An Experiment on Bare-Metal BigData Provisioning

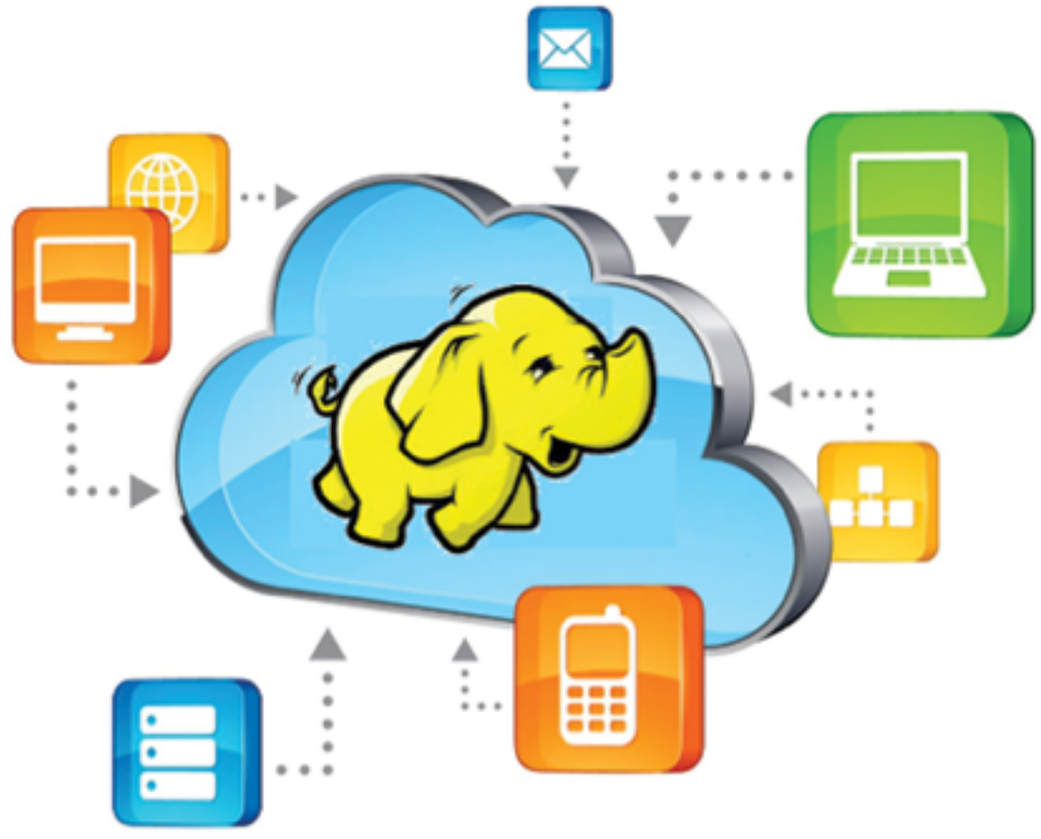
Ata Turk, Ravi S. Gudimetla, Emine Ugur Kaynar, Jason Hennessey,
Sahil Tikale, Peter Desnoyers, Orran Krieger

BOSTON UNIVERSITY

Northeastern

BigData Analytics on the Cloud

- BigData deployments are moving to the cloud
  - On-demand usage (Cost), Elasticity, Agility, Simplicity, …
  - Virtualized IaaS solutions: Amazon EMR, Azure HDInsight, …
- Virtualization drawbacks
  - Overhead, unpredictability, security concerns, device functionality, …
  - Bare-metal cloud solutions: IBM, Rackspace, and Internap, …



Bare-Metal BigData Cloud Solutions

- Bare-Metal cloud provisioning
  - Automated provisioning: Ironic, MaaS, …
  - Image copy to local disk => long waits => loss of agility & elasticity
- OS streaming*, Lazy copy & de-virtualization**
- What about network booting?
  - *incur an ongoing unacceptable overhead during runtime*

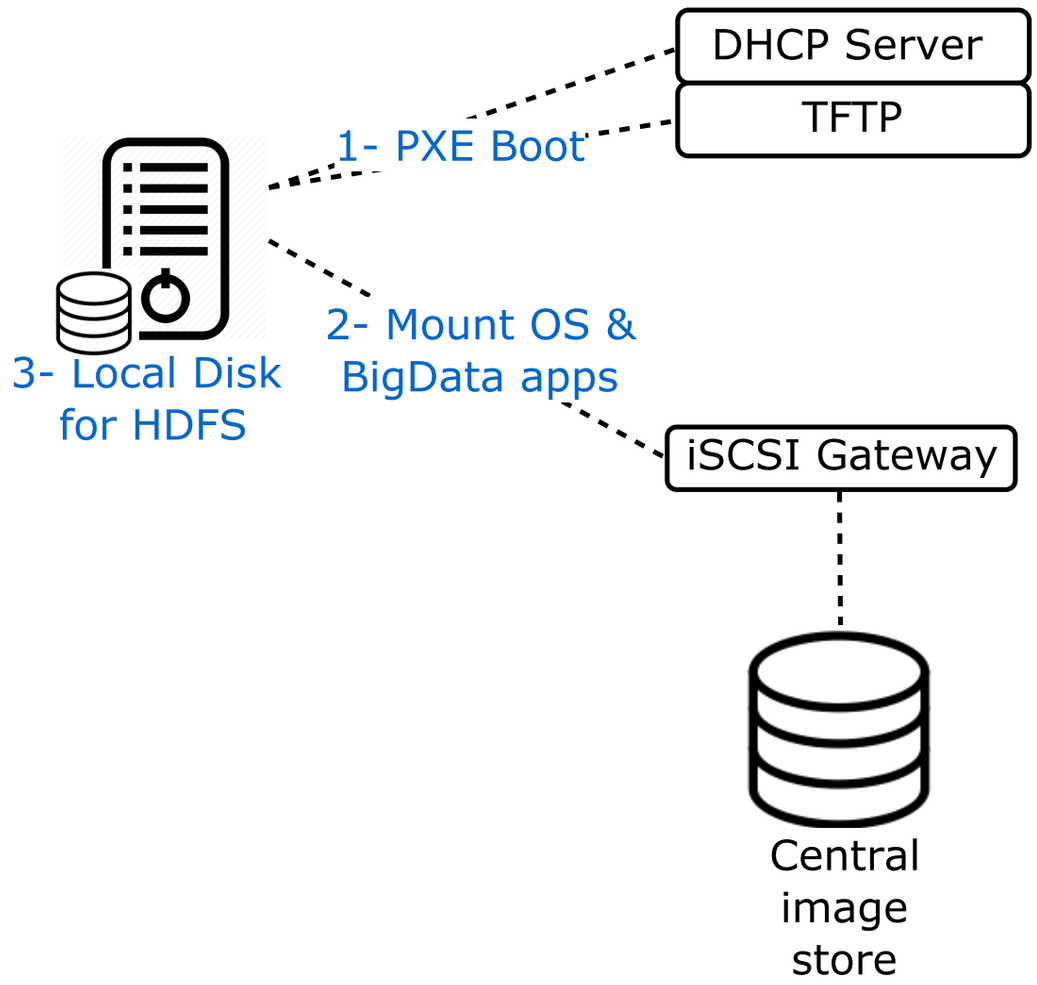
* David Clerc, “OS Streaming Deployment”, in IPCCC’10, *pp. 169–179, 2010.*

** Y. Omote, T. Shinagawa, and K. Kato, “Improving Agility and Elasticity in Bare-metal Clouds,” in ASPLOS*’15, pp. 145–159, 2015.*

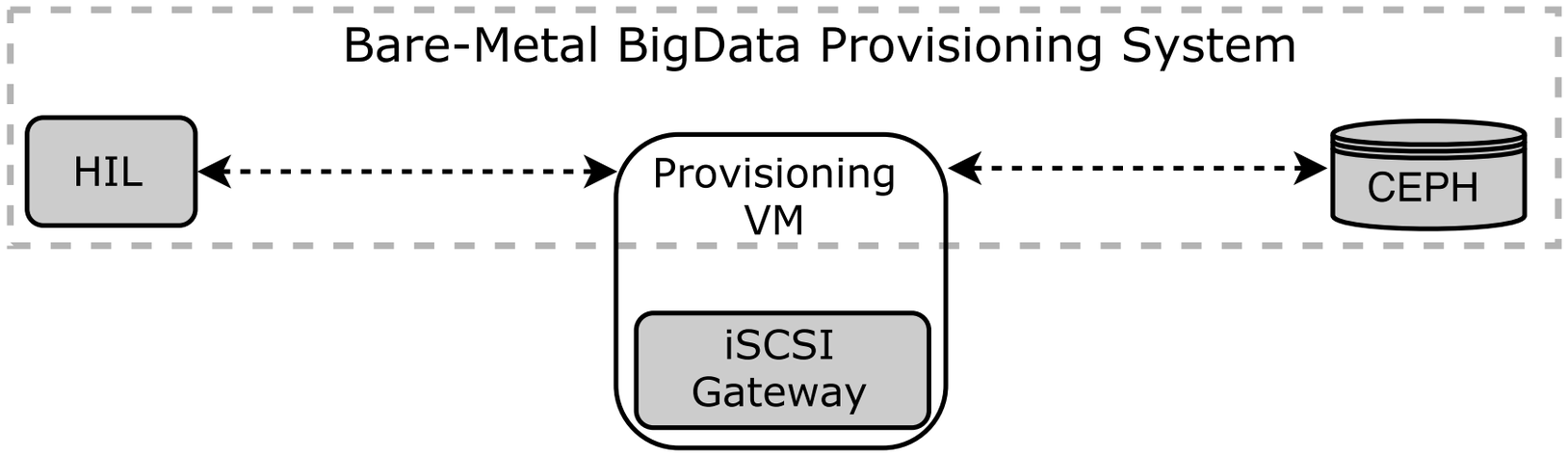
- Large parts of the HPC community has been doing it for the last 20 years.
- Virtualized IaaS is doing it all the time.
- Why not bare-metal cloud?

