Containerizing .NET Core Applications with Microsoft Azure and AKS

Manoj Ganapathi

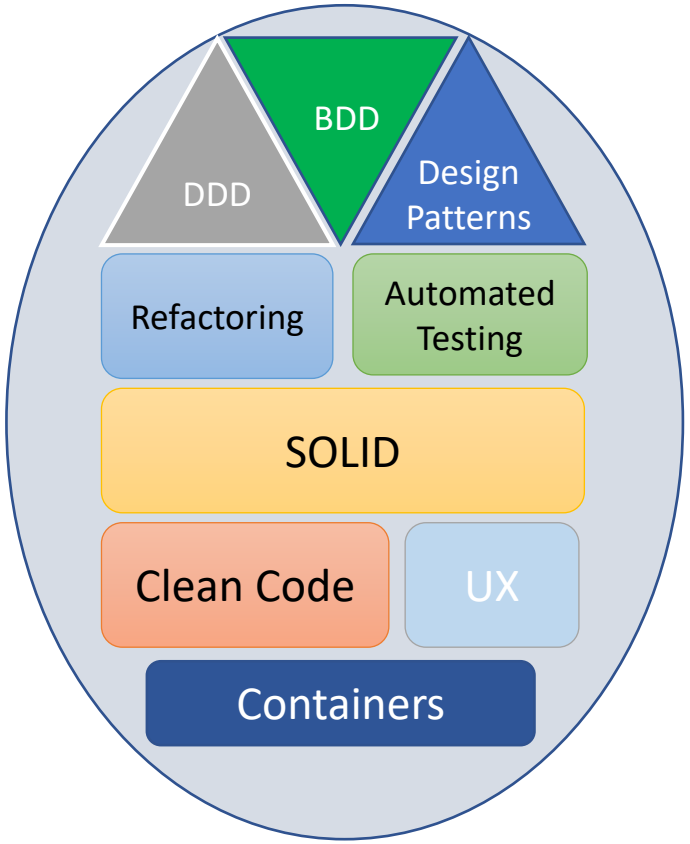
Chief Architect, CodeOps

About Me

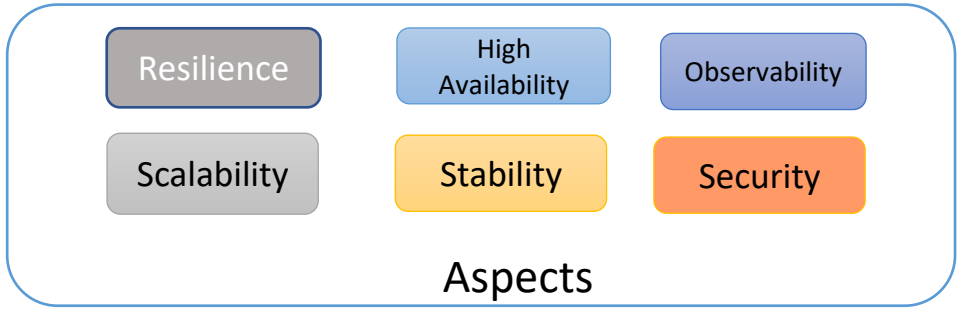
- Manoj is a seasoned IT professional with more than 20 years of experience. He has extensive experience in enterprise & solution architecture, design and implementation of large & complex enterprise systems. As an architect and technology consultant, he has consulted with several large, fortune 500 enterprises and worked with ISVs and startups. In his career, he has worked in multiple technology-oriented and leadership roles across all phases of software development life cycle. He is experienced in building and running technical communities and has been a speaker in several technology conferences.
- Over the last seven years, he has worked extensively on consulting, architecture and implementation of Cloud-based solutions, specializing on building highly scalable, resilient systems and DevOps practices.
- Currently, he is the Chief Architect at CodeOps Technologies (http://codeops.tech/) and a Digital Technology Consultant.
- LinkedIn profile: https://www.linkedin.com/in/manojg
- @manojgr, manoj@codeops.tech

