

# Cloud Native

Microservices Best Practices

# Outline

Why Microservices?

Introducing Cloud Native

Unikernel & LightVM

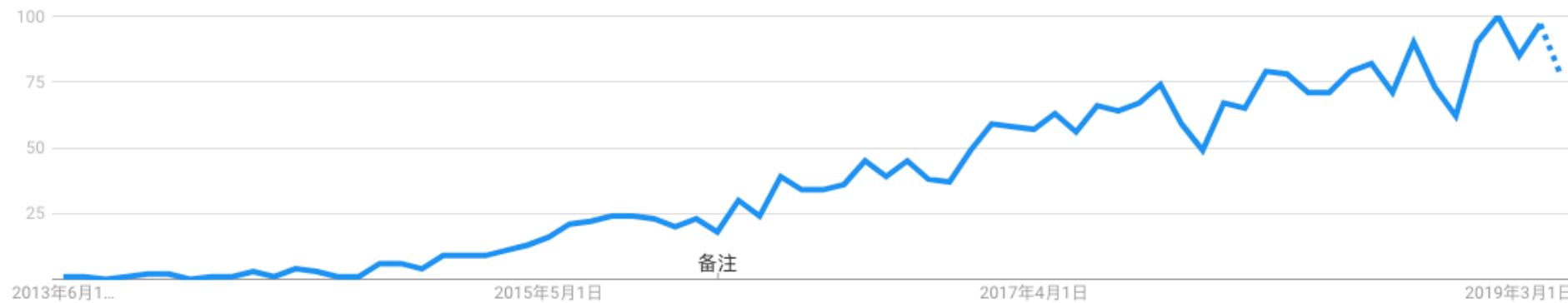
References

# Why Microservices?

# Disaster

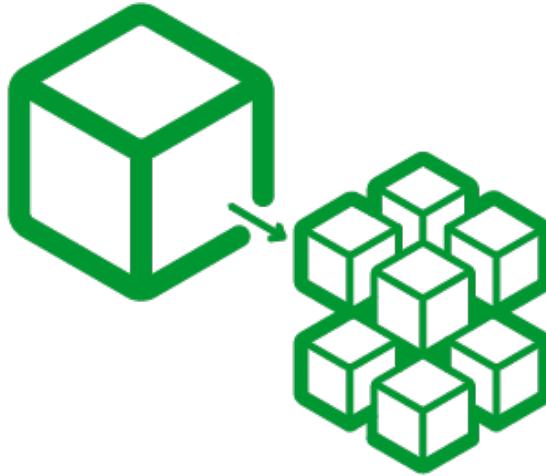


# Why Microservices?



Microservices have been getting more and more popular in recent years

# Why Microservices?



Motivations, Issues and Benefits for Migrating to  
Microservices

# Motivations

- Maintainability
- Delegation of Team Responsibilities
- Fault Tolerance
- Because everybody does
- Devops Support



# Benefits

- Maintainability Improvement
- System Understandability
- ROI
- Architectural Complexity Reduction
- Simplifies Distributed Work



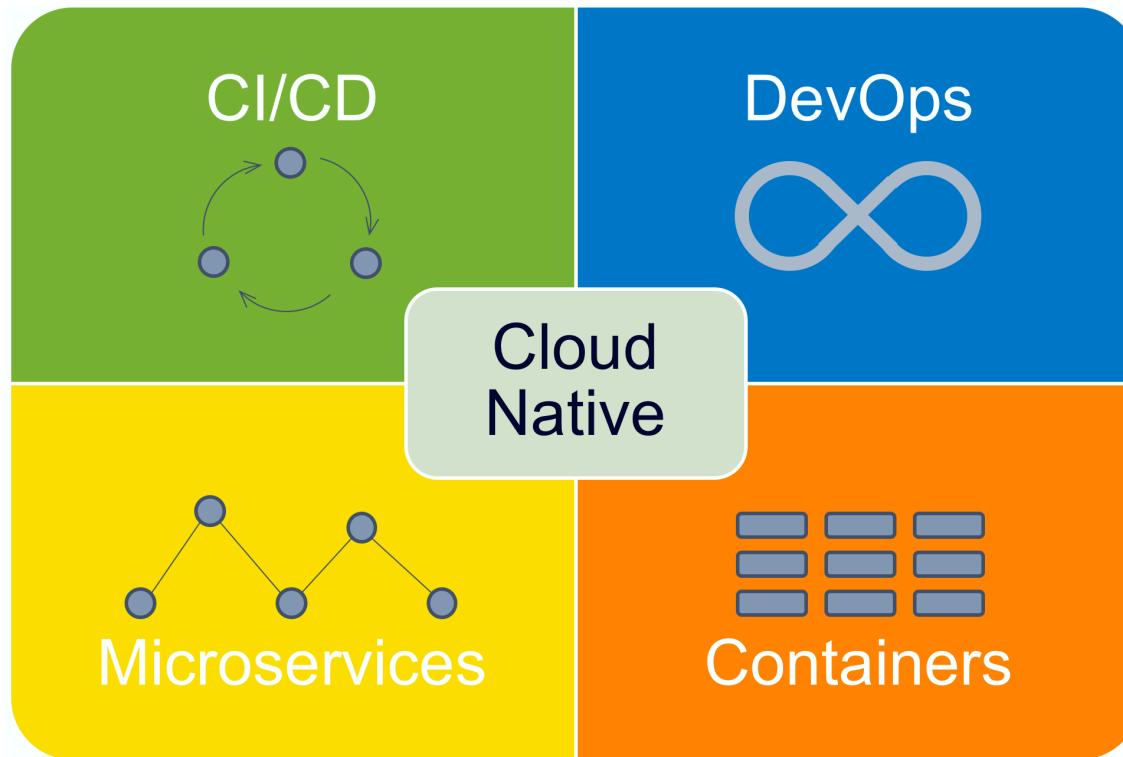
# Issues

- Database Migration and Data Splitting
- Effort Estimation and Overhead
- People's mind
- Effort Required for Library Conversion
- Effort Required for the Infrastructure