Network-Mounted BigData System

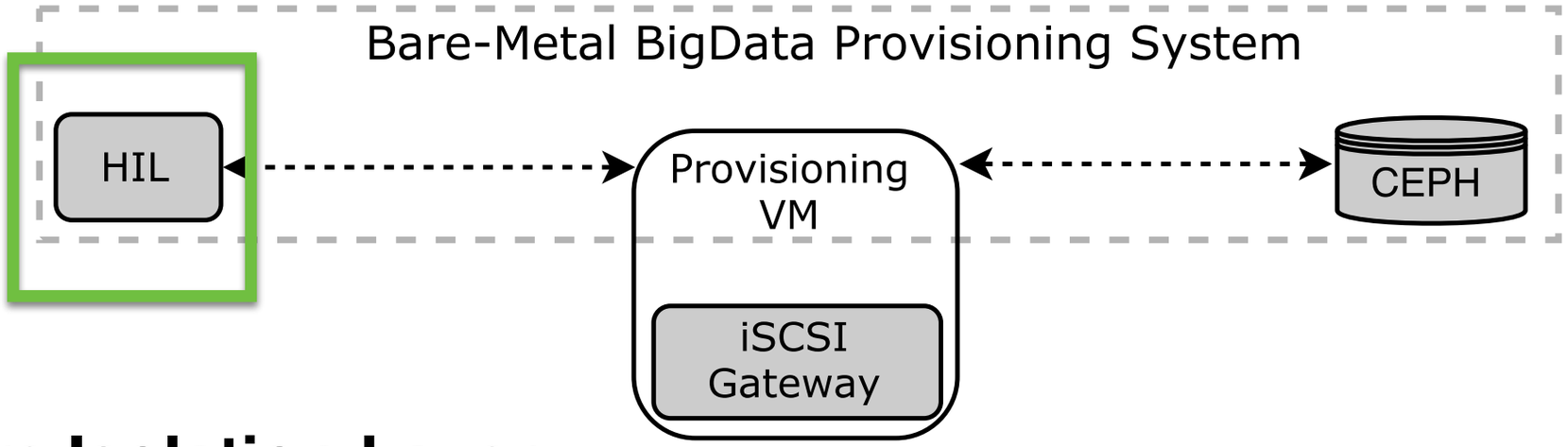
- Clients access kernel and init ramdisk via PXE
- Mount OS & BigData apps from a remote iSCSI volume
- Use local disk for ephemeral storage (HDFS, `/swap`, `/tmp`, …)



Bare-Metal BigData Provisioning Prototype

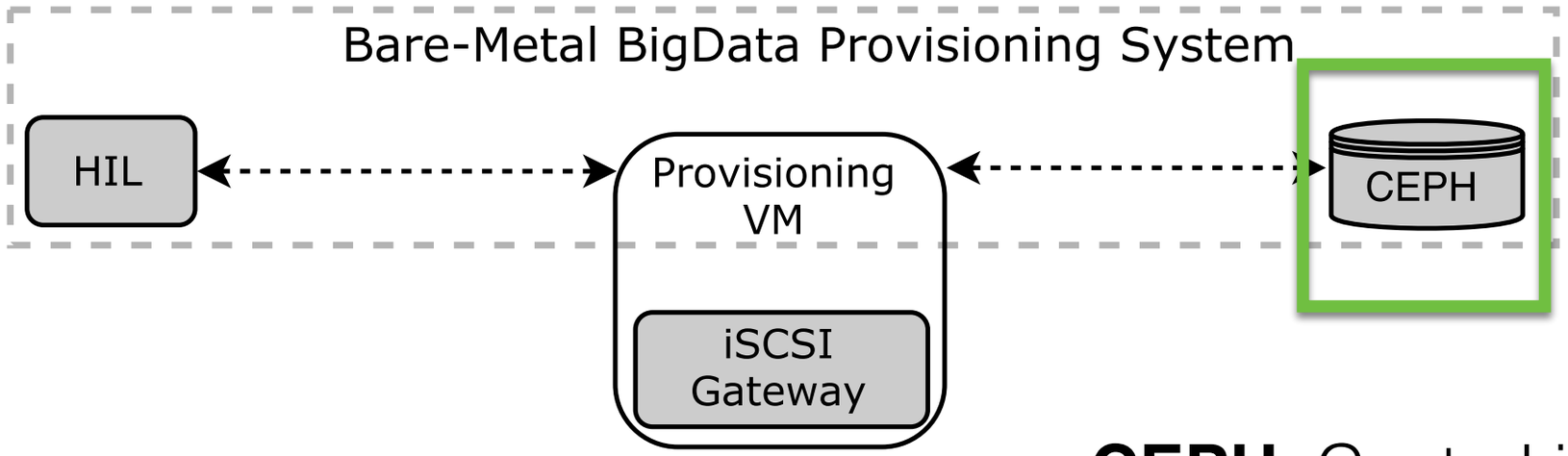


Bare-Metal BigData Provisioning Prototype



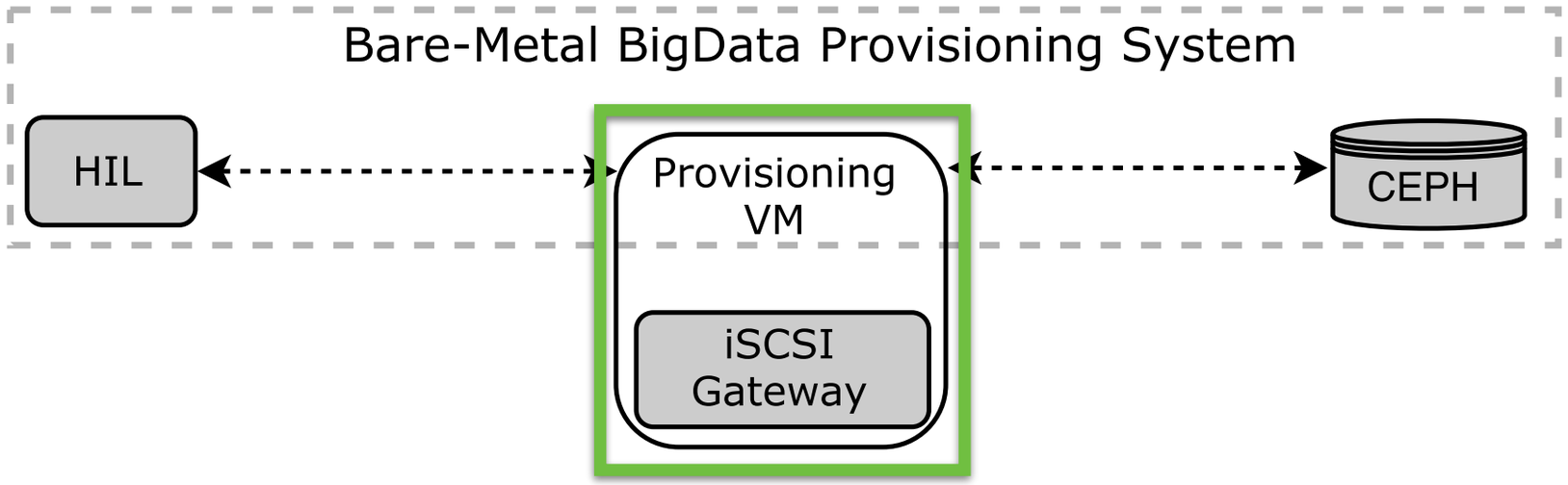
Hardware Isolation Layer: A service to allocate bare-metal nodes out of a shared pool and isolate network

Bare-Metal BigData Provisioning Prototype



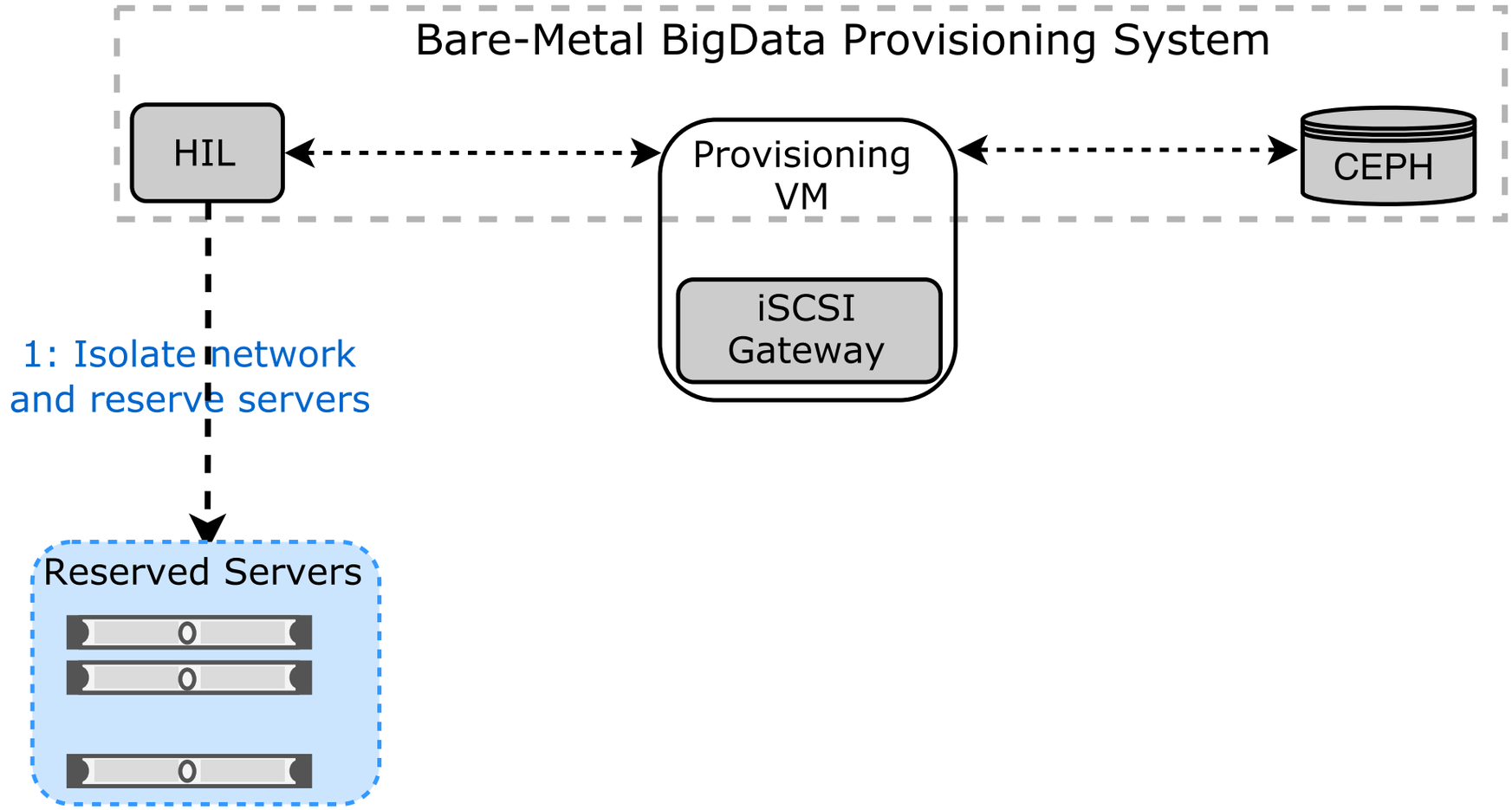
CEPH: Central image store hosting user images with BigData applications

