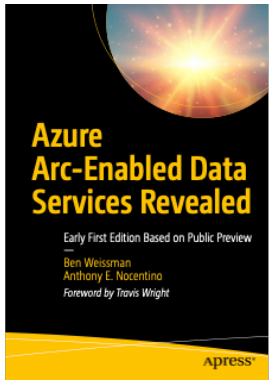
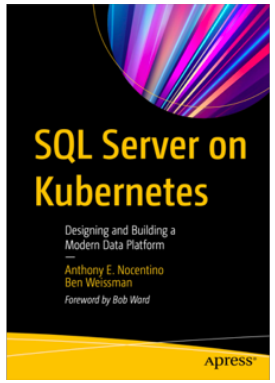
Containers - You Better Get on Board!

Anthony E. Nocentino
anocentino@purestorage.com

Anthony E. Nocentino

- Principal Field Solution Architect @ Pure Storage
  - Specialize in system architecture and performance
  - Masters Computer Science
- email: anocentino@purestorage.com
- Twitter: @nocentino
- Blog: www.nocentino.com
- Pluralsight Author: www.pluralsight.com



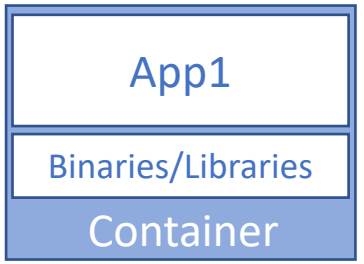


Agenda

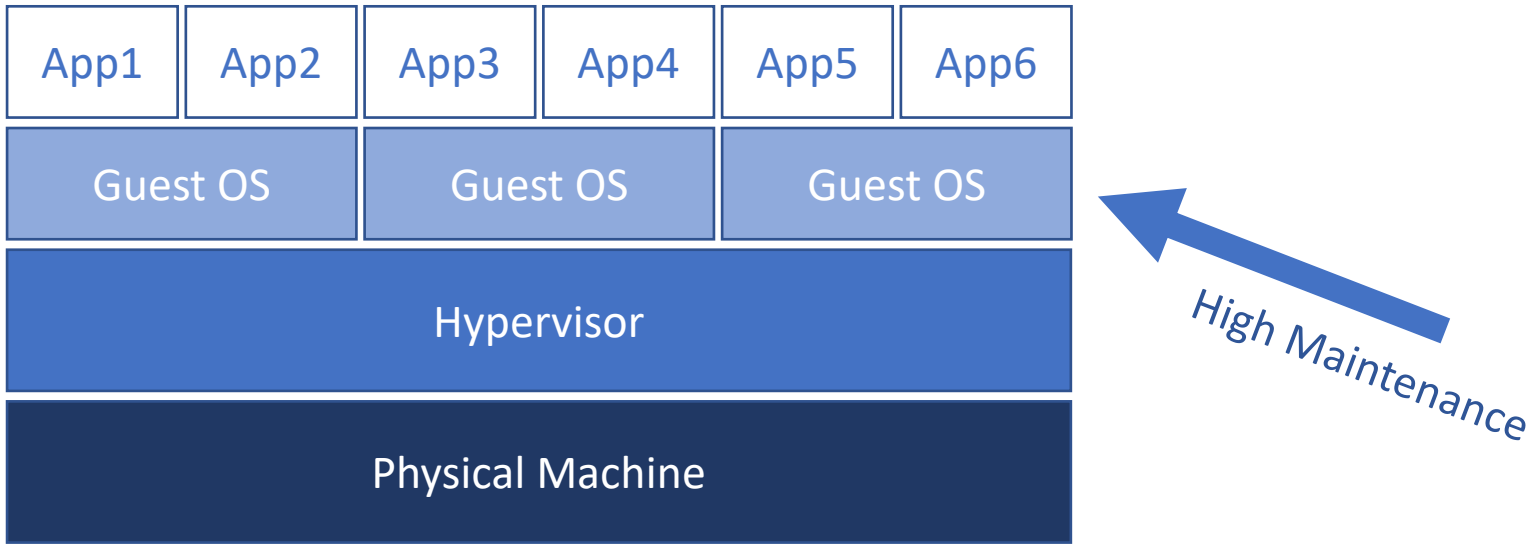
- Introducing Containers
- Building and Running Apps in Containers
- The Container Universe
- Hands on with Containers
- Container Orchestration and Kubernetes