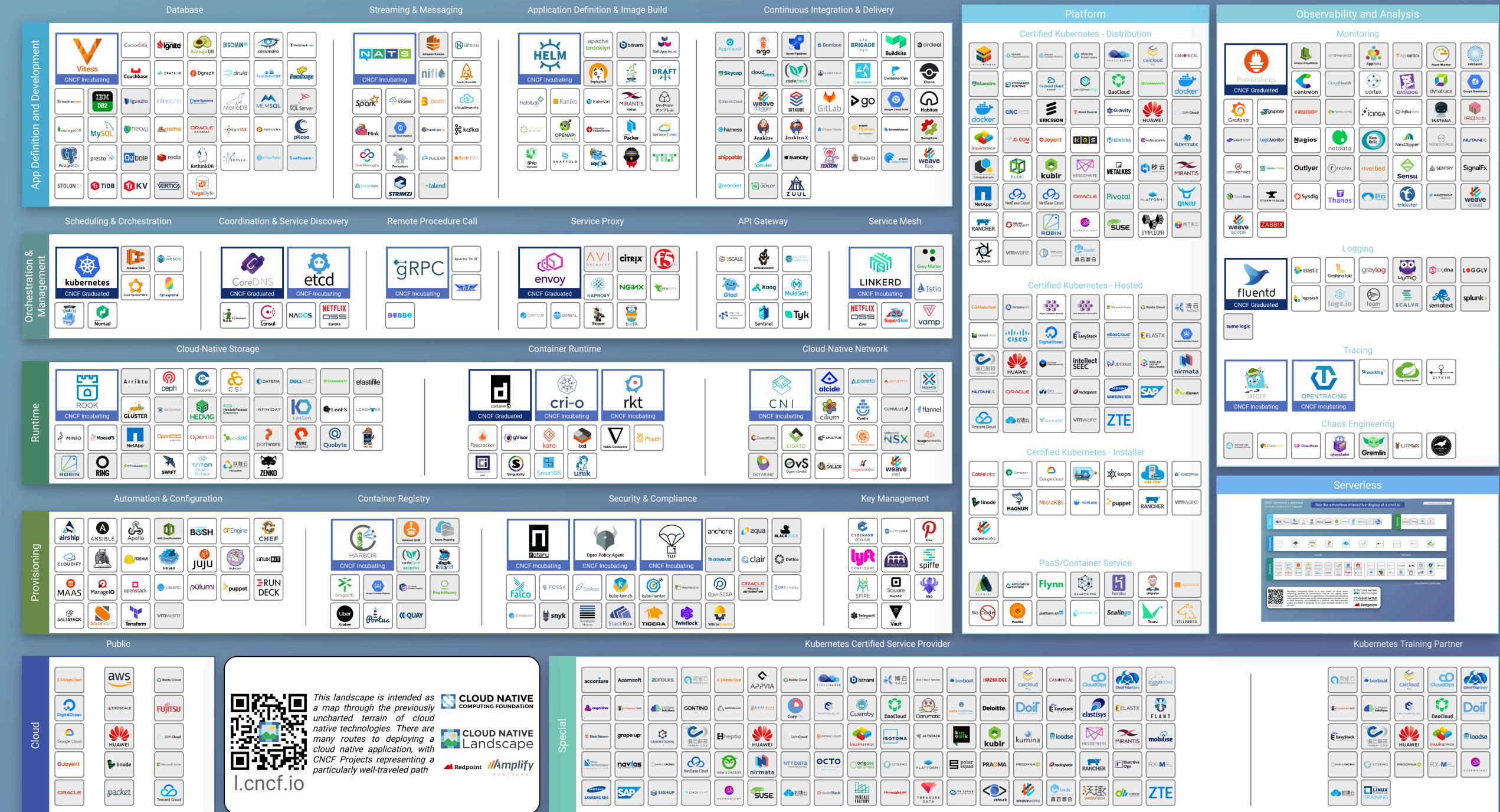


# Introducing Cloud Native

# Cloud Native Features



Overwhelmed? Please see the CNCF Trail Map. That and the interactive landscape are at [l.cncf.io](https://l.cncf.io)