Bare-Metal BigData Provisioning Prototype

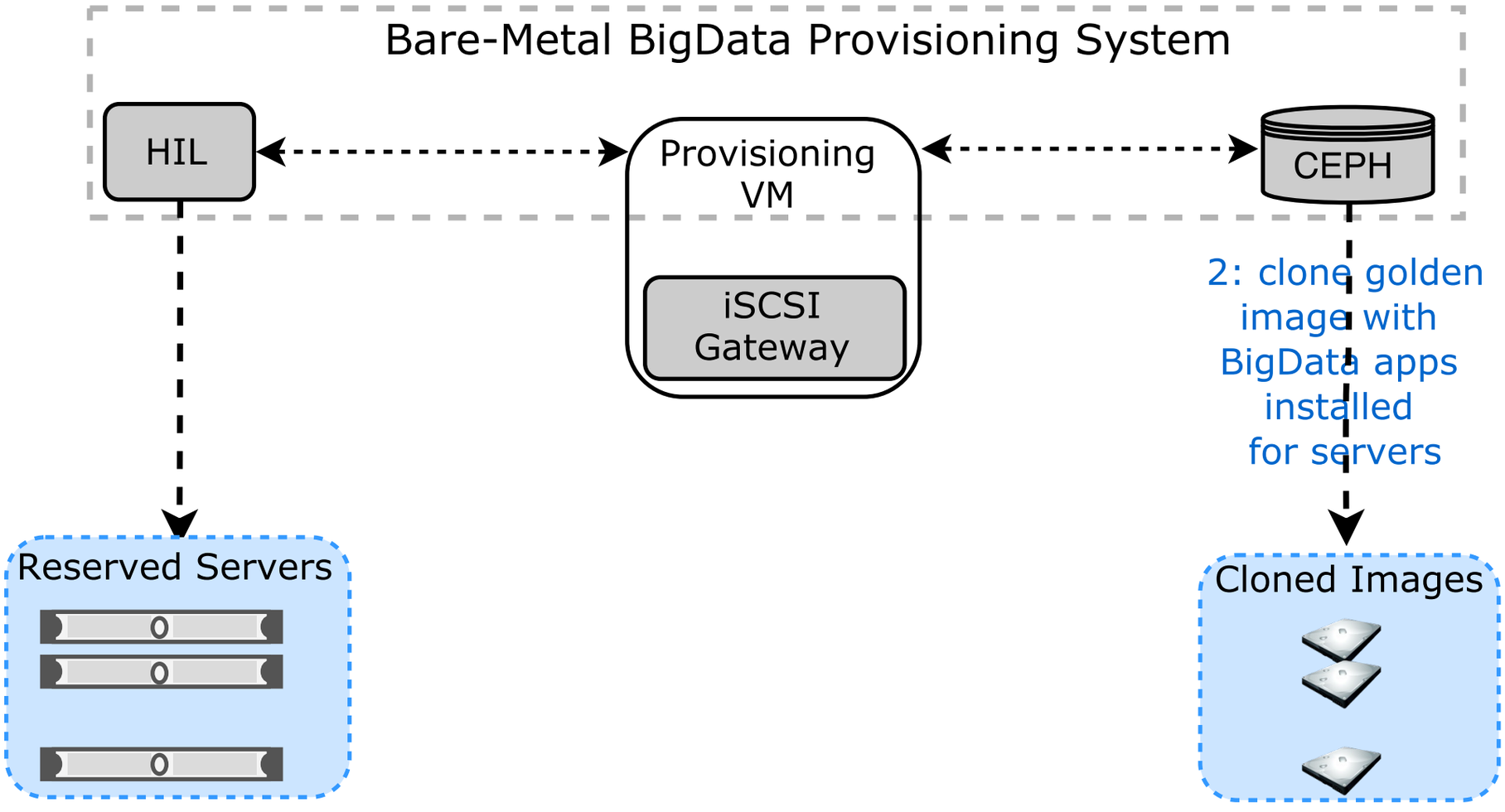


Provisioning VM: Gateway between isolated servers and image store

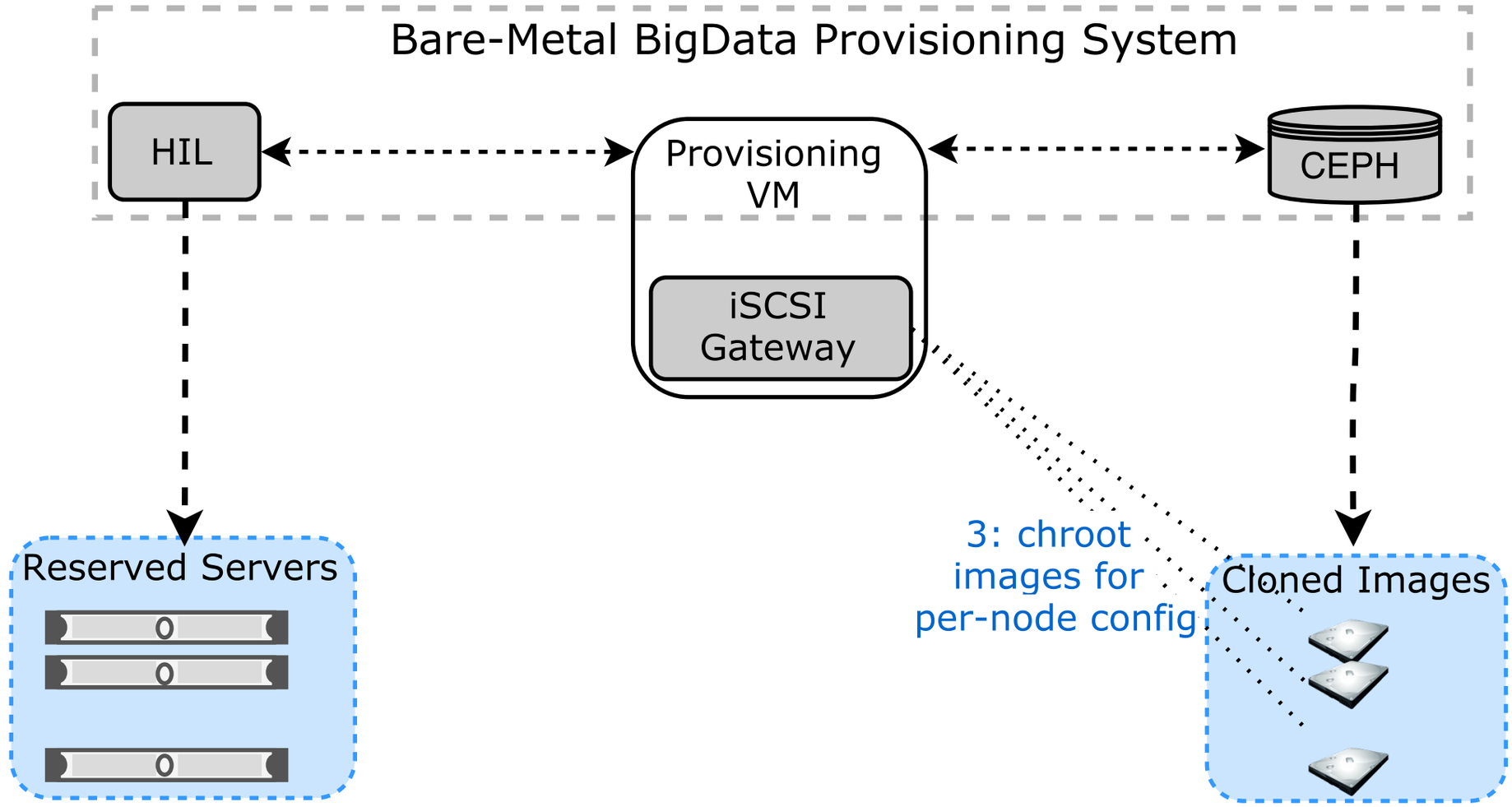
Bare-Metal BigData Provisioning Prototype



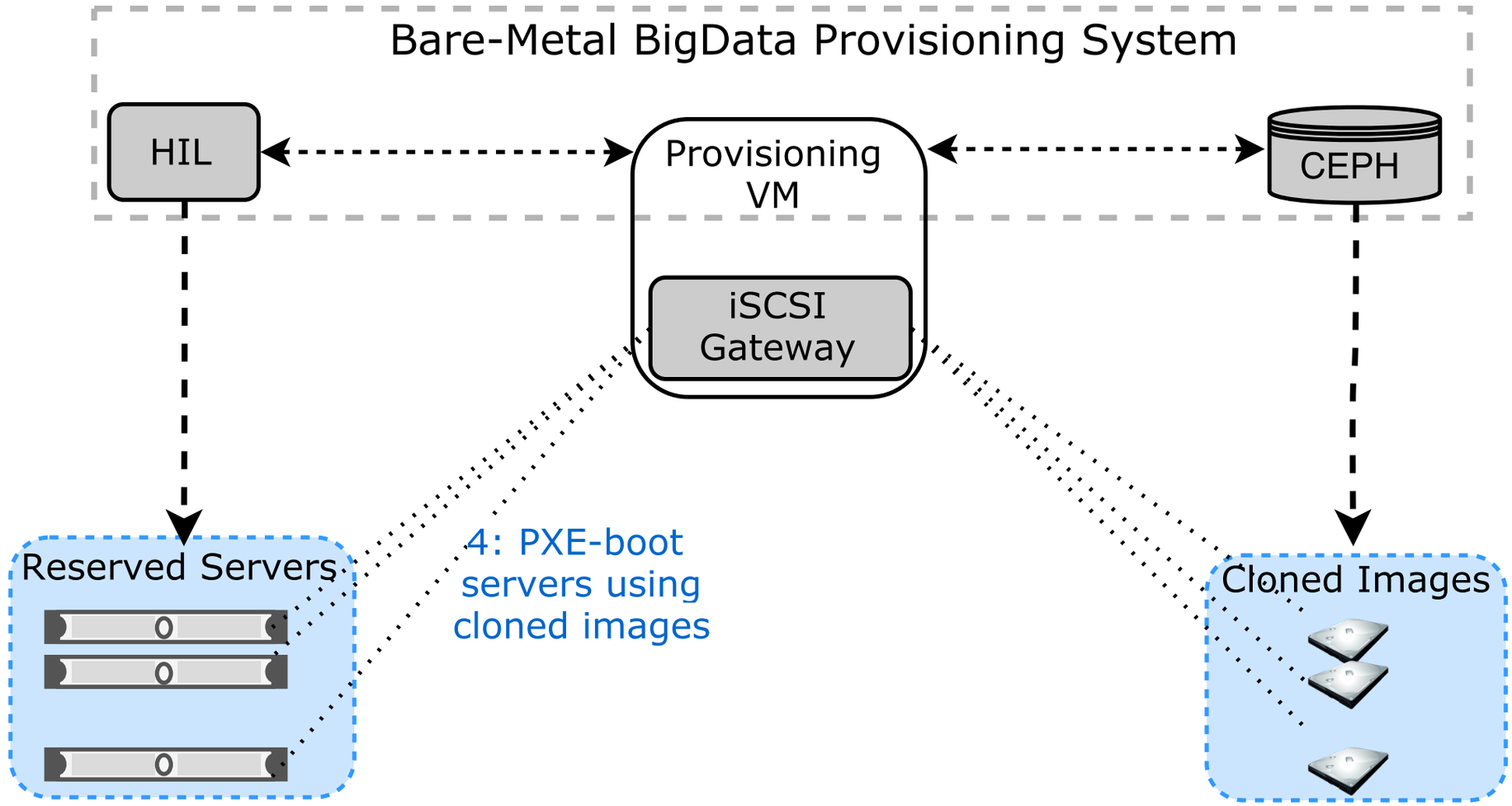
Bare-Metal BigData Provisioning Prototype