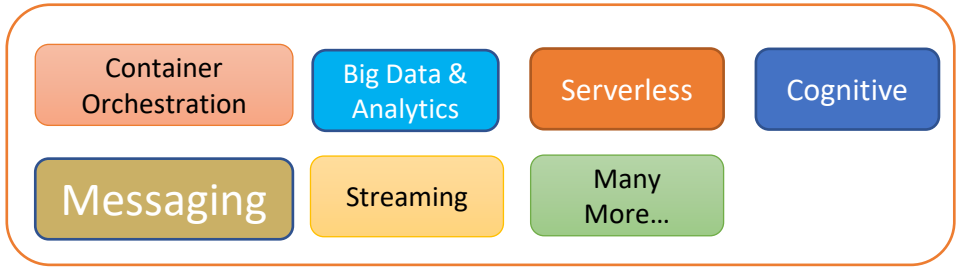
Perspective on Modern Software Delivery

MicroServices



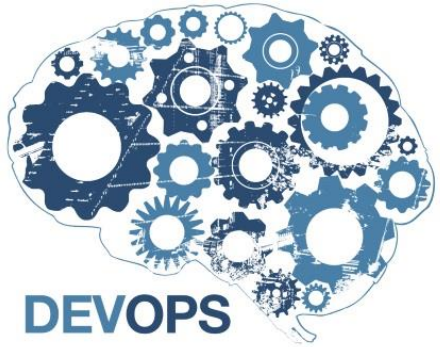
Cloud-Native

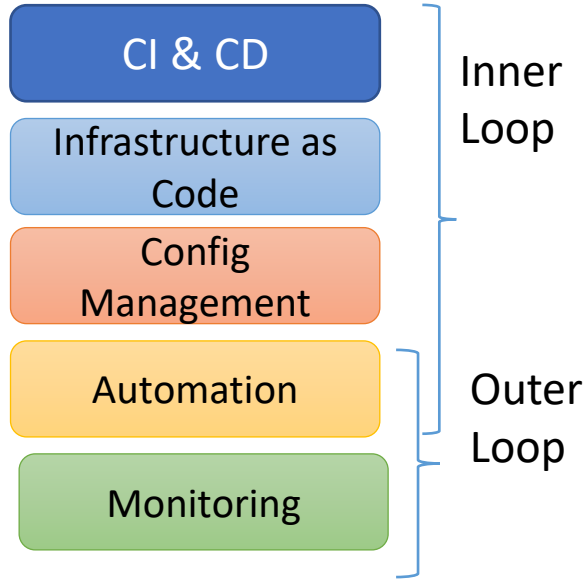




Cloud-Enabled Technology

DEVOPS





Design & Build → Deploy → Operate

Lean & Agility

Containers v/s VMs

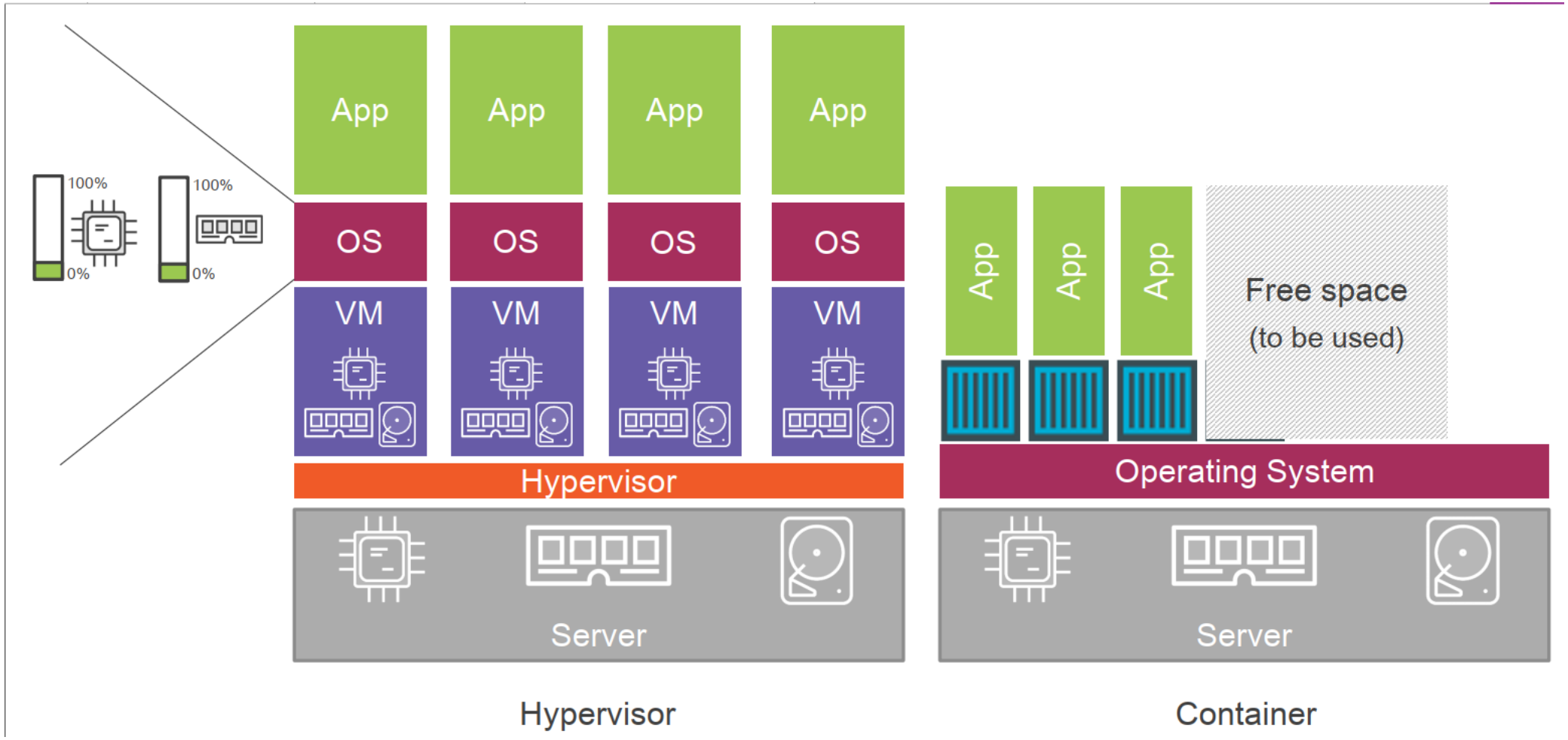
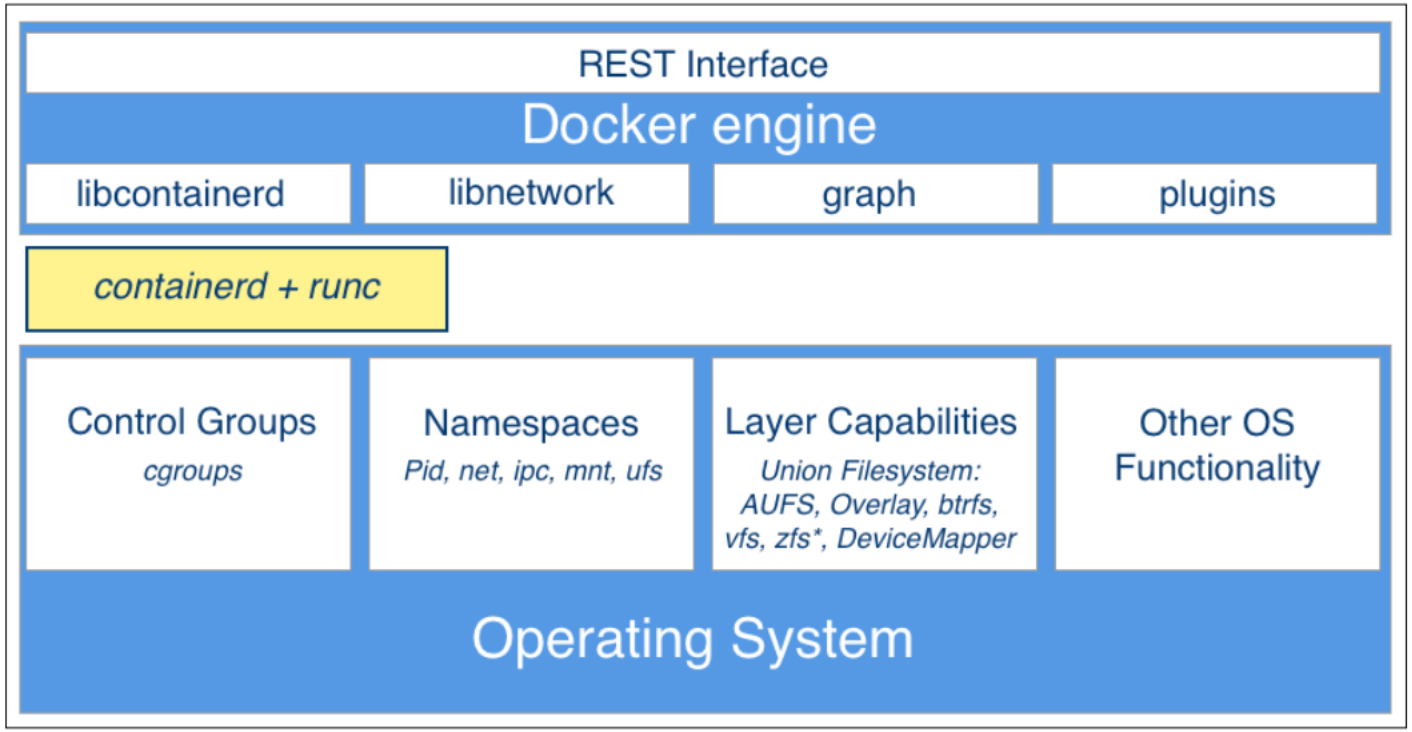