Introducing Containers

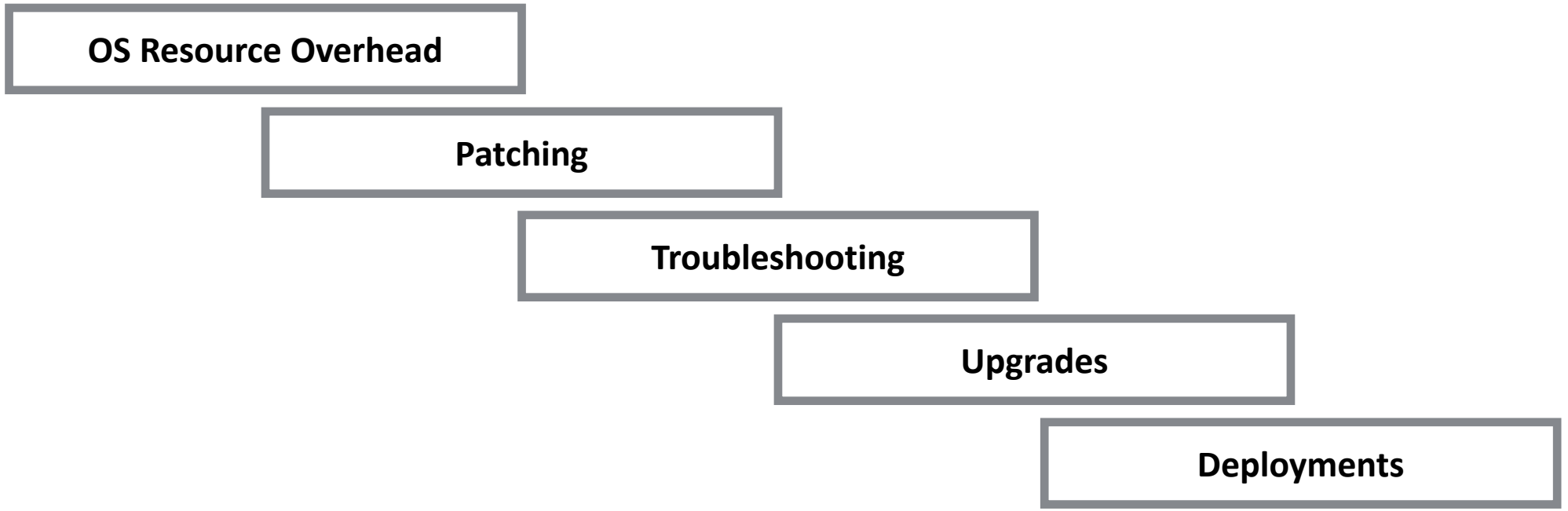
- Operating system virtualization
  - Shared kernel and system resources
- Container…contain…
  - Binaries, libraries and file system
- One app inside the container
  - This is the unit of work
- Containers are ephemeral
- Let's start off with a comparison…