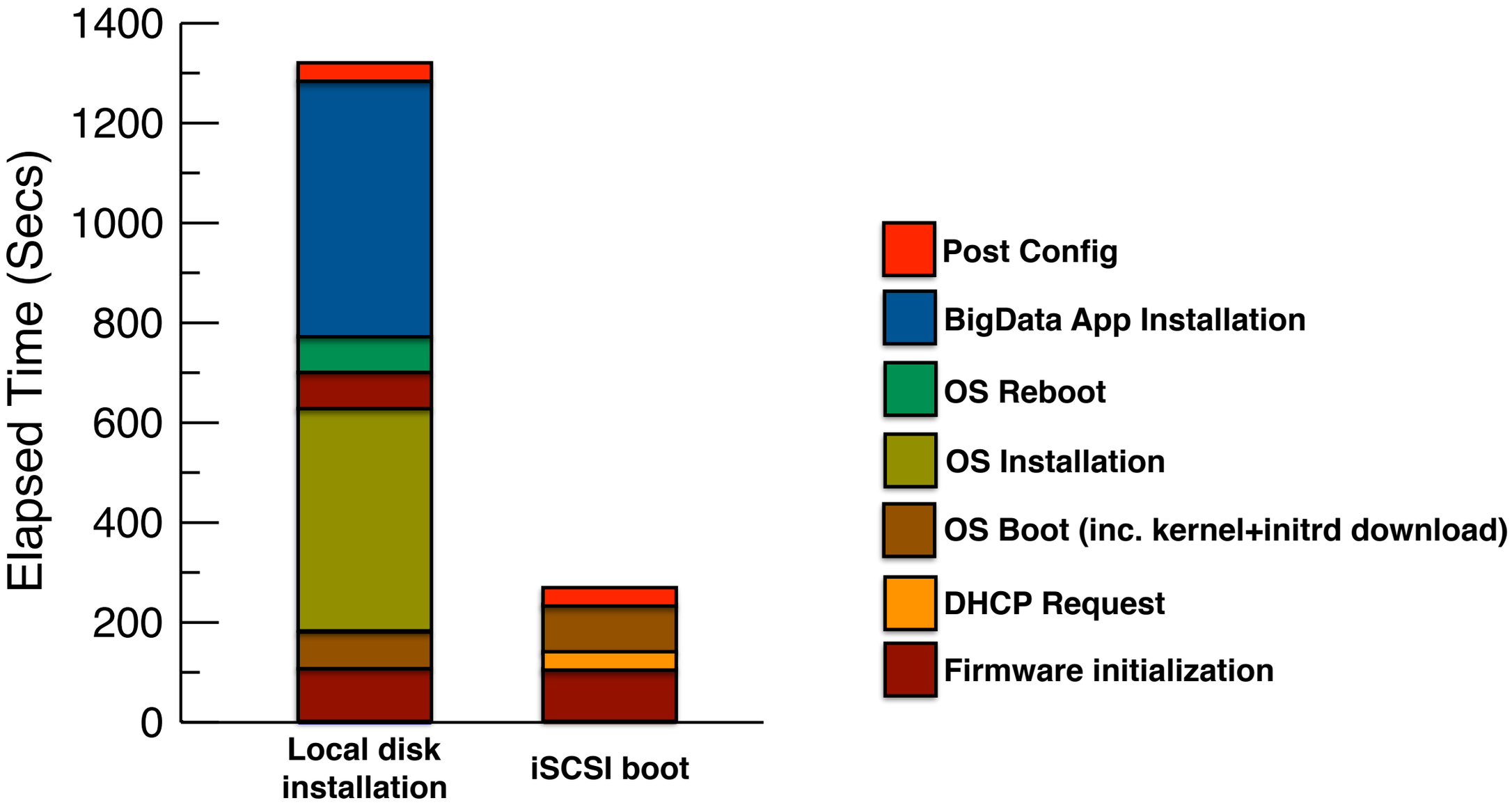
Bare-Metal BigData Provisioning Prototype



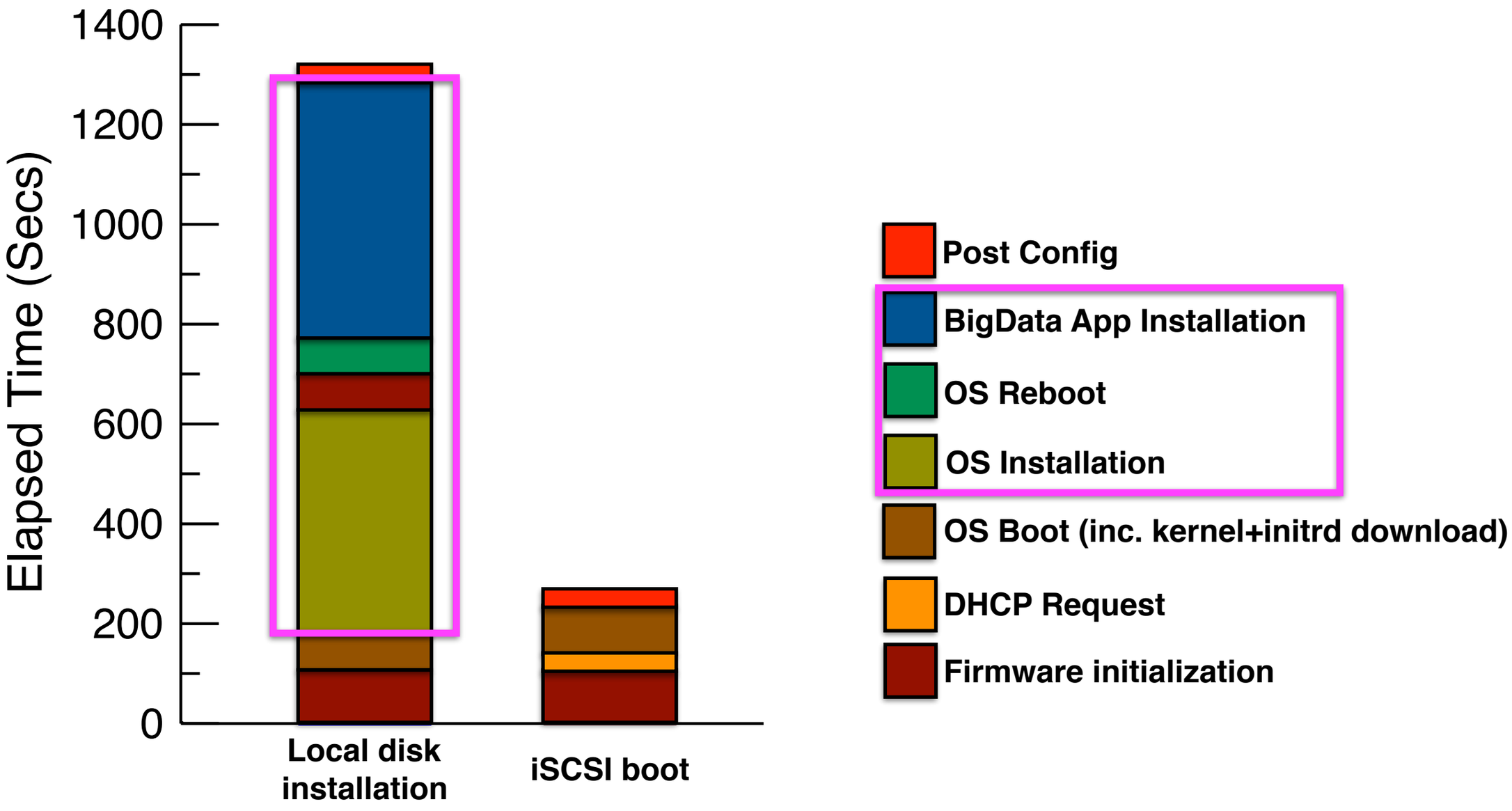
Bare-Metal BigData Provisioning Prototype