Image Ref: Nigel Poulton

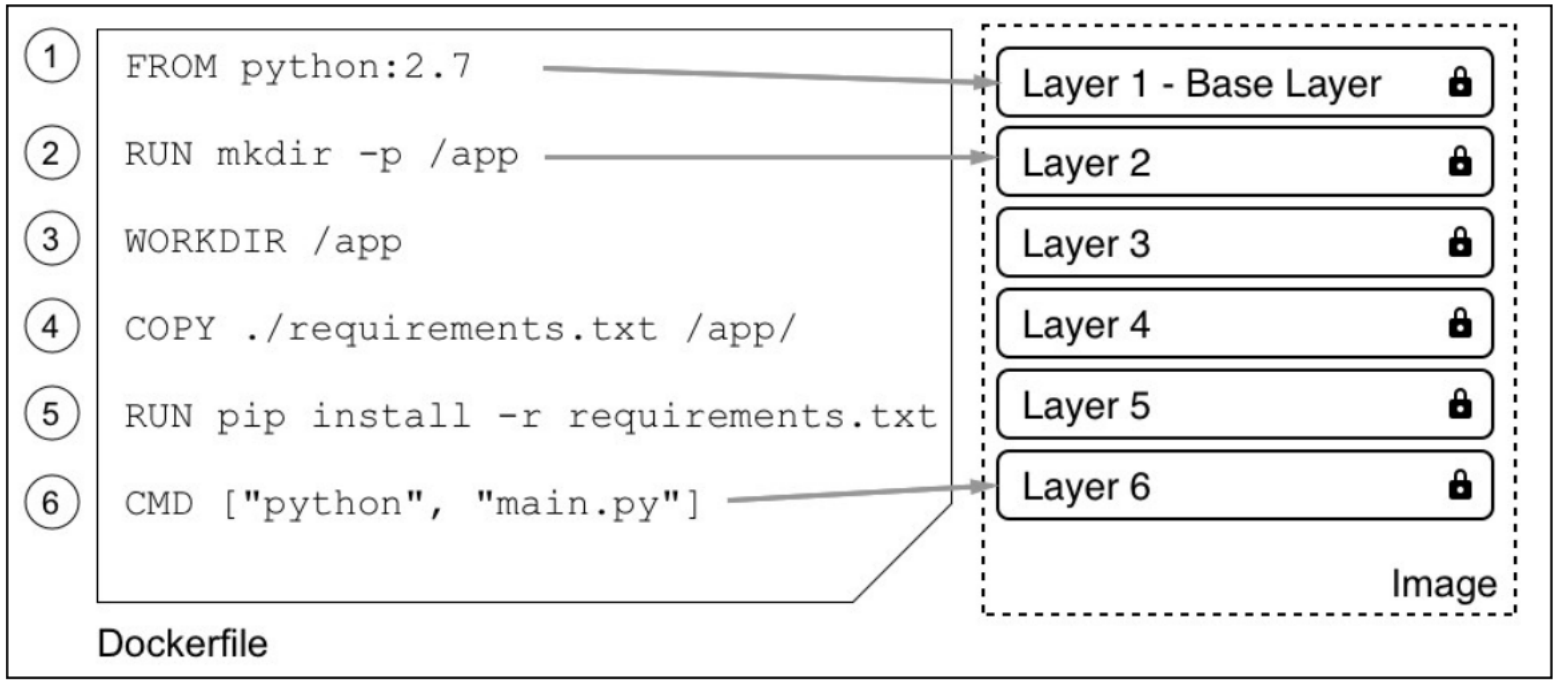
Docker Architecture



- Containers are encapsulated, secure processes running on the host platform
- Benefits
  - Security
  - Isolation
  - Standardized Infra