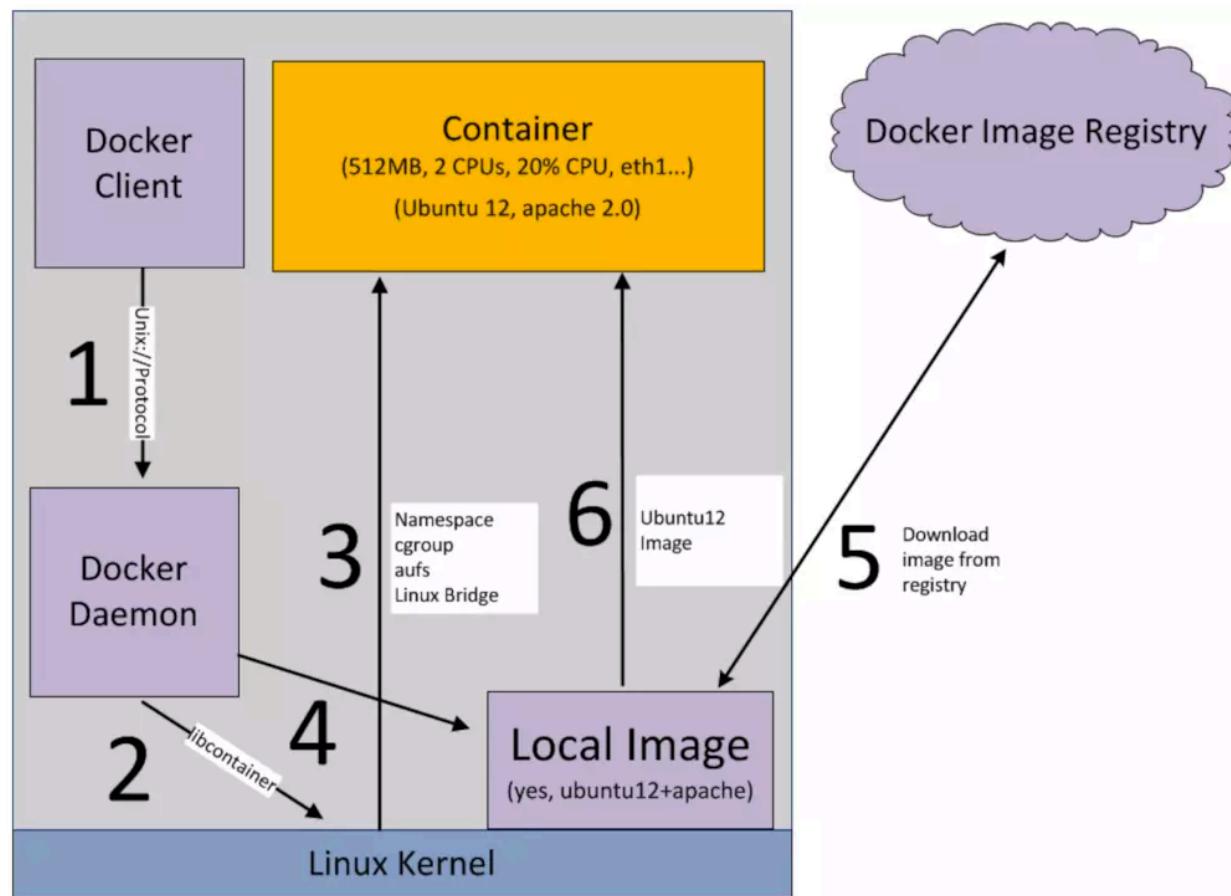
# A Cloud Native Example



# Open Issues in Cloud Native

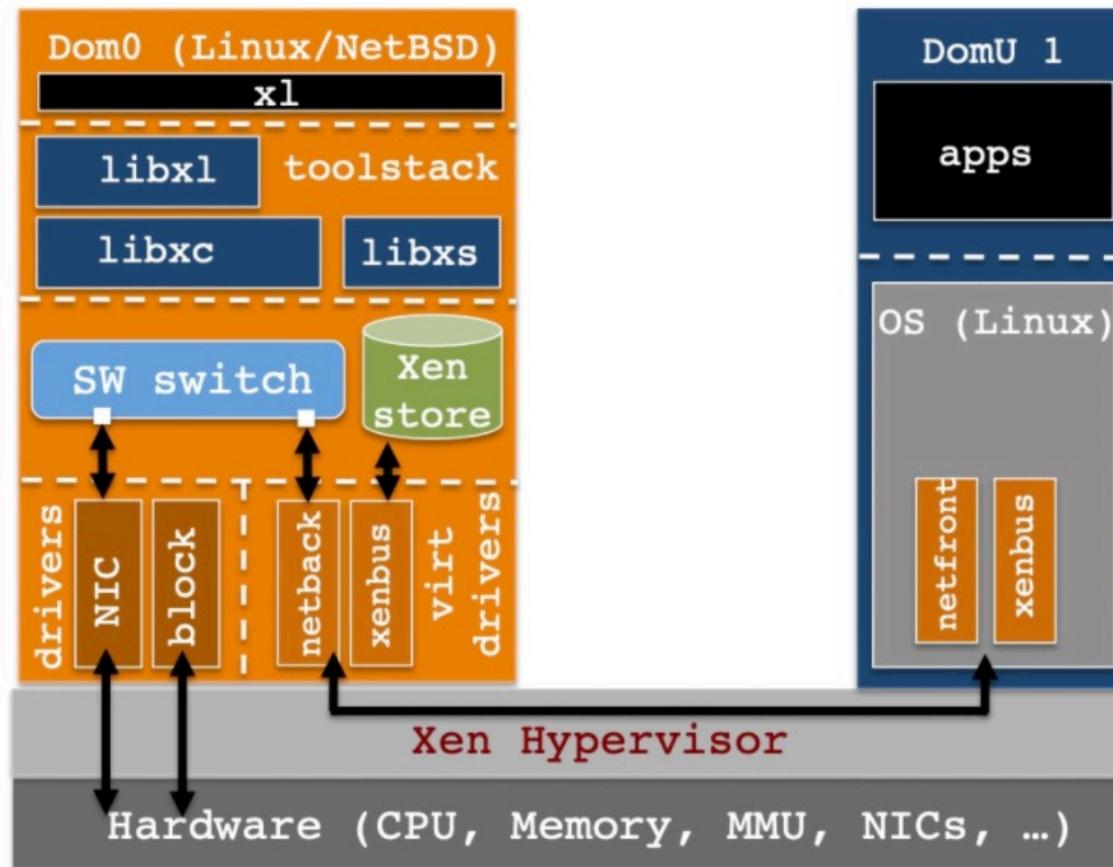
- Application Monitoring
- Performance and Isolation