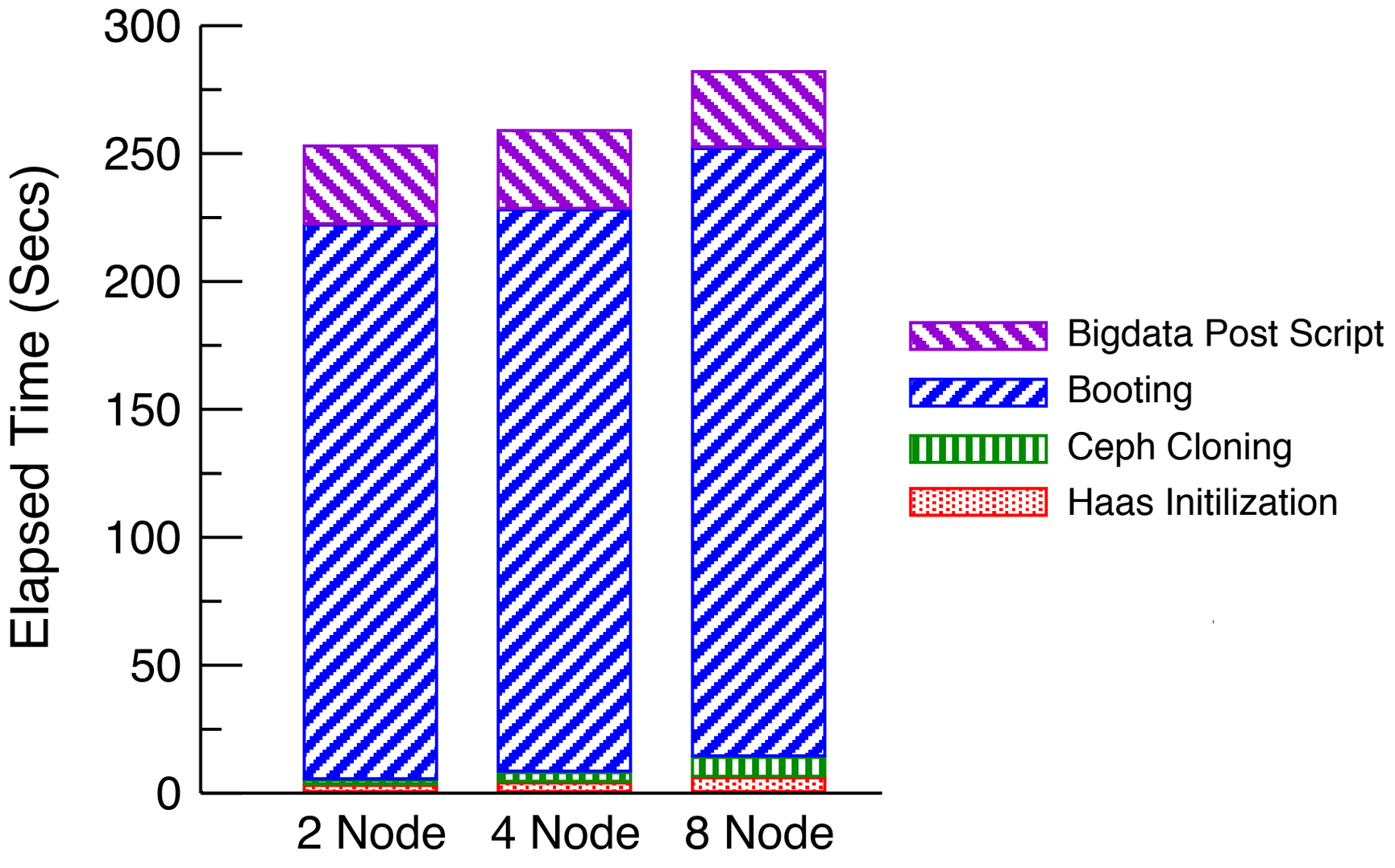
Provisioning Time



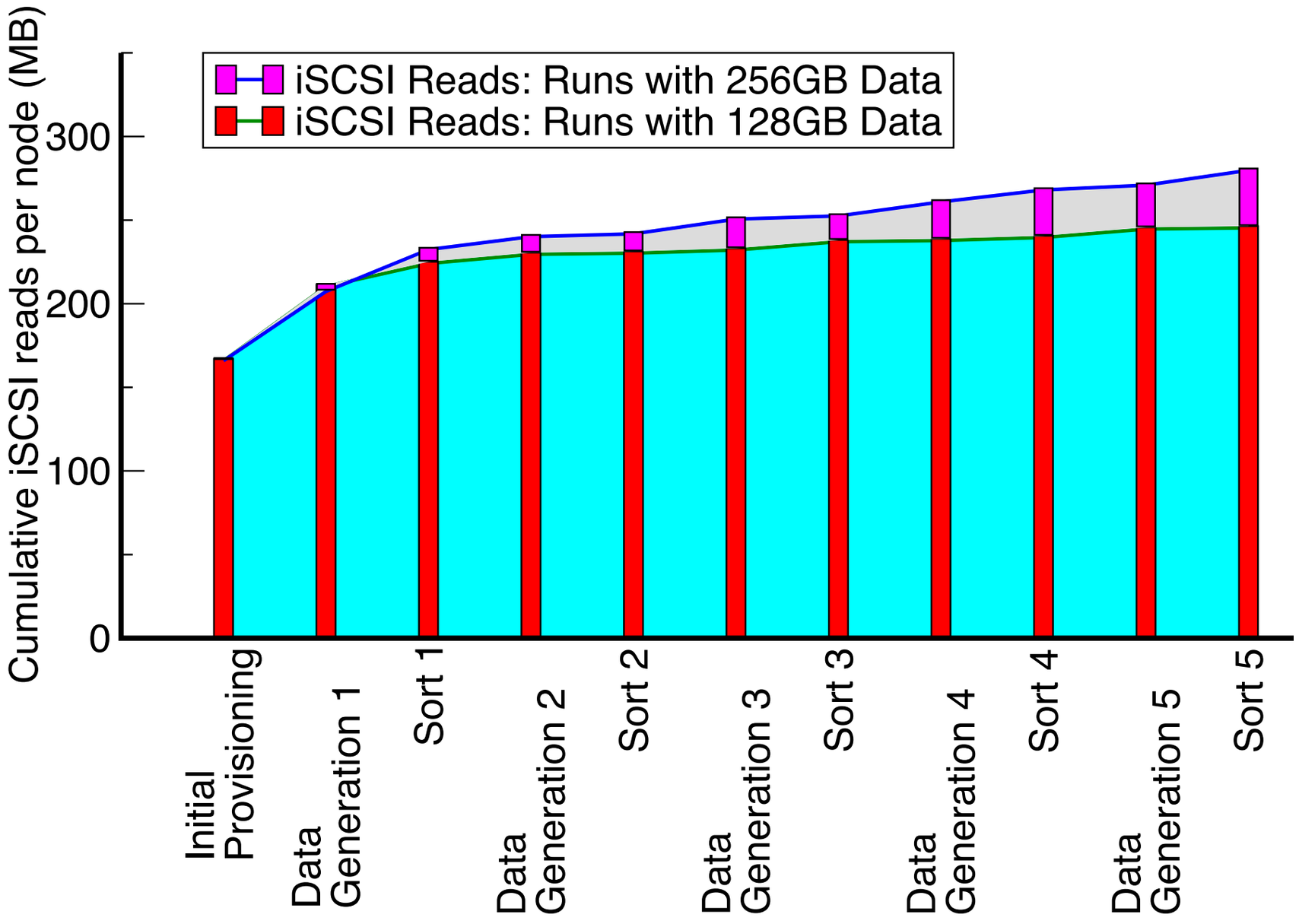
Provisioning Time



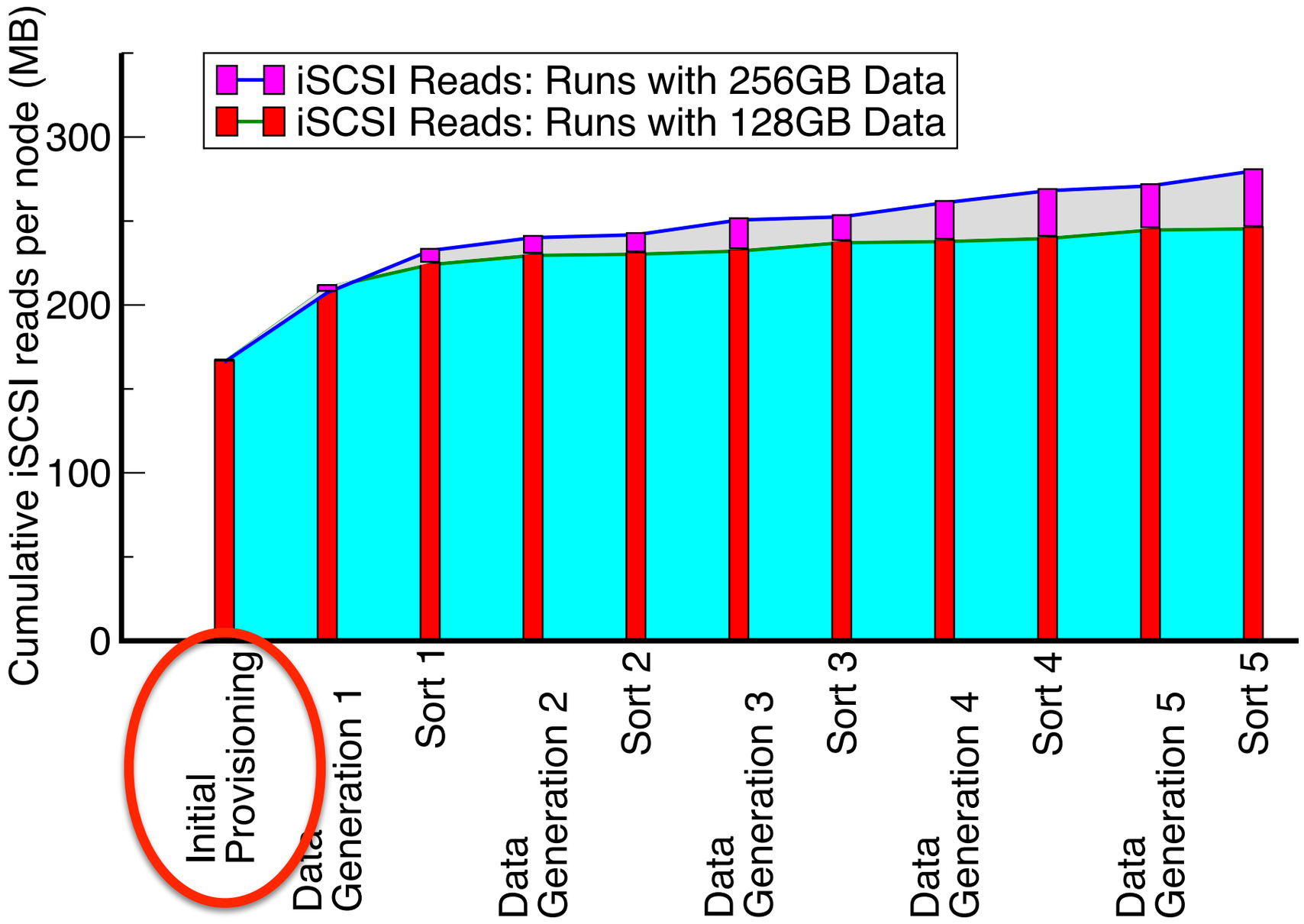
Provisioning Time Scaling



Read Traffic over Boot Drive