Ref: Containerize your apps with Docker & Kubernetes - book

Container Images & Dockerfile

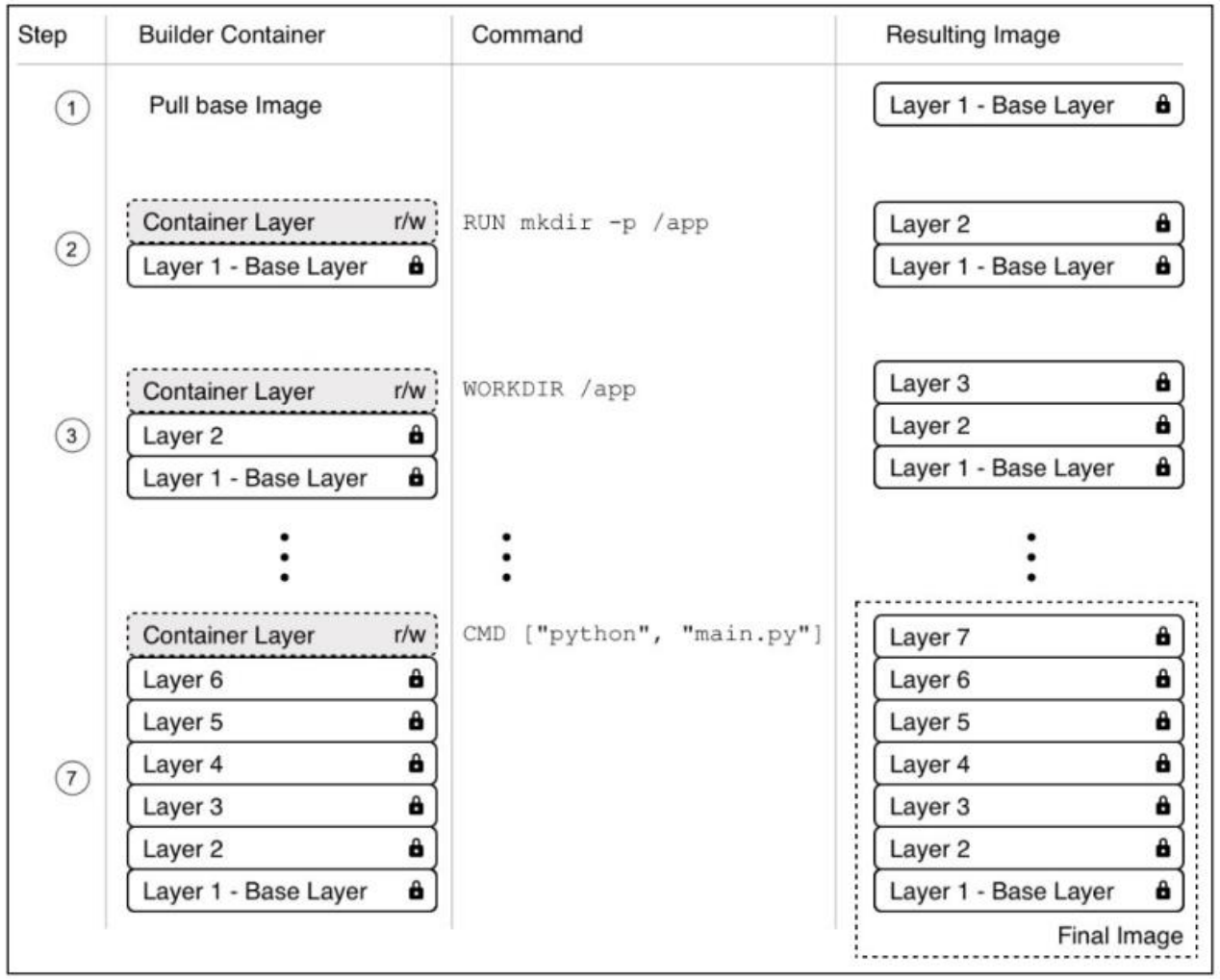


The relation of Dockerfile and layers in an image

- Templates for Containers
- Starts with a base layer, usually the OS
- All layers immutable, only top layer is writable