# Docker Startup

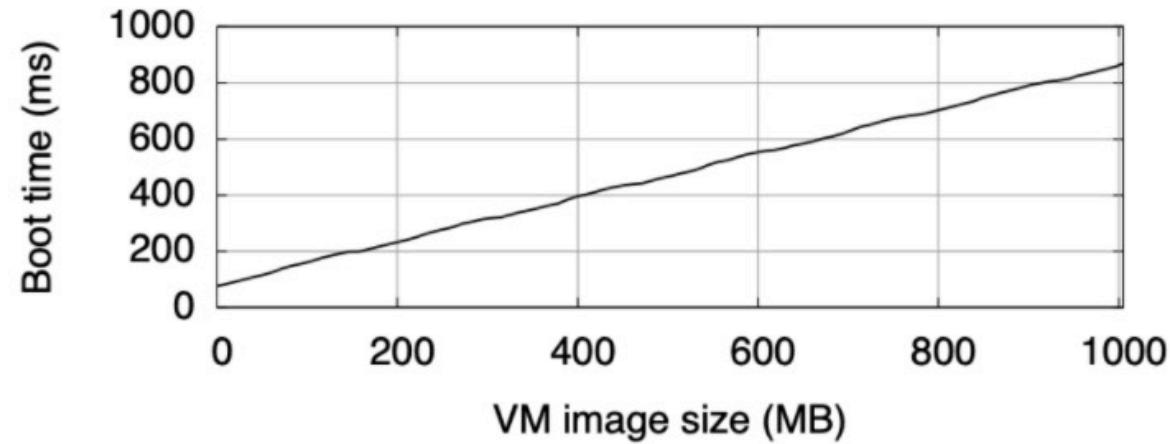


Can we have the improved isolation of VMs,  
with the efficiency of containers?

# Xen Architecture

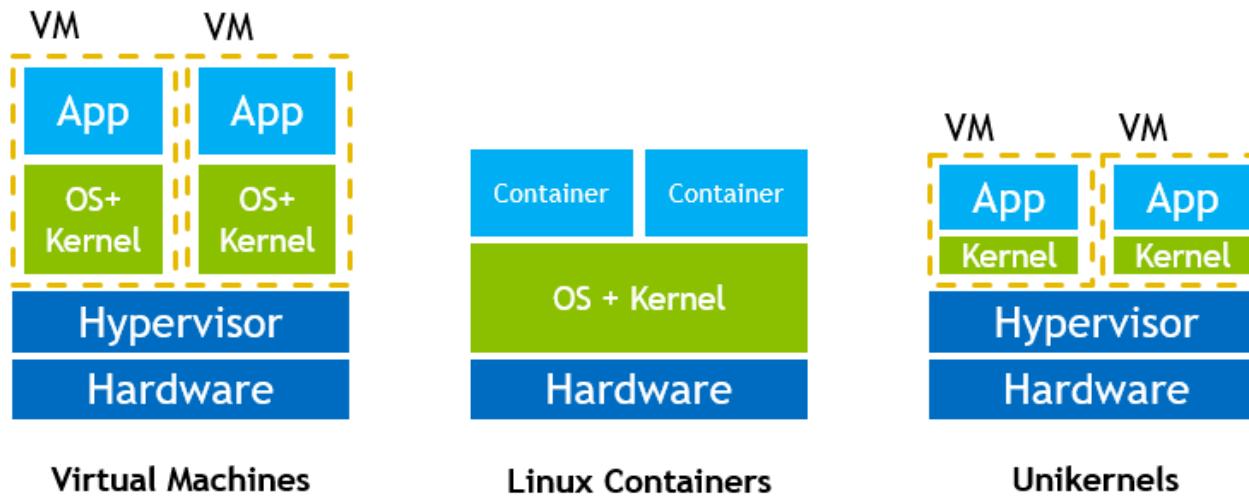


# About boot time



# Unikernel

# What is Unikernel ?



Unikernels are specialized, single-address-space machine images constructed by using library operating systems.