Virtual Machines

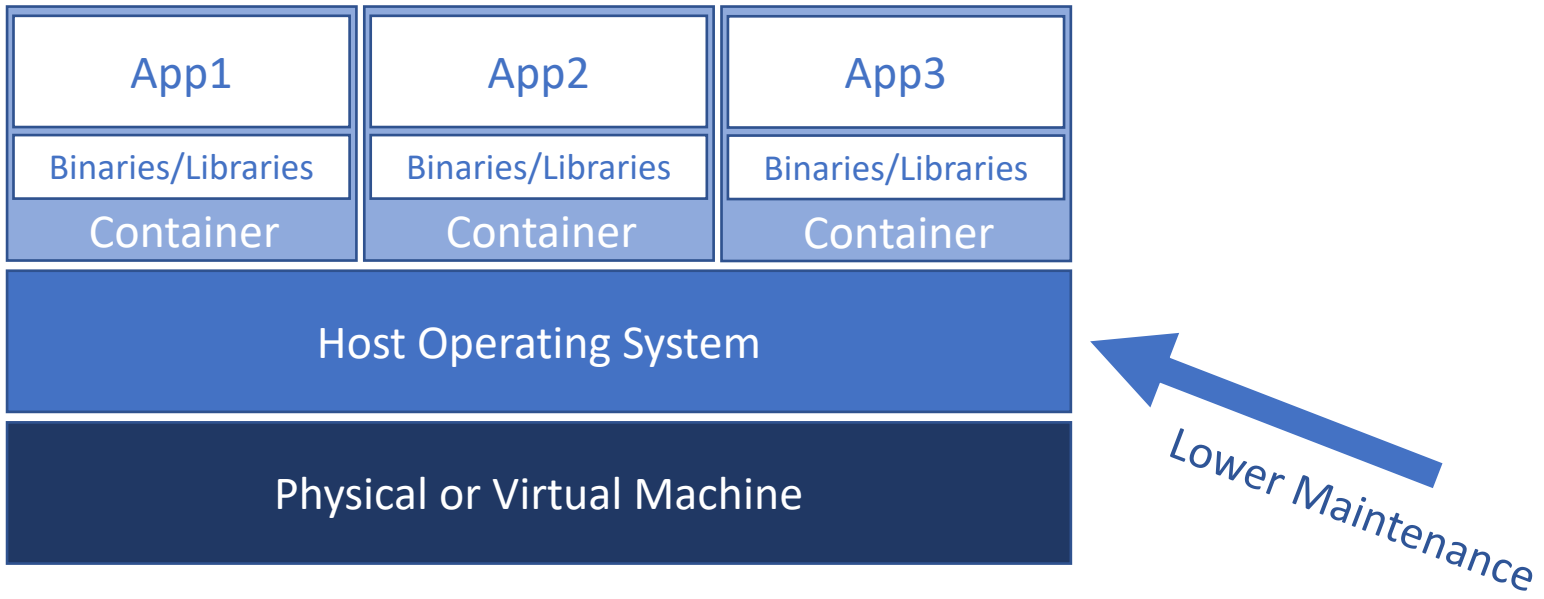


What's so Hard About Virtual Machines?



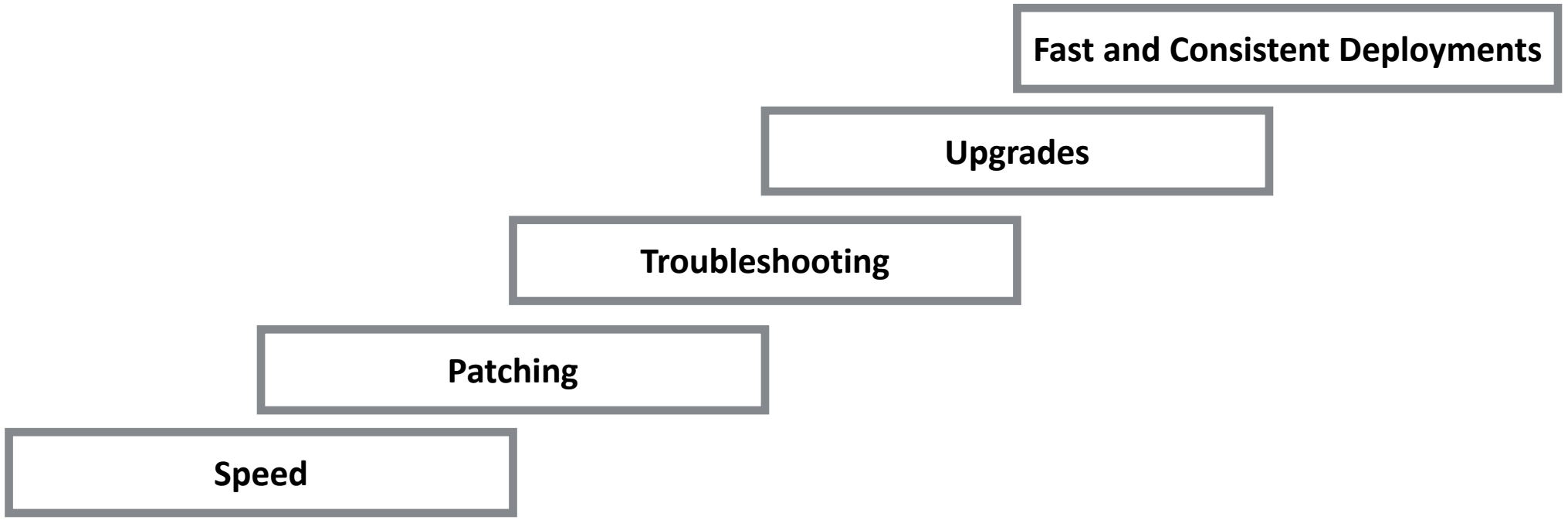
Does any of this move your business forward?