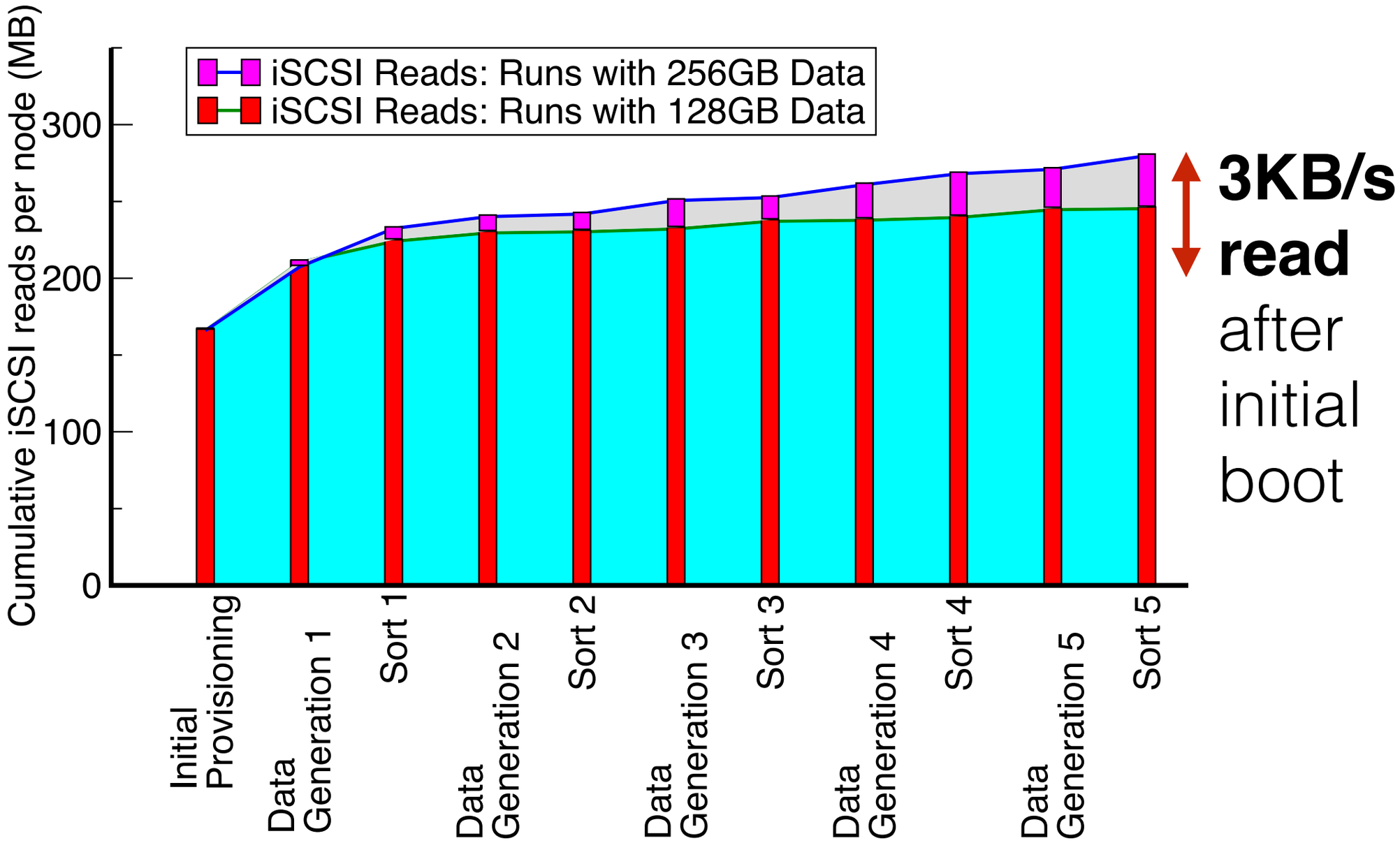


Read Traffic over Boot Drive

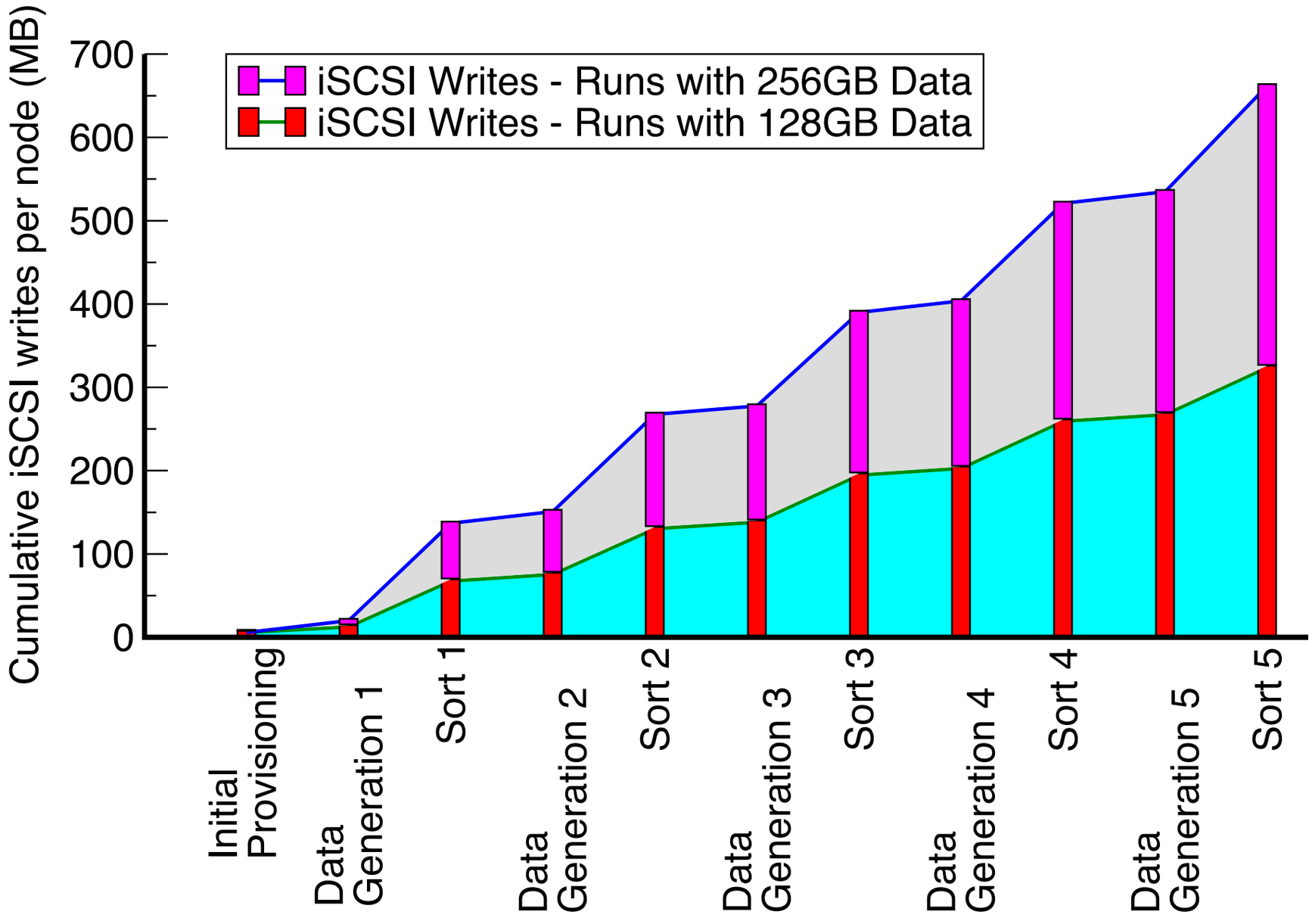


~170MB / 8GB Boot Image => 2%

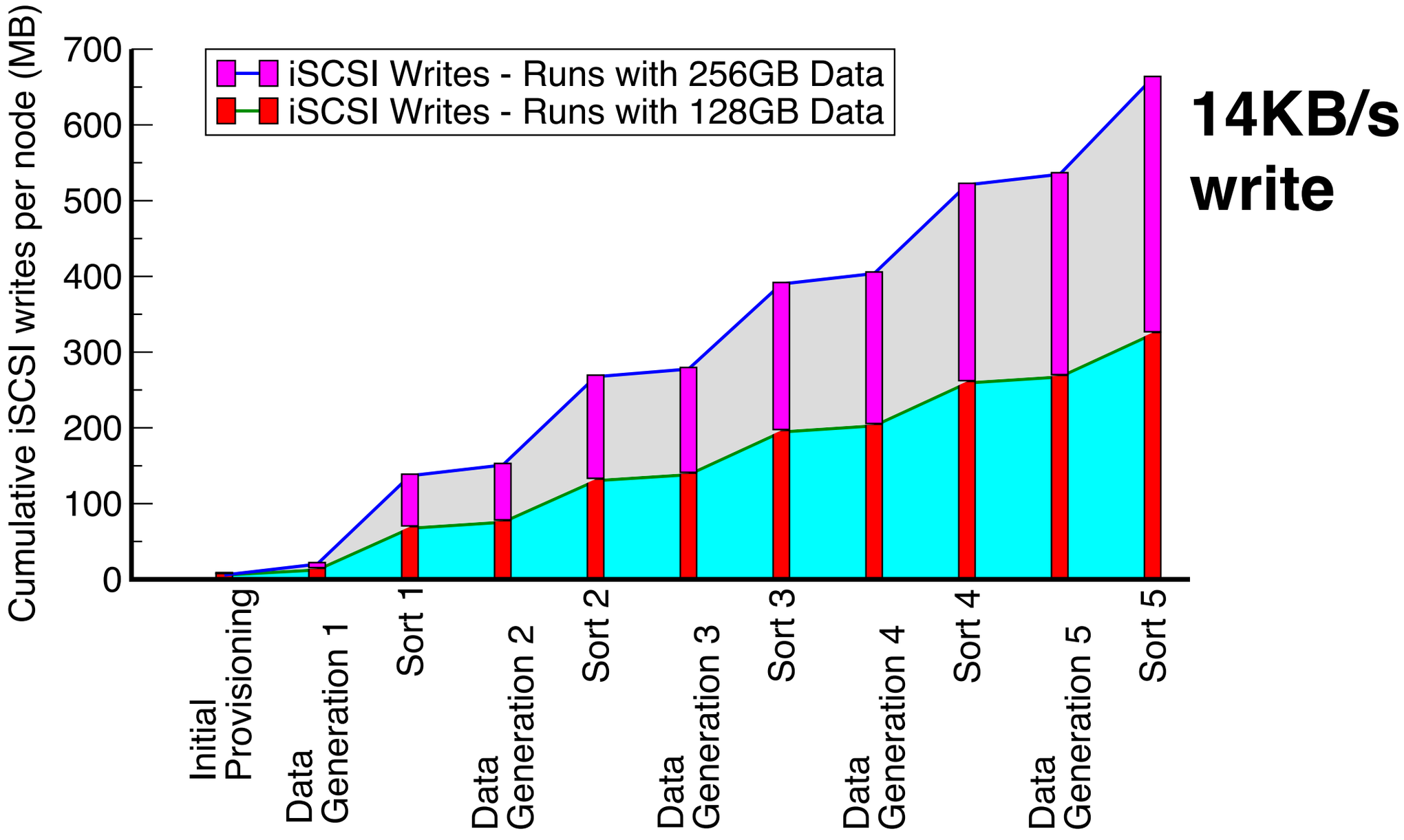
Read Traffic over Boot Drive