# Unikernel Features

- Small
- Fast
- Safe

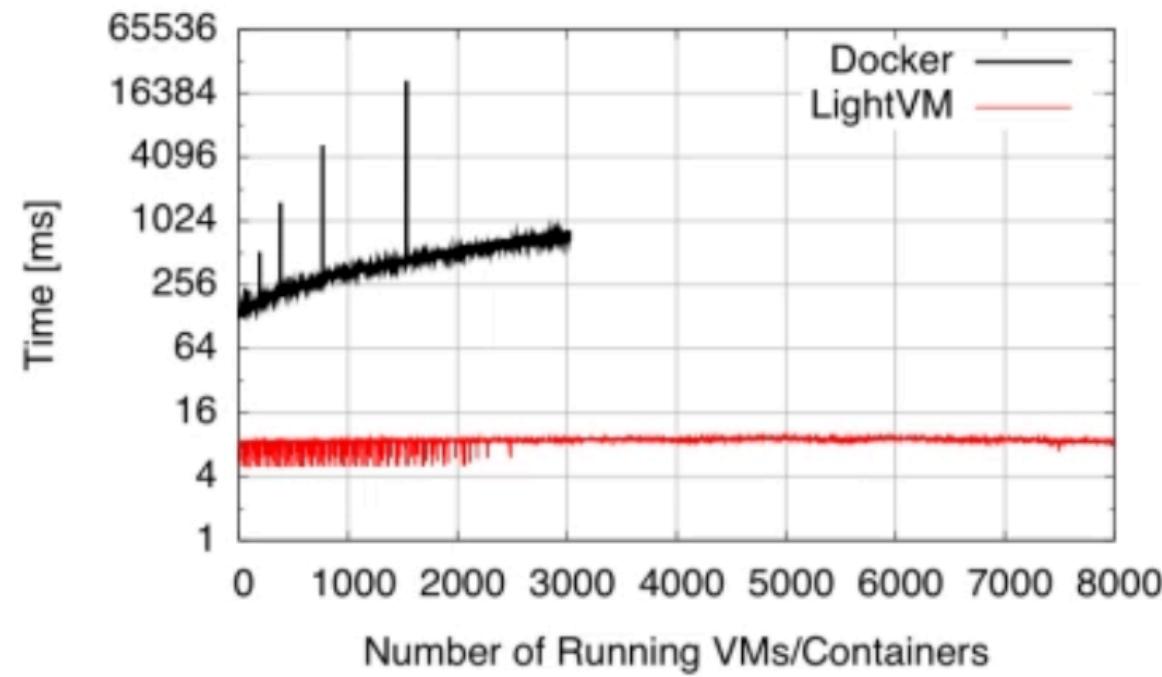


# Container VS Unikernel

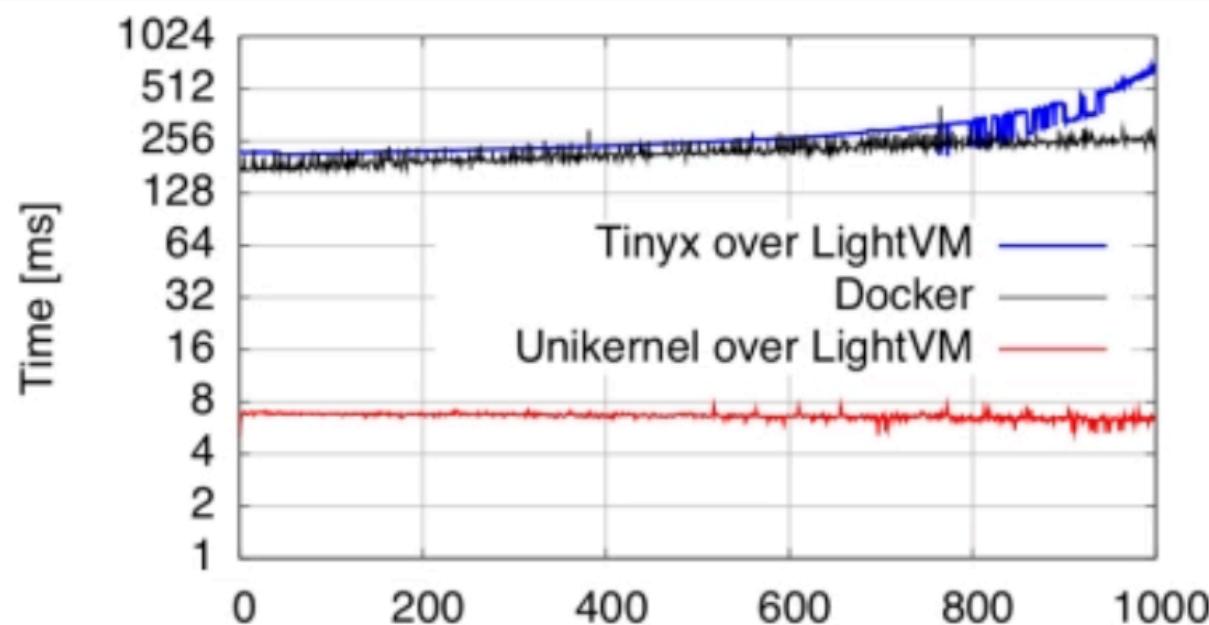
<b>Image Size</b>	<b>Dozens MB ~ Hundreds MB</b>	<b>KB level</b>
<b>Boot time</b>	Several to dozens seconds	Ms level
<b>Feature</b>	Mul containers in one kernel	Every process has one build-in kernel
<b>Environment</b>	LXC	Hypervisor
<b>Address space</b>	Isolation with different containers address space	Single address space
<b>Process</b>	One container can run a set of processes	Only one process in a unikernel
<b>Ecosystem</b>	Easy to build, mature ecosystem	Difficult to build, immature ecosystem