Ref: Containerize your apps with Docker & Kubernetes - book

Best Practices

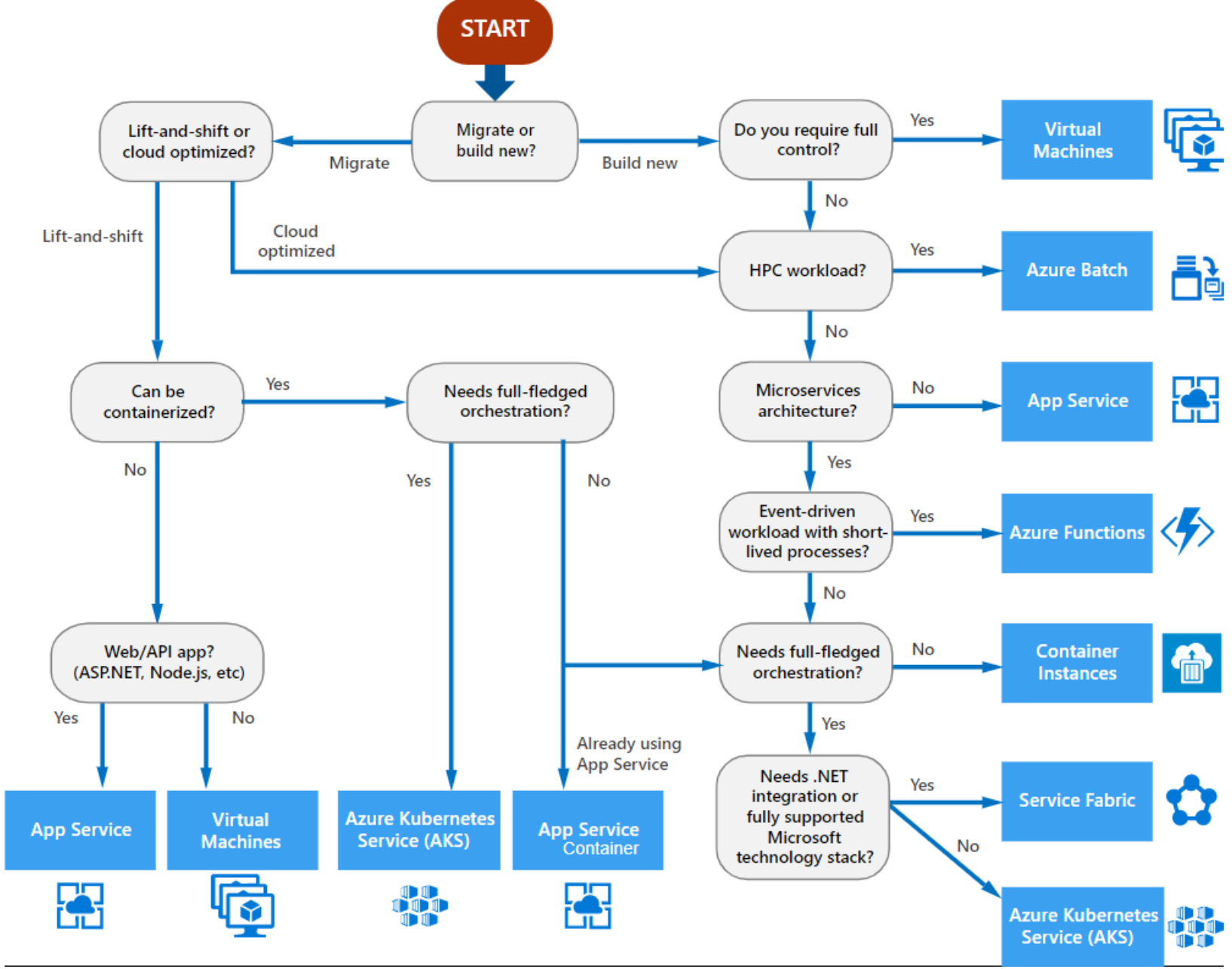


The image build process visualized

- Keep Containers ephemeral
- Order commands to leverage caching
- Avoid installing multiple packages
- Use .dockerignore
- Use multi-stage builds

Ref: Containerize your apps with Docker & Kubernetes - book

Choosing the right compute option



https://docs.microsoft.com/en-us/azure/architecture/guide/technology-choices/compute-decision-tree

Need for Container Orchestrators

VELOCITY

SCALING (SOFTWARE & TEAMS)