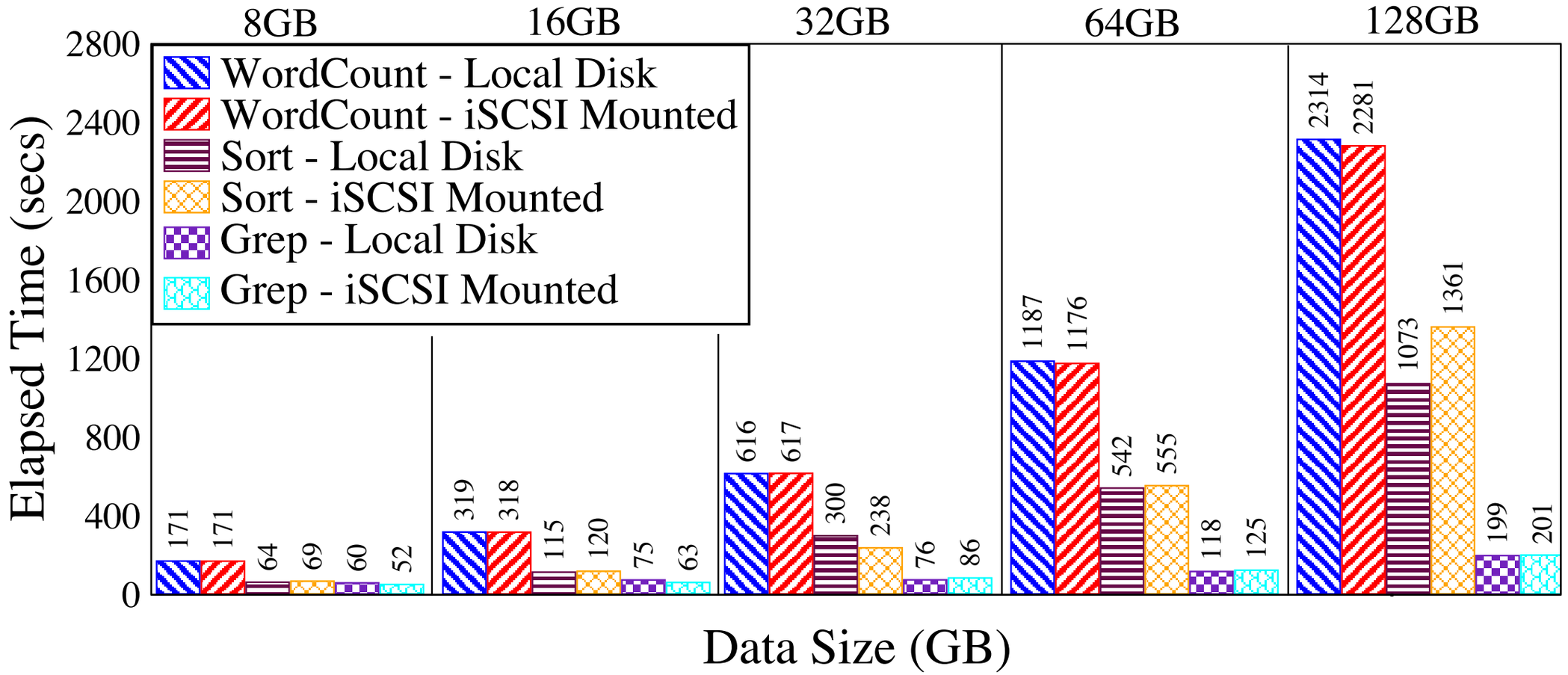
Write Traffic over Boot Drive



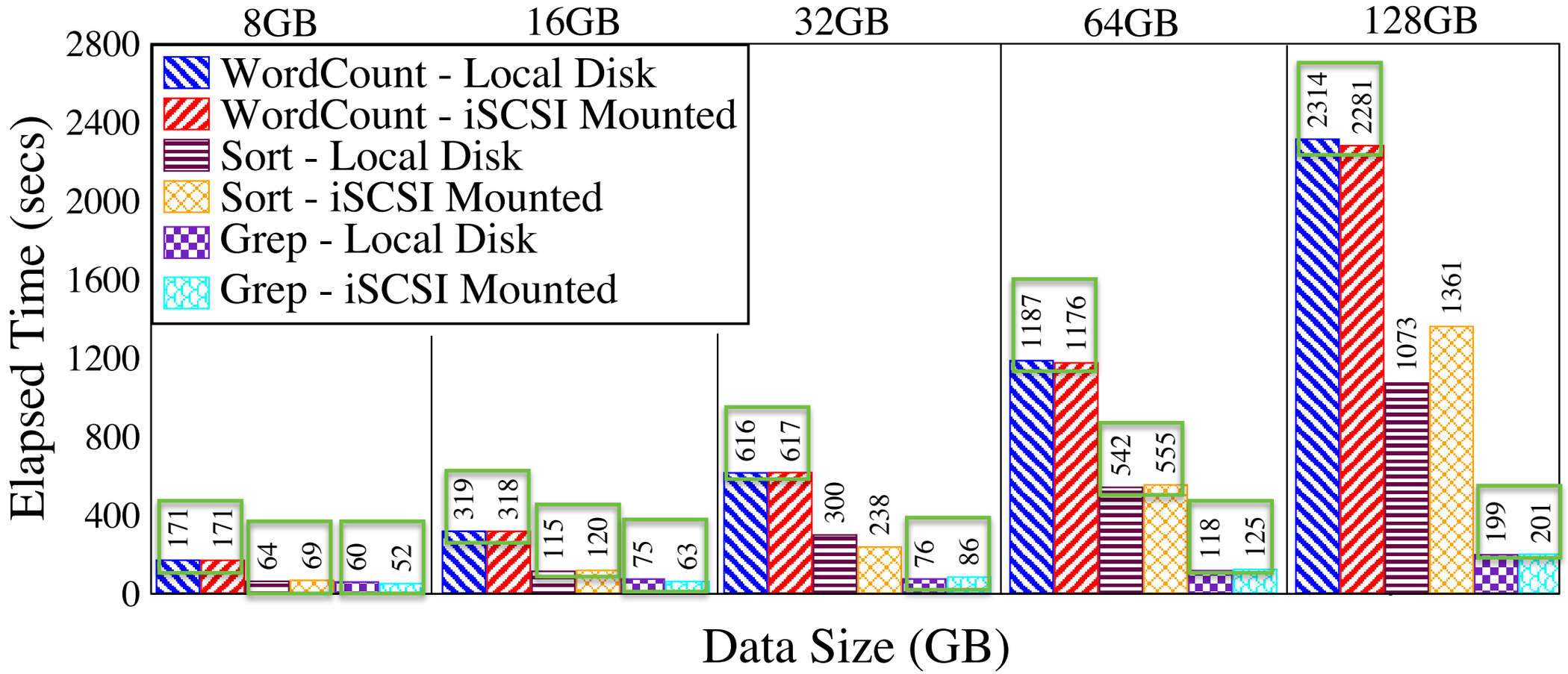
Write Traffic over Boot Drive



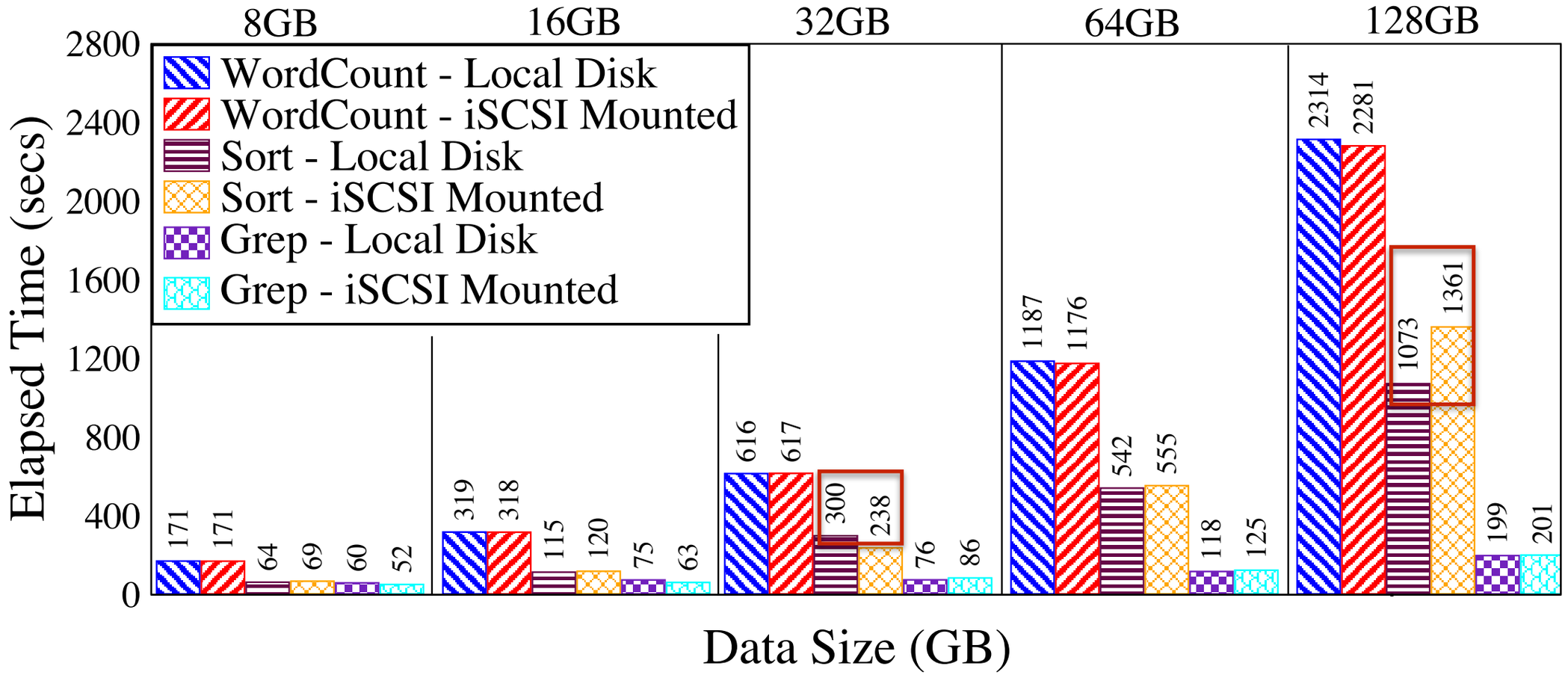
Runtime Performance of Network-Mounted Boot Drive



