployment is suitable for their workload types – ranging from general purpose server virtualization and VDI to big data and non-virtualized applications. Attendees from companies looking to modernize their IT infrastructure with a goal of being more agile and “cloud-like” will gain insight into whether hyperscale, hyperconverged, or a mixture of both systems provides the right solution to support their storage needs.

- ◆ Learning Objectives
 - › How to distinguish hyperconverged from hyperscale and the advantages of each
 - › How software-defined storage is deployed in each architecture
 - › How organizations get started with software-defined storage

The shift in IT architectures

- Online & social media companies have changed our world forever
- New infrastructure approaches pioneered out of necessity
- The need for speed and lower-cost IT inspired (required) innovation



What they want in infrastructure



Scale out & back
predictably



Run on
commodity hardware



Change and
adapt quickly



Survive hardware
failures



Deliver performance
and efficiency



Simplify and
automate

TODAY'S ENTERPRISES WANT THE SAME!

Why this approach for enterprises?

Enterprises, like web-scalers:



Have unpredictable data growth



Need to lower costs



Want to simplify infrastructure

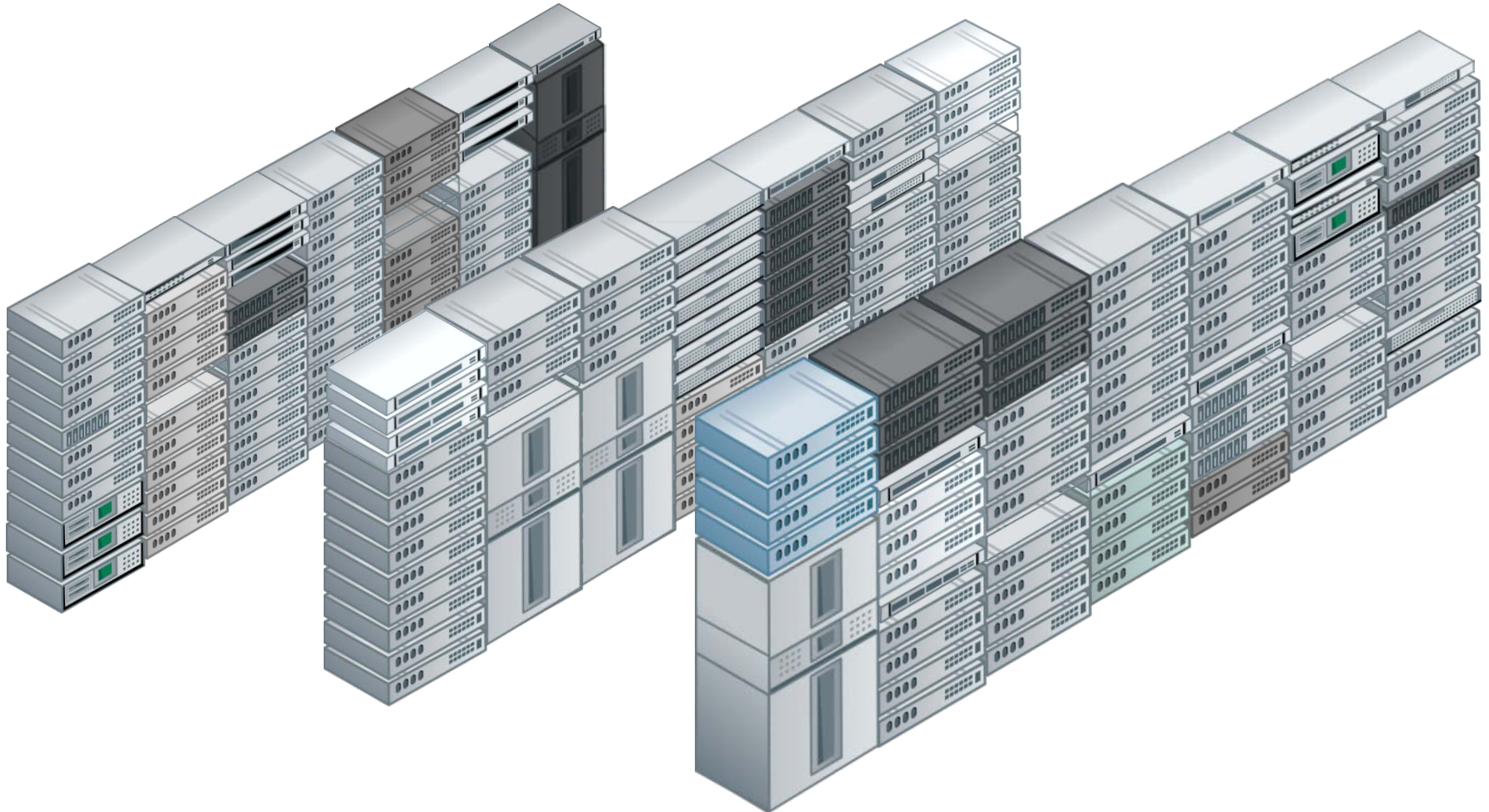


Want to manage IT with fewer personnel

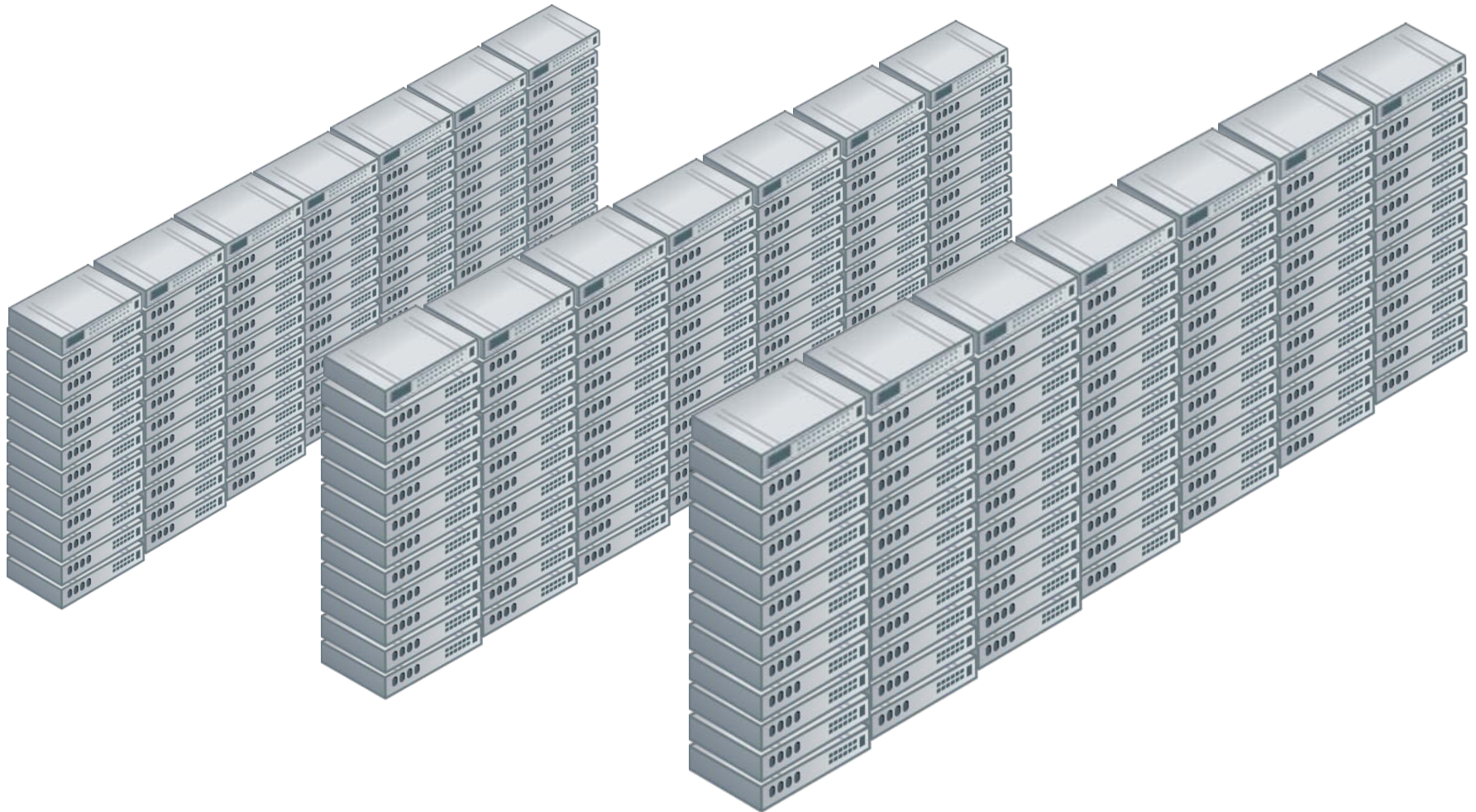


Want to react quickly to business opportunities

From this...



To this. . .



Commodity ≠ Cheap

- **Hardware still matters**
 - ◆ Yes, software-defined is in part hardware-defined!
 - ◆ Deploying the right components will make life easier
- **Systems have become powerful and standardized**
 - ◆ Moore's law has brought us a long way (and more to go)
 - ◆ What's behind your array bezel?
- **There are benefits to moving away from custom hardware**
 - ◆ Cost is one
 - ◆ Availability (easy-to-acquire) is another

What is this “hyper” stuff?