ABSTRACTING INFRASTRUCTURE

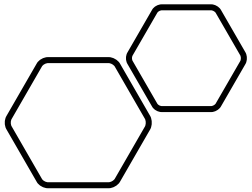
EFFICIENCY

Achieving Velocity

IMMUTABILITY

DECLARATIVE CONFIGURATION

SELF-HEALING



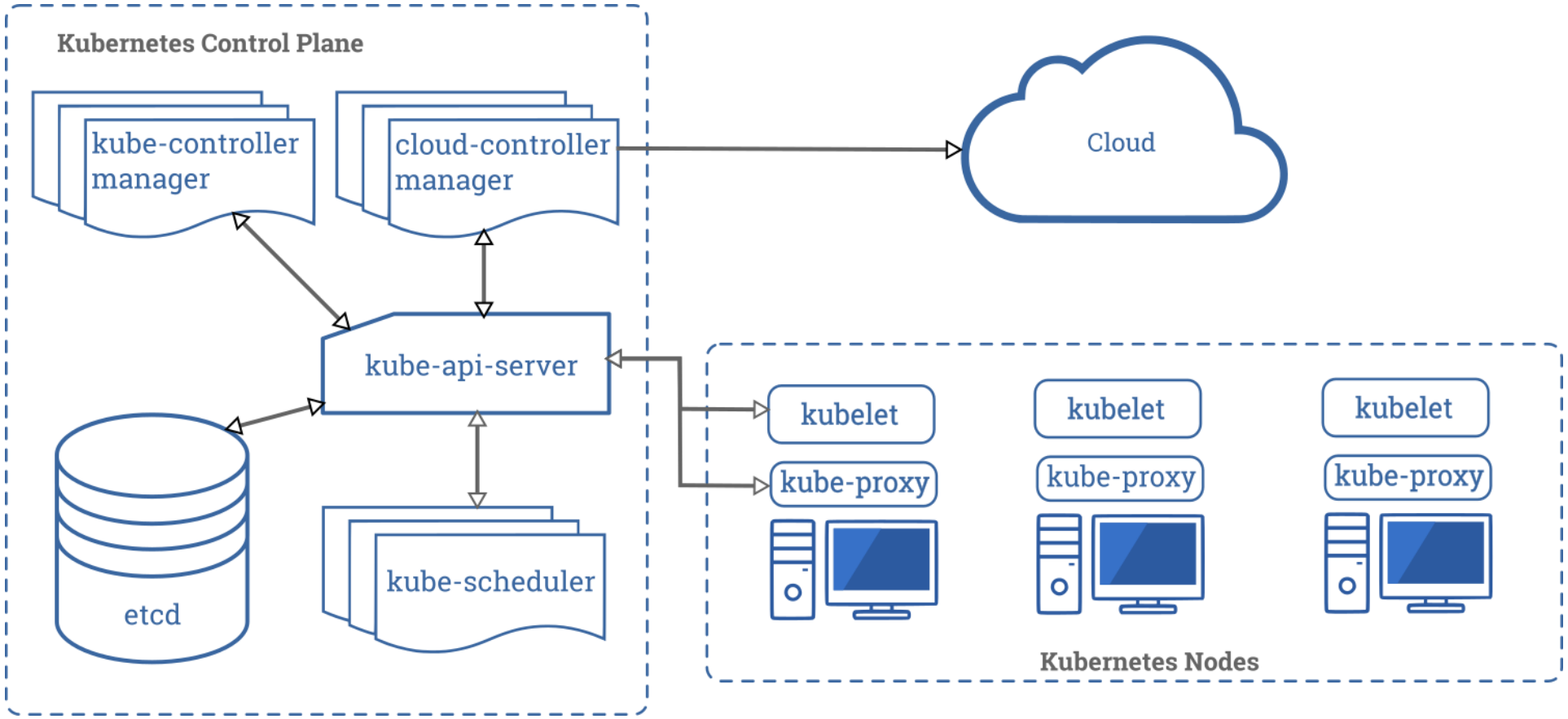
Other benefits

DECOUPLED ARCHITECTURE

HIGH UTILIZATION

EFFICIENCY

K8S Architecture



https://kubernetes.io/docs/concepts/overview/components/

Pod Lifecycle

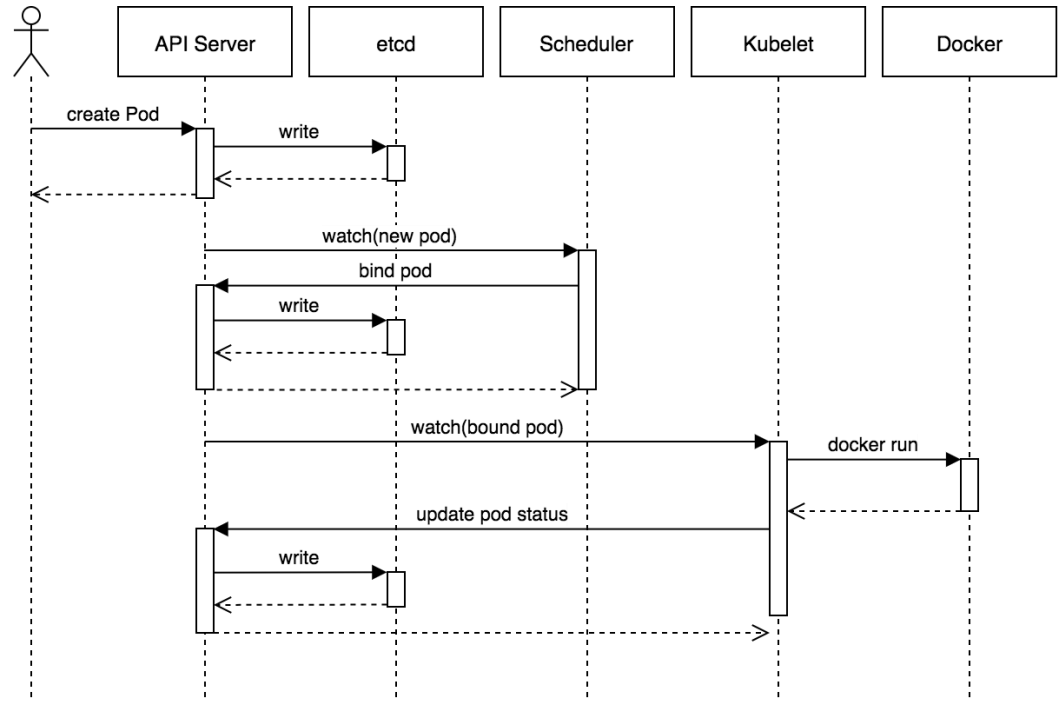


Image Ref: [Joe Beda's Blog]

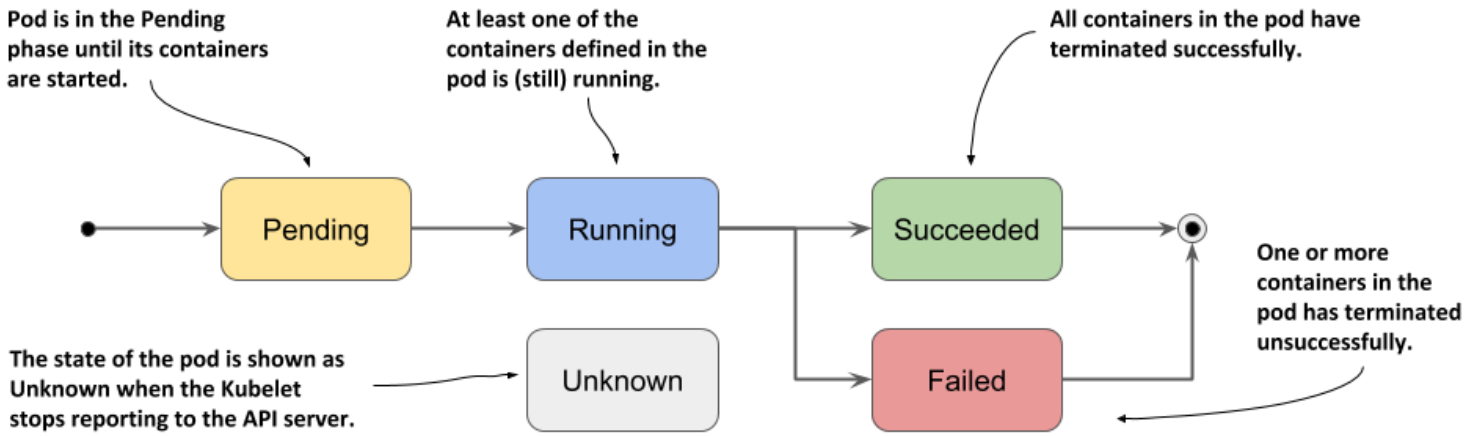


Image Ref: [Manning Books]