Runtime Performance of Network-Mounted Boot Drive



Runtime Performance of Network-Mounted Boot Drive

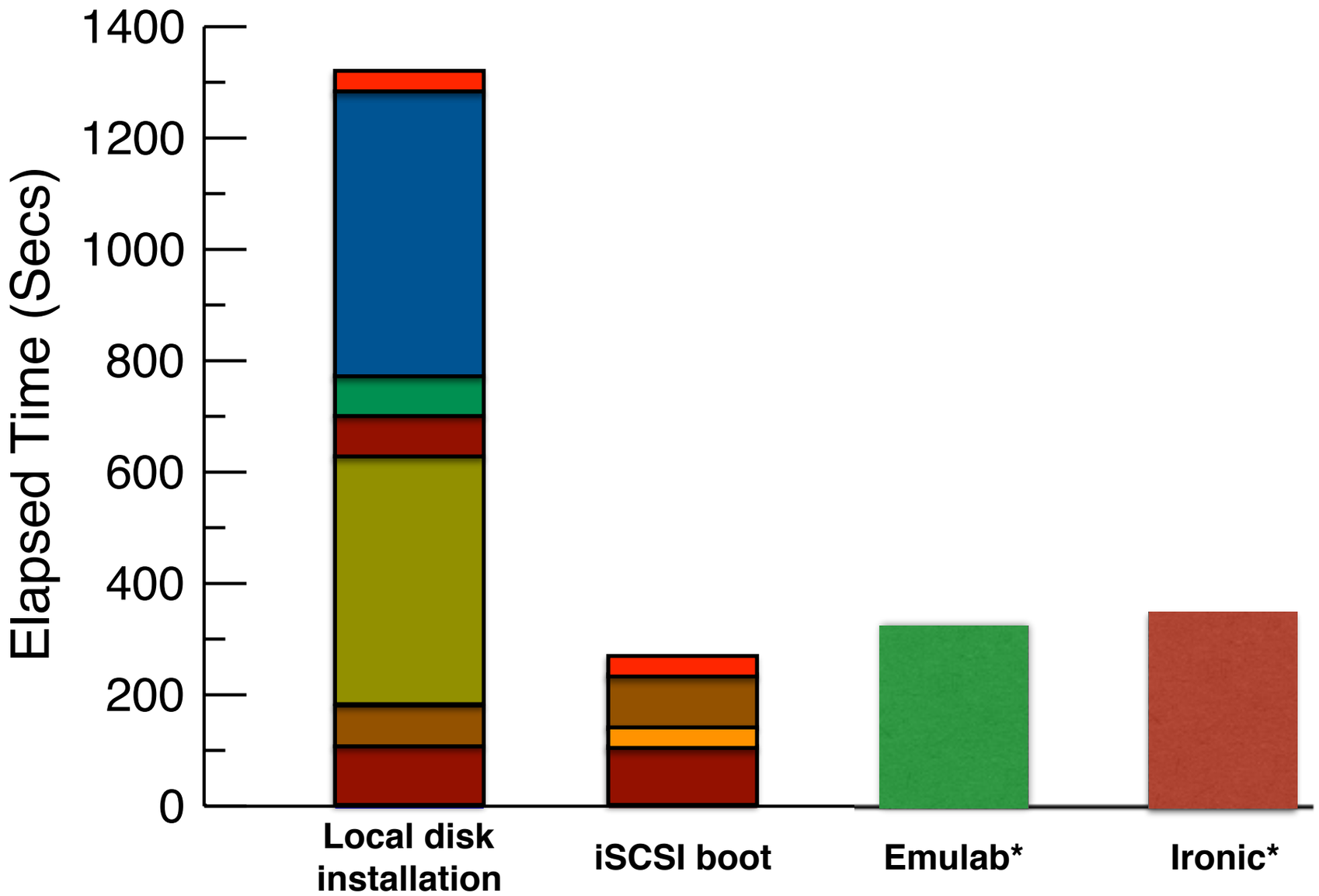


Take-aways

- Network booting the OS for bare-metal BigData
  - uses only a fraction of boot disk during start-up
  - improves provisioning time with no runtime degradation
    - provisioning time < 5 mins, boot disk reads: ~3KB/s, writes: ~14KB/s
- Enormous effort on bare-metal provisioning on local disks may be unnecessary, especially for BigData deployments
- We are building a new Bare Metal Imaging Service using remote network boot mechanisms
  - enable capabilities available on virtualized platforms (e.g. snapshotting, cloning, …) to bare metal cloud solutions

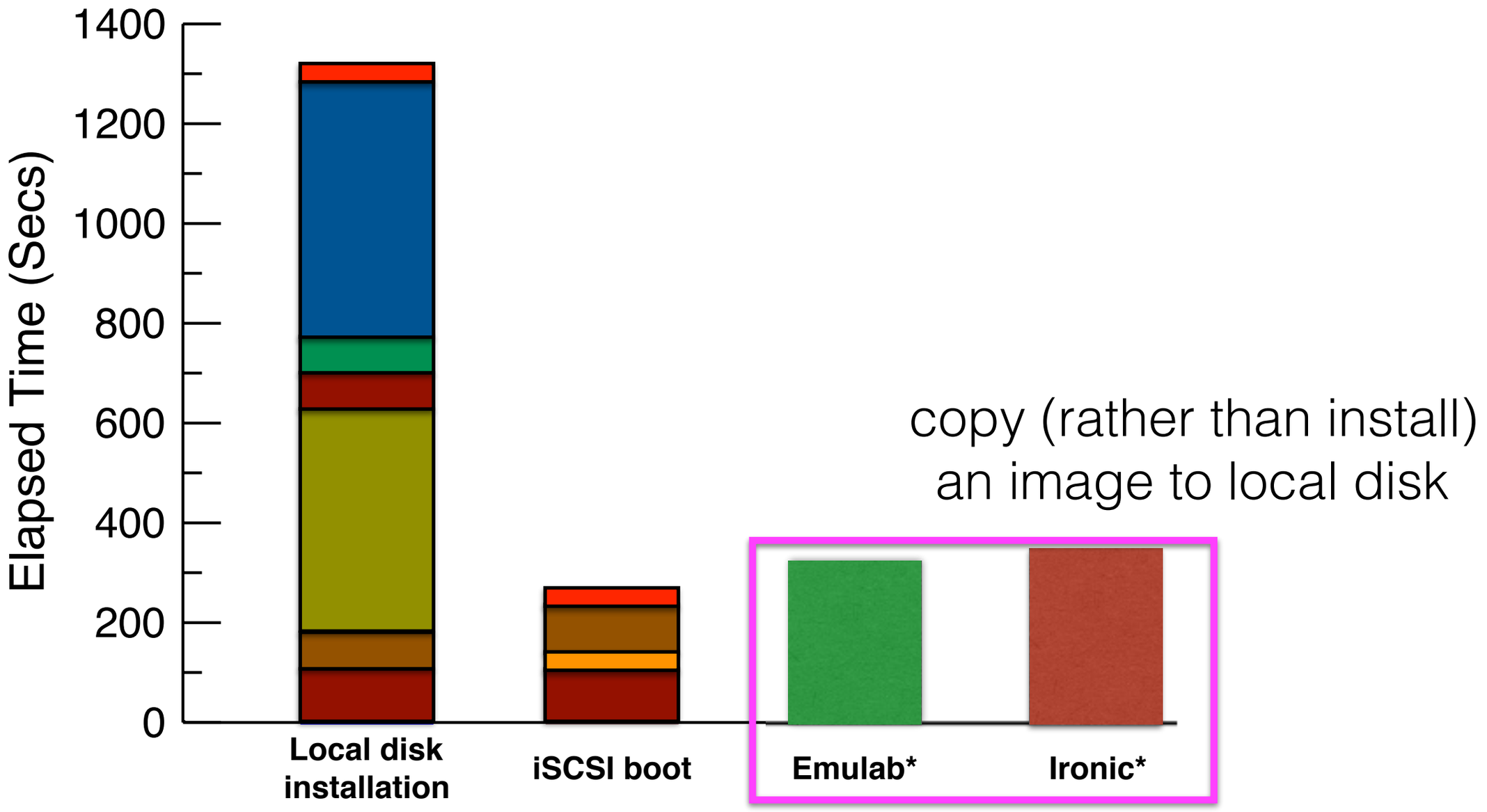
Questions

Provisioning Time



* A. Chandrasekar and G. Gibson, “A comparative study of baremetal provisioning frameworks,” Parallel Data Laboratory, Carnegie Mellon University, Tech. Rep. CMU-PDL-14-109, 2014.

Provisioning Time



* A. Chandrasekar and G. Gibson, “A comparative study of baremetal provisioning frameworks,” Parallel Data Laboratory, Carnegie Mellon University, Tech. Rep. CMU-PDL-14-109, 2014.