Terminology level-set

➤ Hyperconverged

- ◆ App compute and storage smarts combined on the same “tin”
- ◆ Scale-out, commodity-based building-blocks
- ◆ Software-defined, distributed systems approach

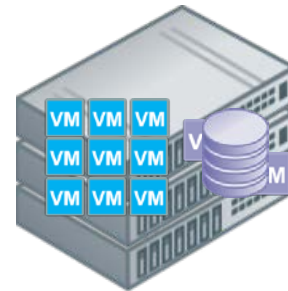
➤ Hyperscale

- ◆ App compute and storage resources separate
- ◆ Scale-out components independently on commodity-hardware
- ◆ Software-defined distributed systems approach

What is this “hyper” stuff?

Hyperconverged

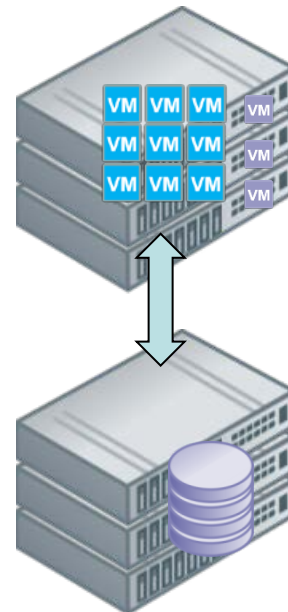
- ◆ Apps and storage utilize same nodes



For storage, controller / storage intelligence run as VMs

Hyperscale

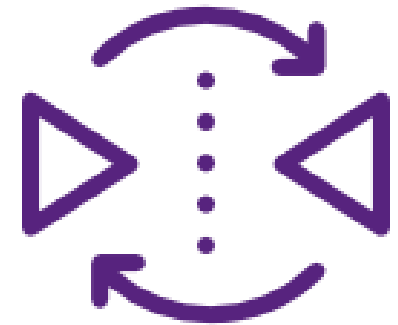
- ◆ Apps and storage utilize distinct nodes



For storage, access VMs decoupled from controller / storage intelligence (typically runs on bare metal)

Things that are common to both

- Both leverage commodity hardware
- Each pool direct attached storage across cluster and make it available to hosts
- Each take advantage of flash/SSDs to drive IOPS performance
- Each distribute data across nodes for availability



Hyperconverged



❖ Criticism:

- ◆ I'm forced to scale compute and storage scale in lockstep
- ◆ The hypervisor adds latency to storage
- ◆ It's expensive
- ◆ There are limits to scaling



❖ Praise

- ◆ It's easy to deploy
- ◆ It's easy to manage
- ◆ It's easy to expand and provides linear scaling
- ◆ It provides data locality (short/no hop)

Hyperscale



❖ Criticism:

- ◆ “Roll-your-own” deployment adds complexity
- ◆ It’s only for big companies
- ◆ It forces me to manage more things in my environment
- ◆ It means I have at least one hop to get my data



❖ Praise

- ◆ It gives me freedom of choice for hardware
- ◆ It lets me scale what I need, when I need
- ◆ It’s easy to expand and provides linear scaling (and I can grow really, really big)

What's the right approach for you?

◆ Questions to ask:

- ◆ Are your apps predictable?
- ◆ Do you favor simplicity over flexibility?
- ◆ Do you need to support bare-metal or containerized apps in addition to VMs?
- ◆ How large do you need to scale?
- ◆ Are you building a cloud-like infrastructure?

Choices and workloads

	VDI	ROBO	General purpose virtualization	Mixed hypervisor / compute	Cloud architectures (e.g. OpenStack, Docker)	Big data / analytics	OLTP
Hyperconverged	●	●	●	○	◐	◐	◐
Hyperscale	◐	○	◐	●	●	●	●

- Best overall fit
- ◐ Good for most
- ◑ Good for specialized deployments
- ◒ Works but not an optimal choice
- Not a good fit

Another set of decision criteria

Selection criteria	Hyperconverge if . . .	Hyperscale if . . .
# of employees	0 to 2,500	5,000 or more
# of VMs	0 to 500	500 or more
# of apps	0 to 250	250 or more
# of total TBs	0 to 250	500 or more
# of total storage admins	0 to 5	5 or more

Can I do both?

Yes! – You have two choices:

- **Deploy individual solutions**
 - ◆ Comes with the overhead of managing two islands
- **Choose a solution that can support both architectures**
 - ◆ Will typically be a software-only solution – BYOH
 - ◆ Enables management as one logical system
 - ◆ Mix/match “nodes” as appliances are starting to emerge

➤ If you've not delved into this world



- ◆ Do some reading – lots of guides, videos, etc. out there
- ◆ Take a few meetings, hear the pitch, ask questions
- ◆ PoC? Vendors are anxious to gain traction and will help you get started

➤ If you've already been there



- ◆ Assess – how has the journey been?
- ◆ Have you hit any bottlenecks (or walls?)
- ◆ Re-assess landscape – LOTS of new choices as of 2015

The SNIA Education Committee thanks the following Individuals for their contributions to this Tutorial.

Authorship History

Eric Carter / May 6, 2016

Updates: May 25, 2016

Additional Contributors

Please send any questions or comments regarding this SNIA Tutorial to tracktutorials@snia.org