Containers


App1
App2
App3
Binaries/Libraries
Binaries/Libraries
Binaries/Libraries
Container
Container
Container
Host Operating System
Physical or Virtual Machine
Lower Maintenance

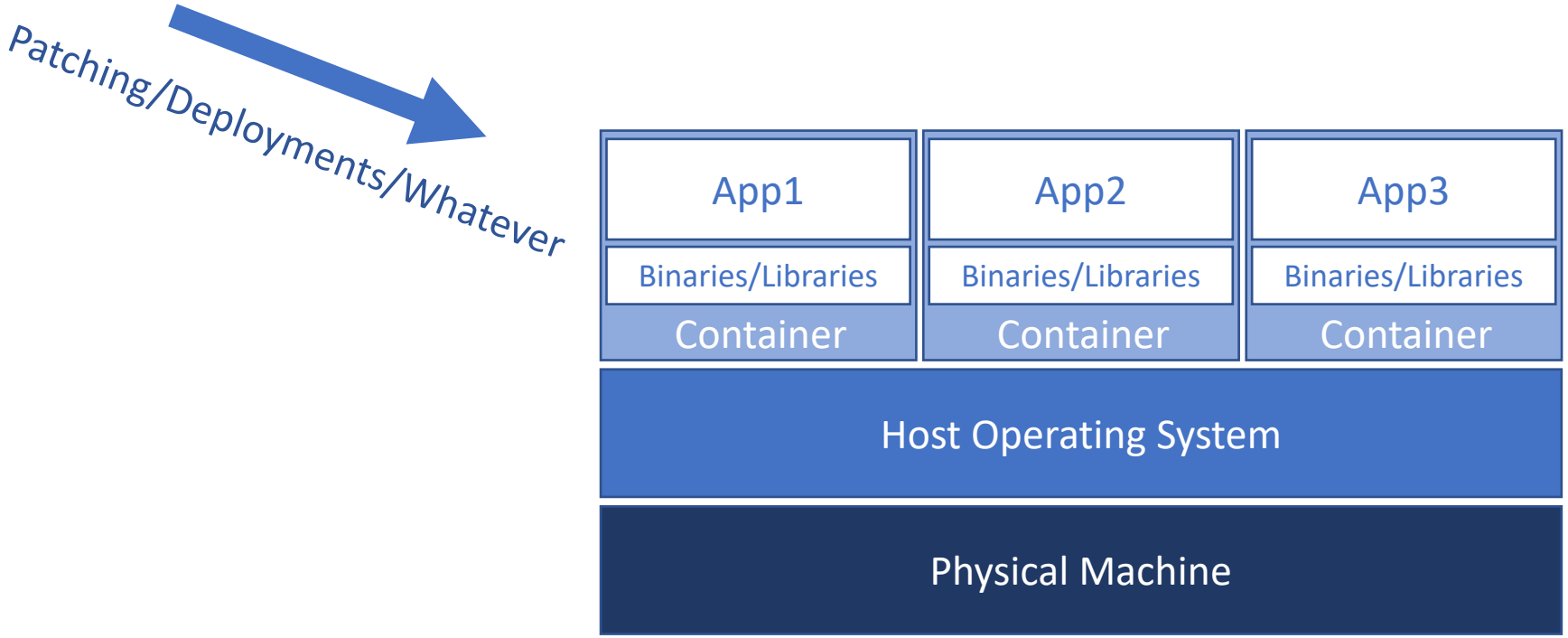
It’s all about goin’ fast!

What do Containers Bring to the Table?



Services, we care about getting work done!