K8S objects

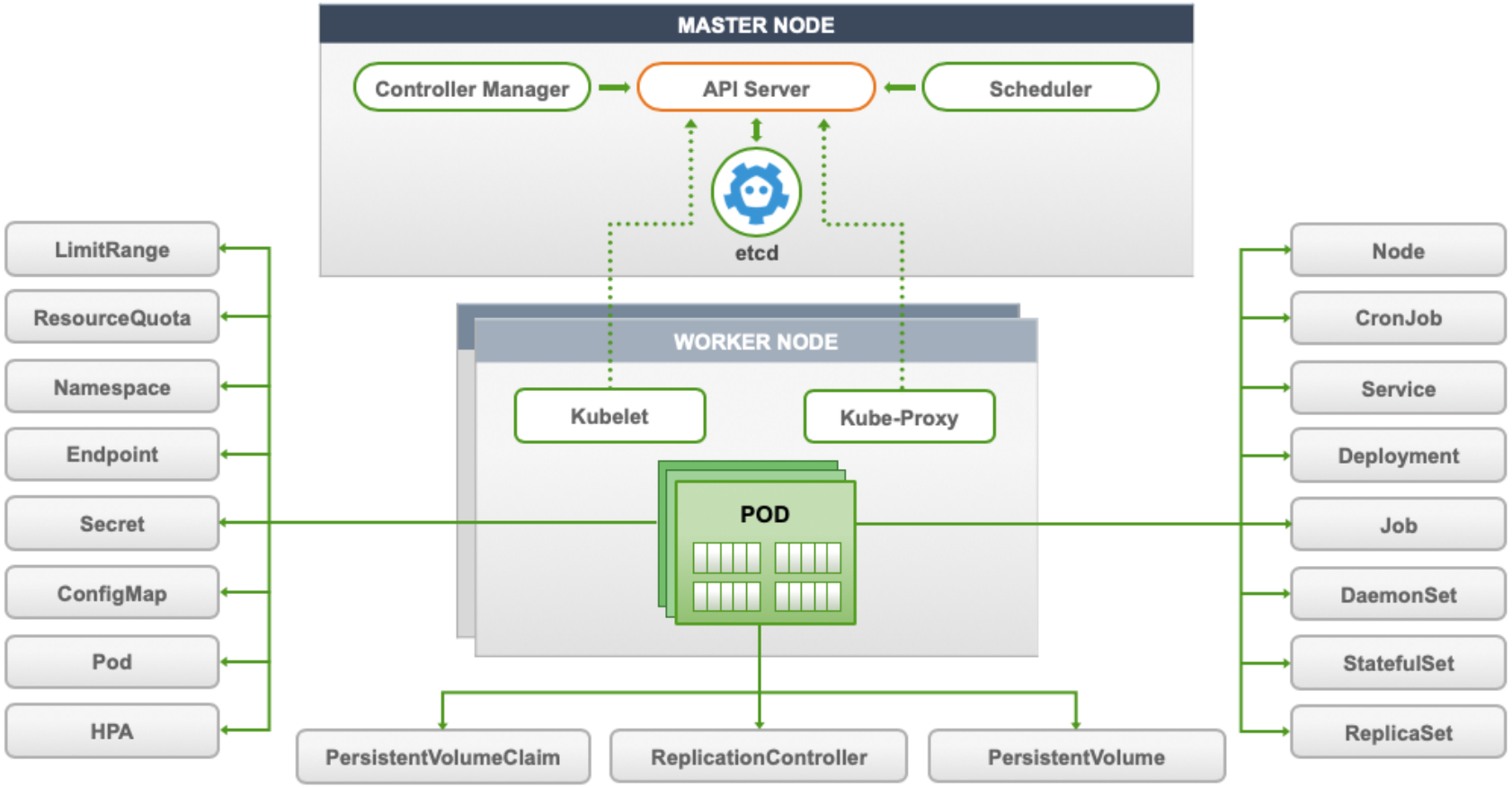
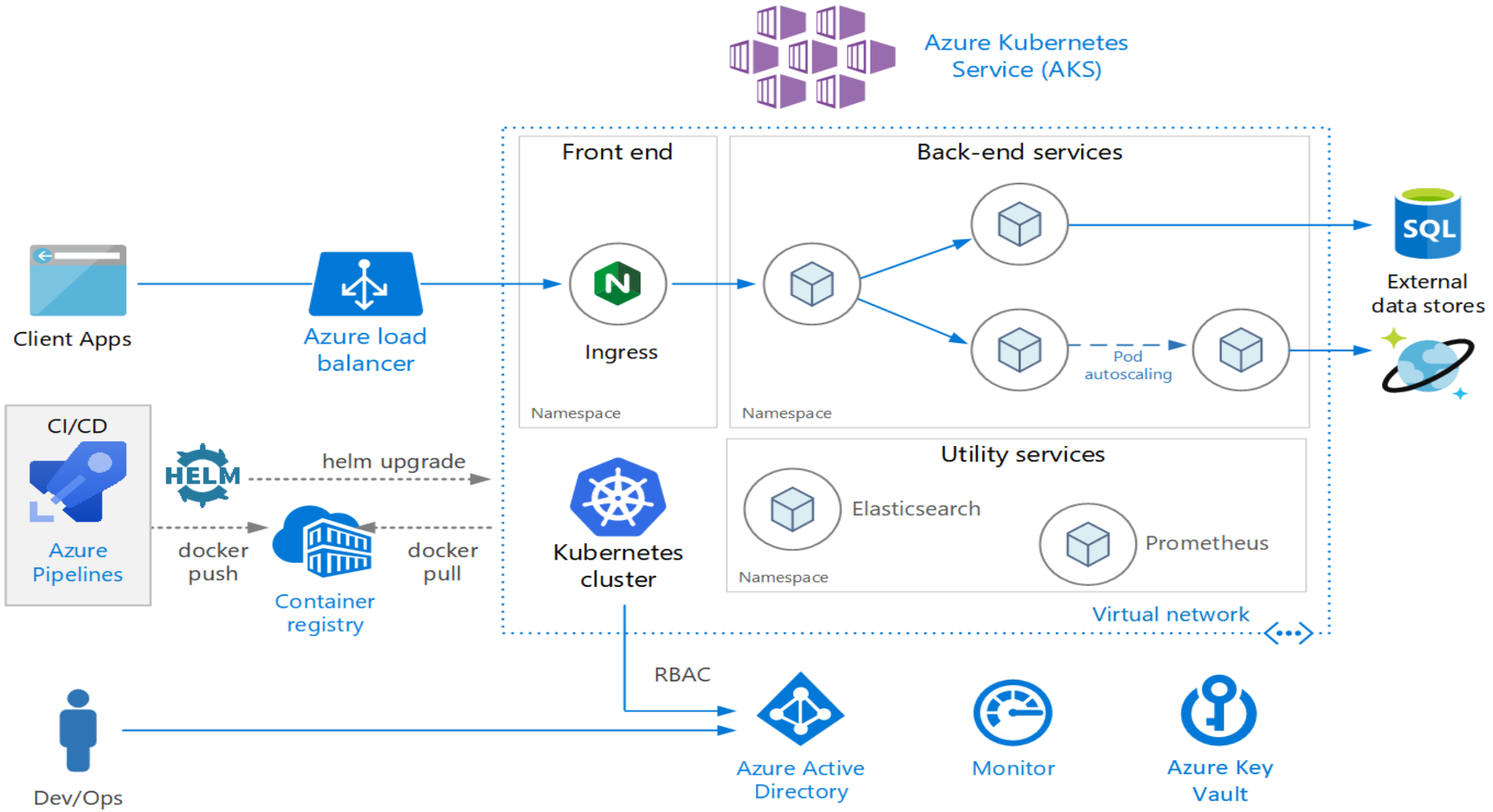
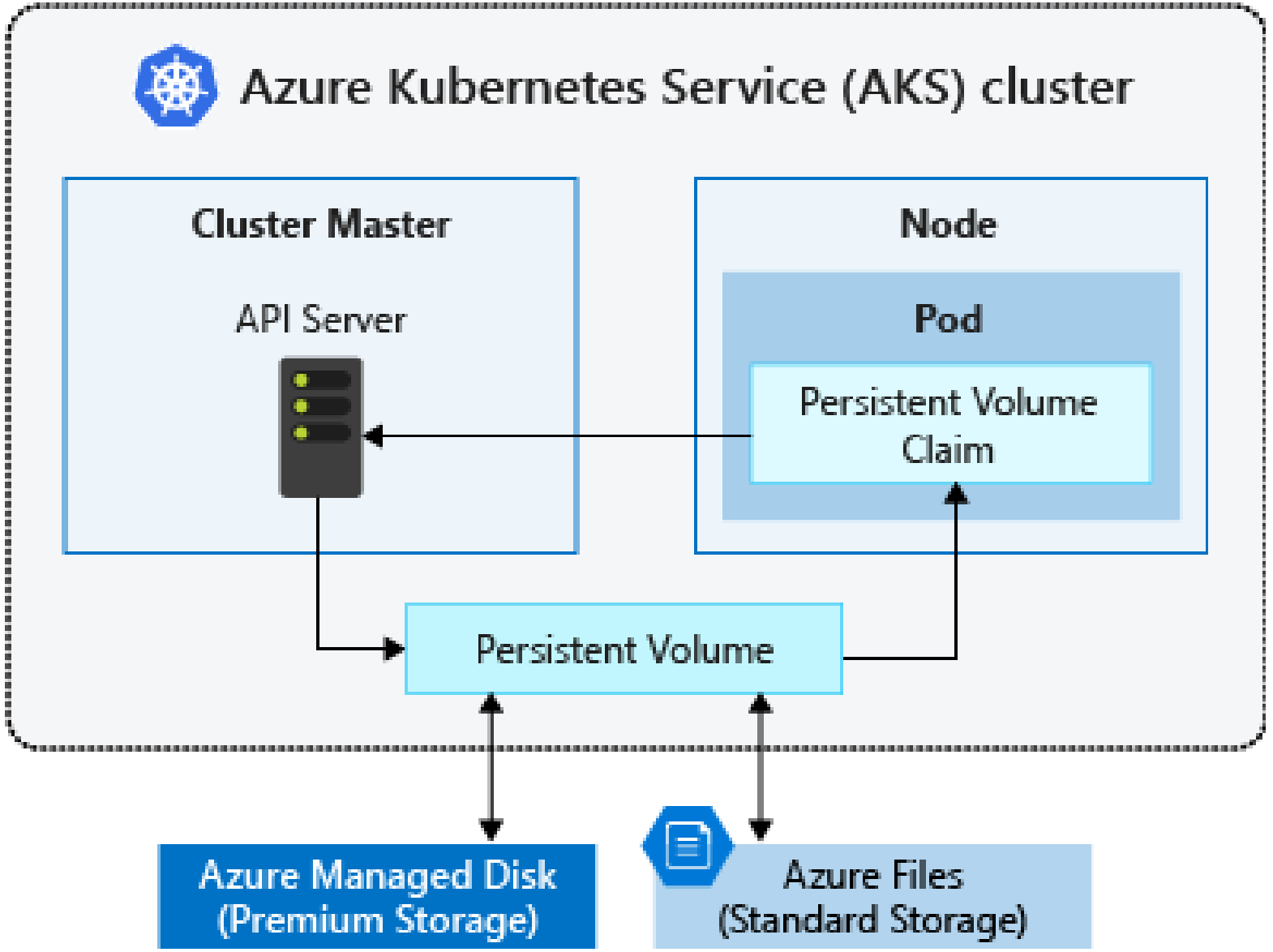