# LightVM

# Hello, LightVM!



# Hello, LightVM!

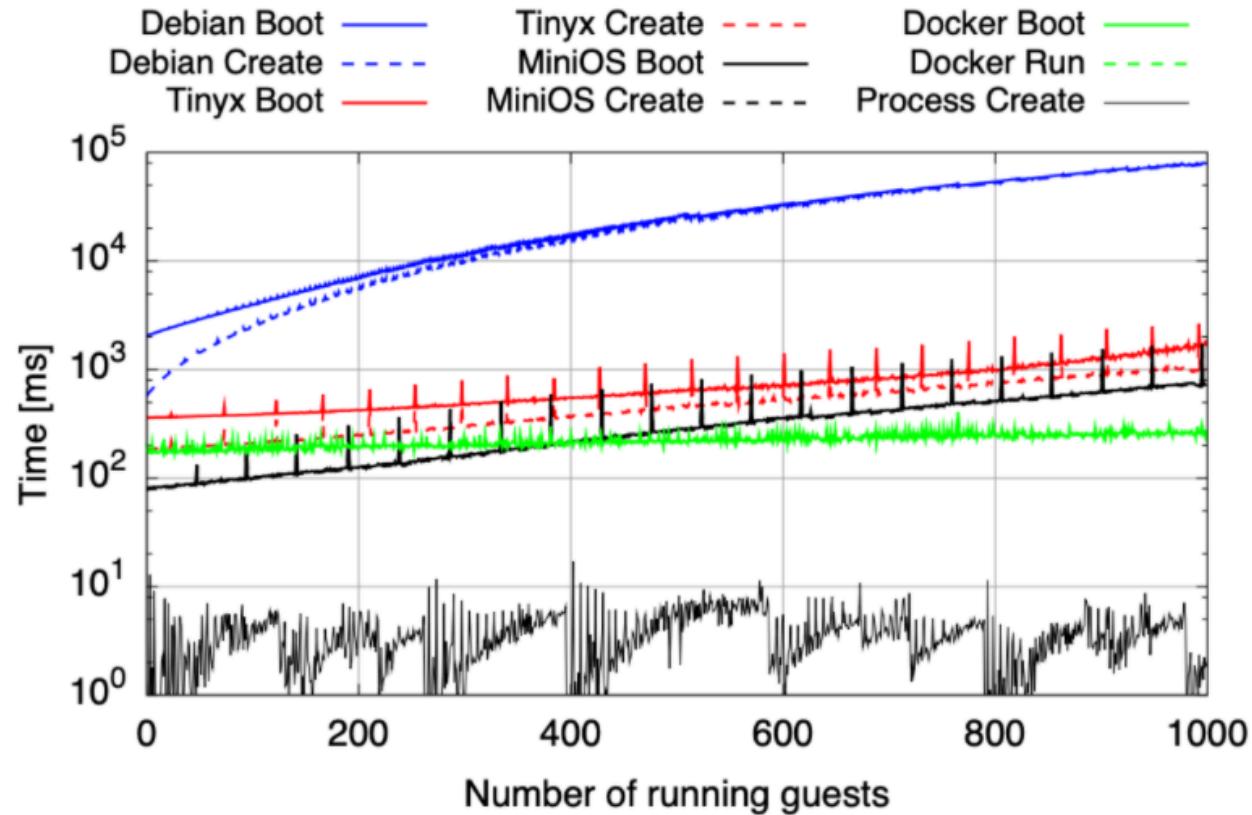


# TinyX

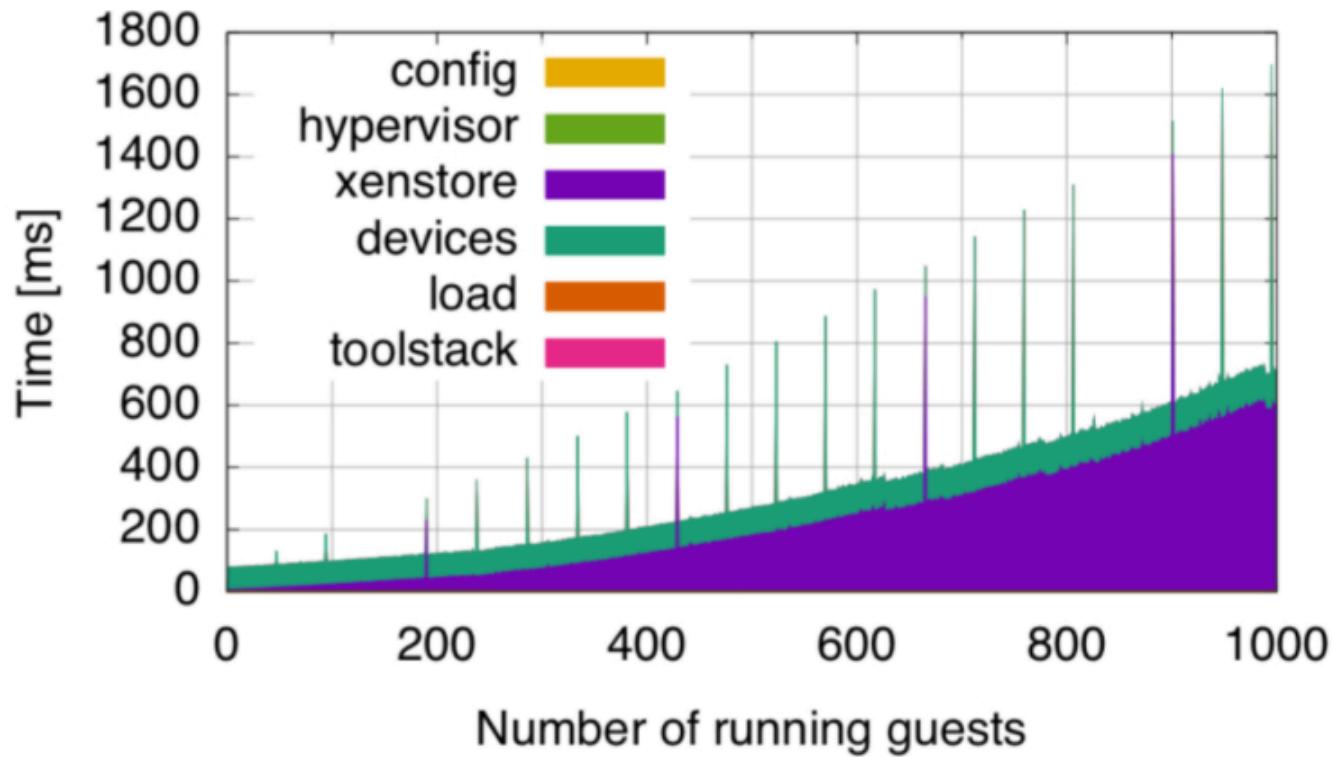
- Use *objdump* command
- Debian package manager

Tinyx creates kernel images that are half the size of typical Debian kernels, and have significantly smaller runtime memory usage (1.6MB for Tinyx vs 8MB for Debian).