Containers

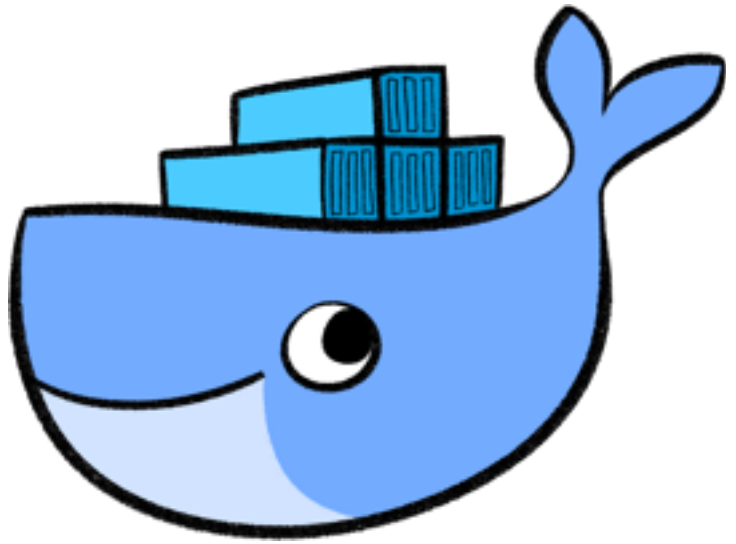

Patching/Deployments/Whatever
App1
App2
App3
Binaries/Libraries
Binaries/Libraries
Binaries/Libraries
Container
Container
Container
Host Operating System
Physical Machine

Containerizing Apps and Data Centers

- Reducing development time
- Deployment automation – speed and consistency
- Enables DevOps and CI/CD scenarios
- Orchestration
- Rethink how you deploy - it's the **service**, not the server

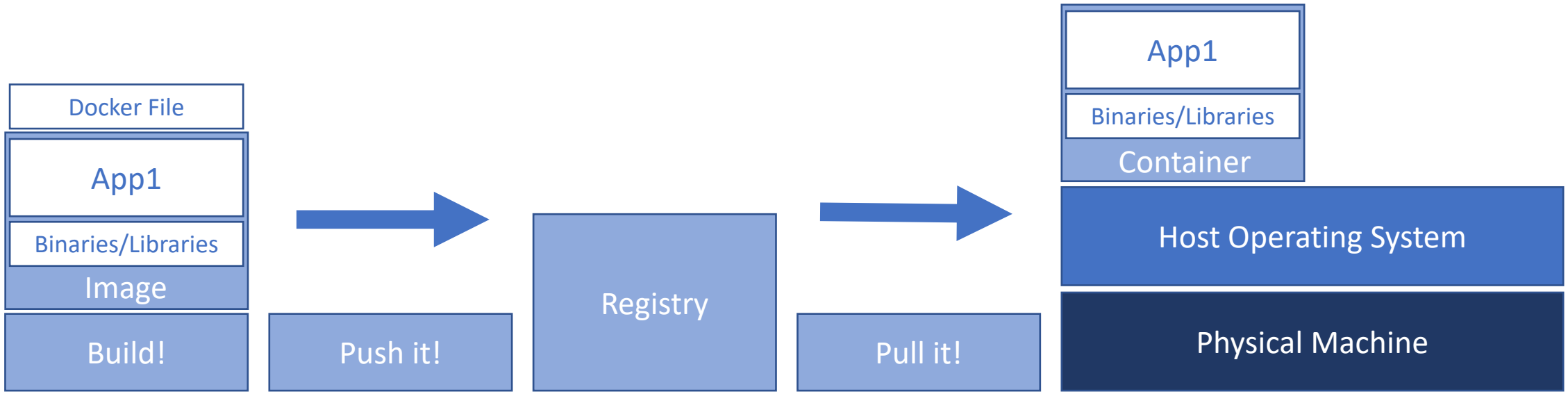
The Container Universe

- Docker
  - Linux
  - Windows
  - Mac
- Docker Inc.
- Other Container Runtimes
  - Containerd
  - CoreOS
  - Podman
  - Windows
  - chroot…chwhat?



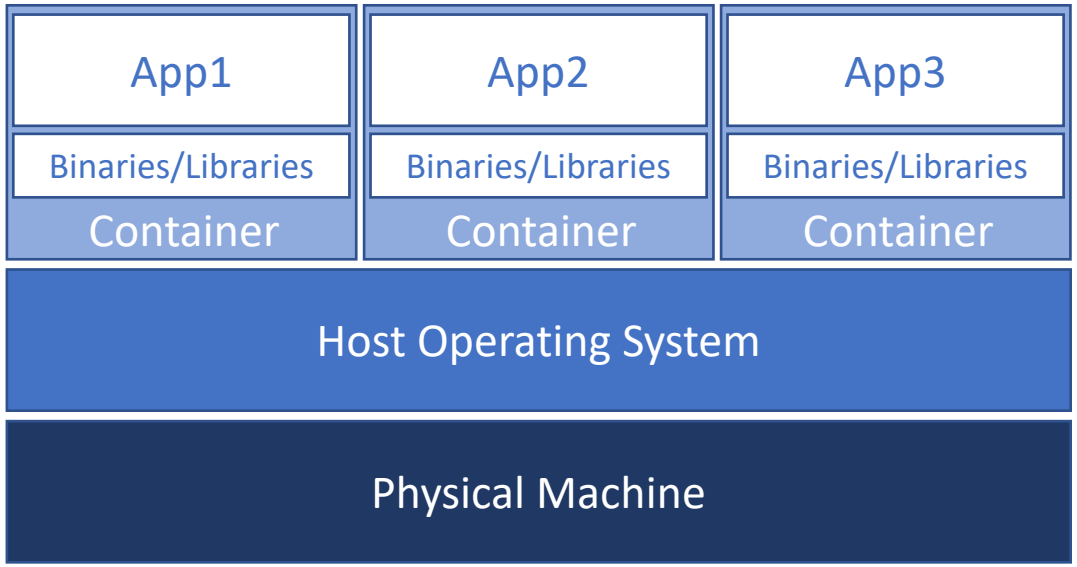
Getting Containers

- **Images** – code, runtimes, libraries, environment variables
- **Registries** – where images live. Docker Hub, Azure Container Registry, internal
- **Docker Files** – defines the container image



Container Internals

- Shared OS
- Resource isolation
  - Namespaces
    - **Process Isolation** - PID
    - **File System** – MNT
    - **Network** – NET
    - **Interprocess Communication** - IPC
    - **Kernel Isolation** - UTS
- Resource governing
  - cgroups
- chroot file system



Demo!

- Pull an Image
- Run a Container
- Access our application
- Connect to the Container
- Persisting data with a Container