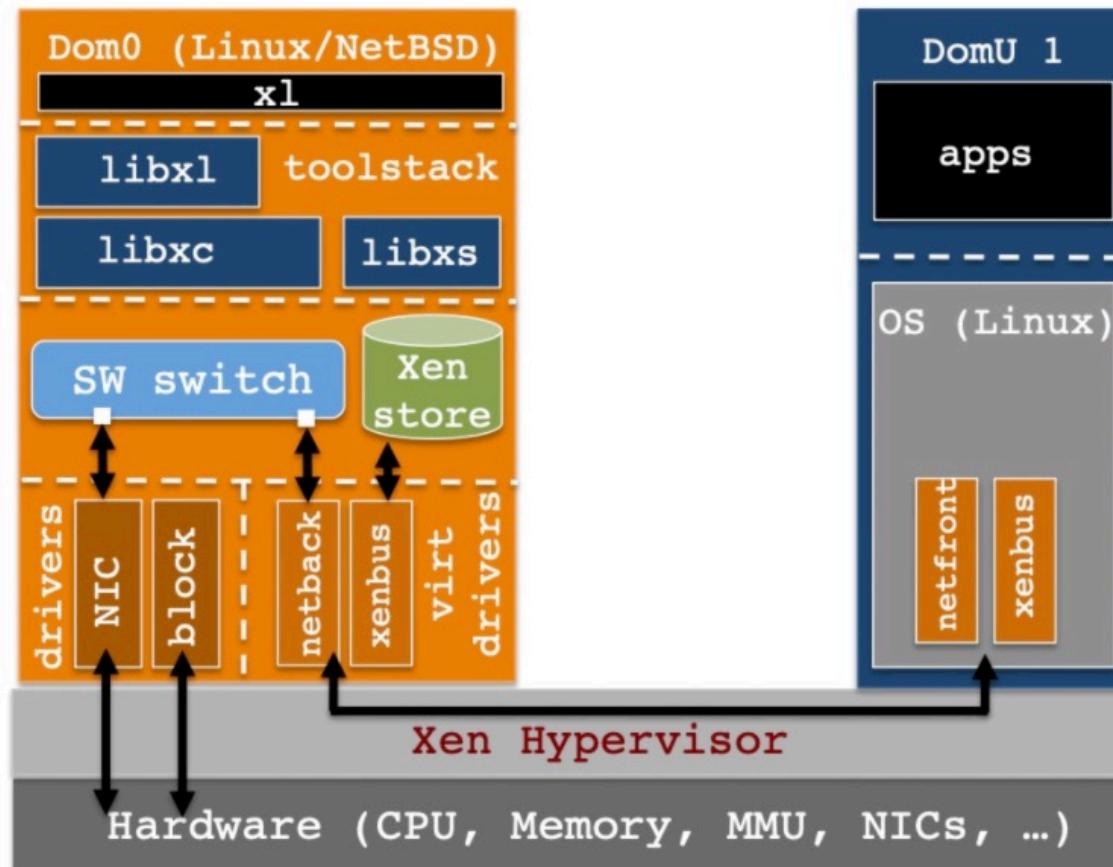
# About create time



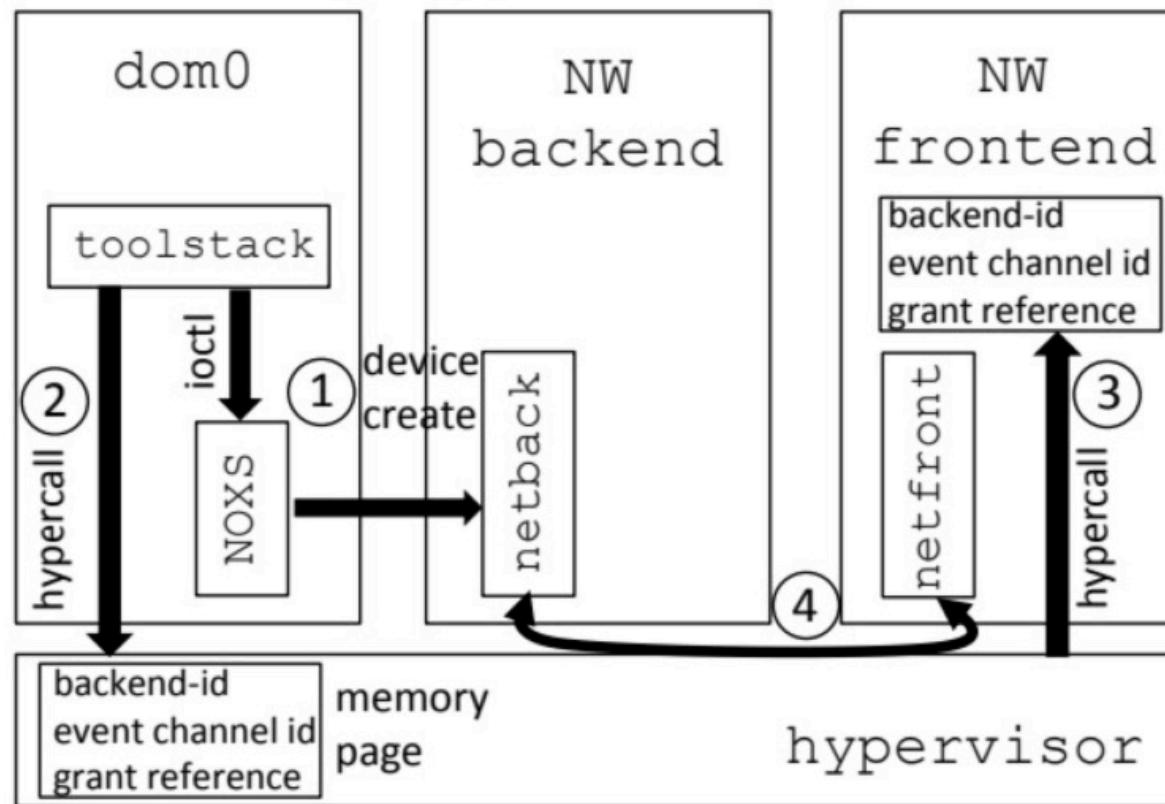
# About create time



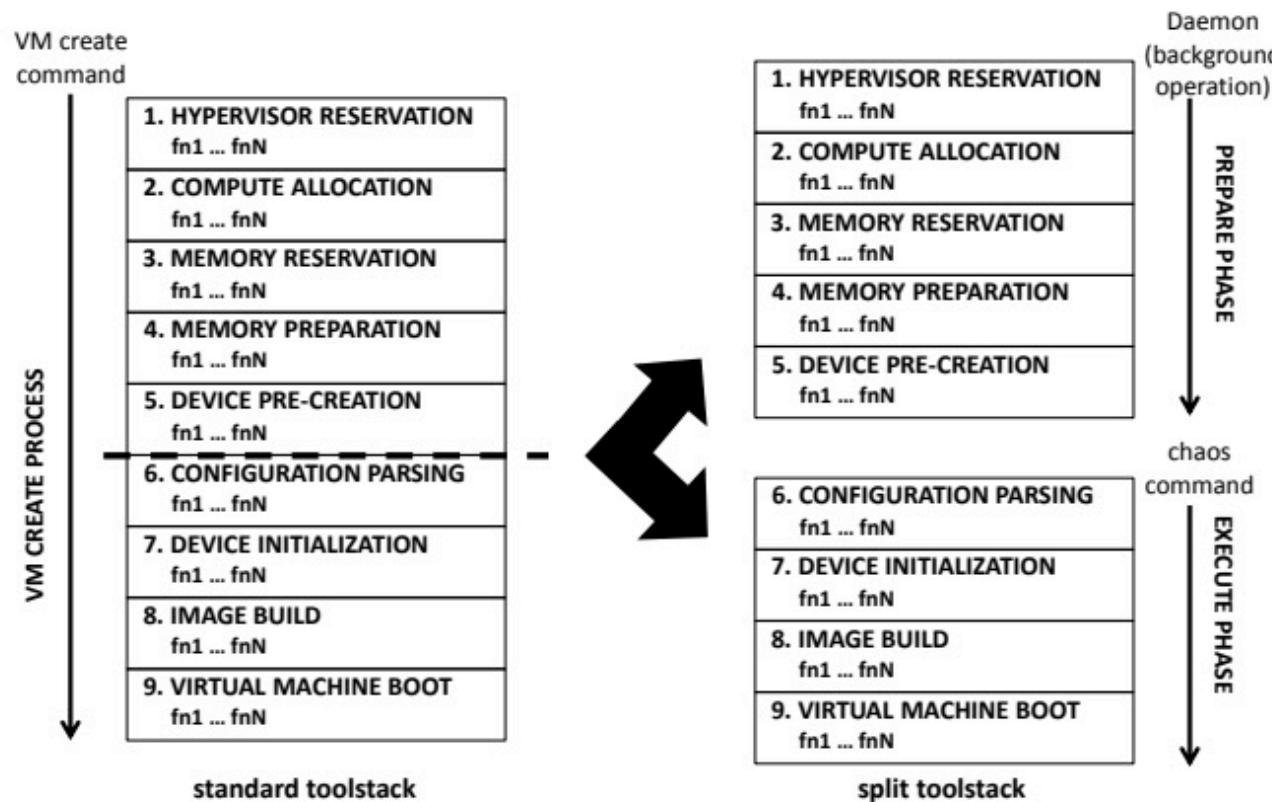
# Xen Architecture



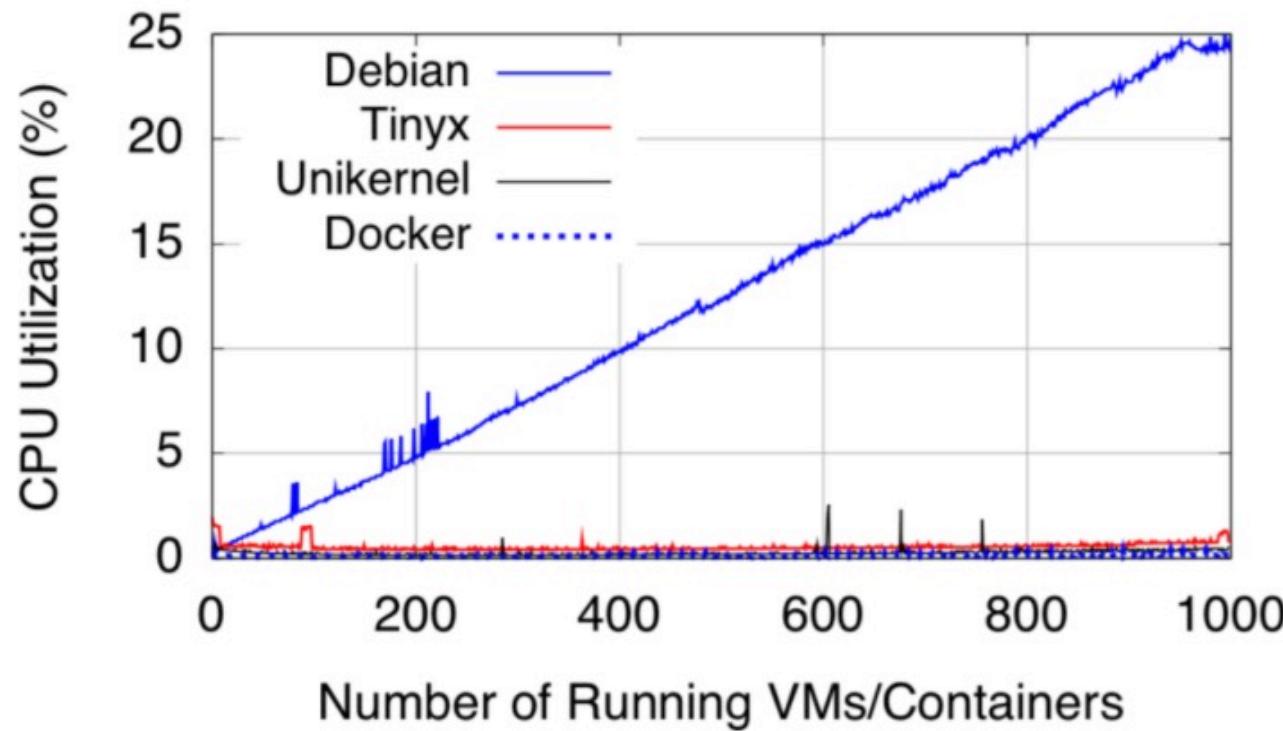
# Noxs



# Toolstack split



# Conclusion



# References

# References

Processes, Motivations, and Issues for Migrating to Microservices Architectures  
IEEE Cloud Computing, 2017

Microservices Architecture Enables DevOps  
IEEE Software, 2016

Open Issues in Scheduling Microservices in the Cloud  
IEEE Cloud Computing, 2016

Unikernels vs Containers: An In-Depth Benchmarking Study in the Context of Microservice Applications  
SC2, 2018

My VM is Lighter(and Safer) than your Container  
SOSP, 2017

Thanks