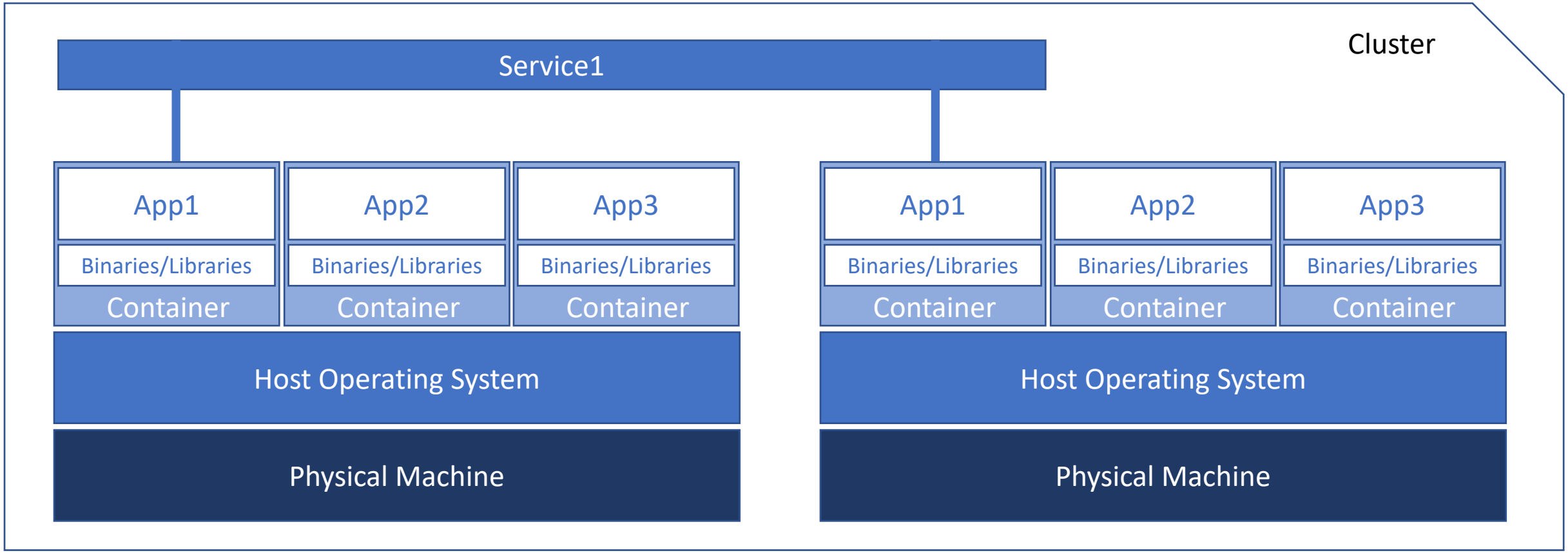
Container Orchestration

- Workload placement
- Managing state, starting things up and keeping things up
- Load balancing for services
- Networking
- Persistent storage
- Declarative model

Container Orchestrators

- Kubernetes
- Red Hat OpenShift
- VMware Tanzu
- Azure Kubernetes Services (AKS)
- Google Kubernetes Engine (GKE)
- Amazon Elastic Container Service for Kubernetes (EKS)

Container Orchestration - Services



Demo!

- Deploying applications in Kubernetes

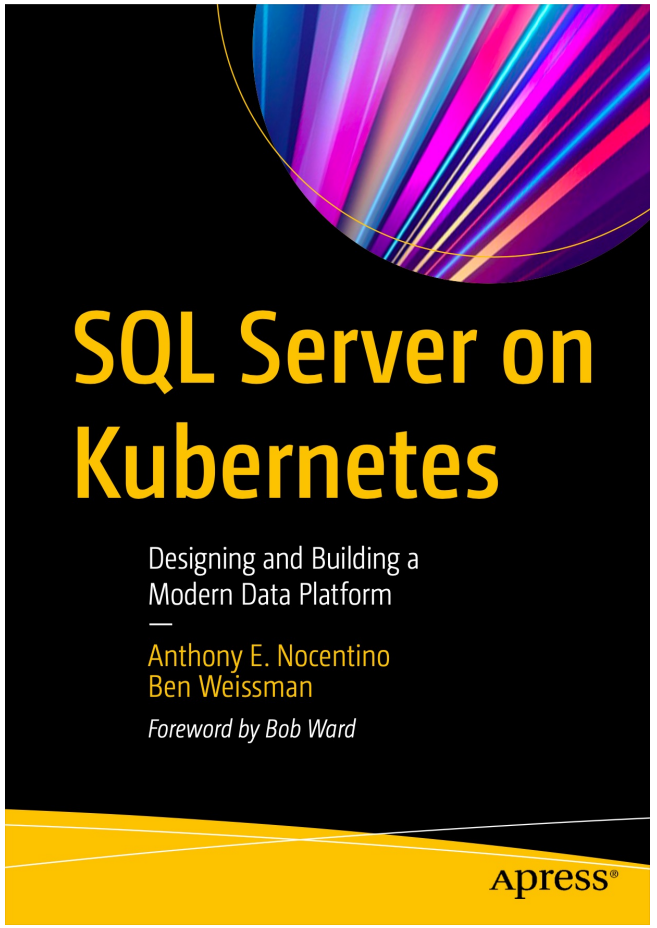
Review

- Introducing Containers
- Building and Running Apps in Containera
- The Container Universe
- Hands on with Containers
- Container Orchestration and Kubernetes

Need more data?

- **Contact me!**
  - **email:** anocentino@purestorage.com
  - **Twitter:** @nocentino
  - **GitHub:** https://github.com/nocentino/Presentations
    - (Look for Containers - What's Next?)
  - **Blog:** www.nocentino.com
  - **Book:** SQL Server on Kubernetes

- **Pluralsight**
  - Linux
  - Kubernetes
  - Azure
  - Hit me up to get free access to this content



Resources

- Installing Docker
  - https://docs.docker.com/desktop/windows/install/
  - https://docs.docker.com/engine/install/centos/
- Running Docker
  - https://docs.docker.com/get-started
  - https://docs.docker.com/storage