Image Ref: https://www.splunk.com/en_us/blog/it/monitoring-kubernetes.html

Microservices architecture on Azure Kubernetes Service (AKS)



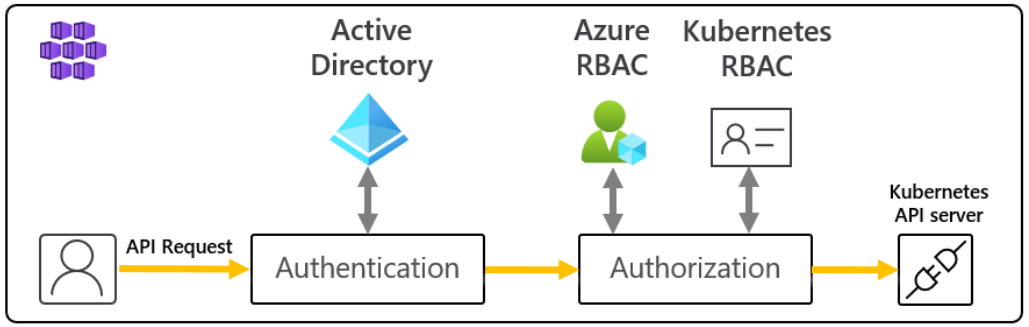
https://docs.microsoft.com/en-us/azure/architecture/reference-architectures/containers/aks-microservices/aks-microservices

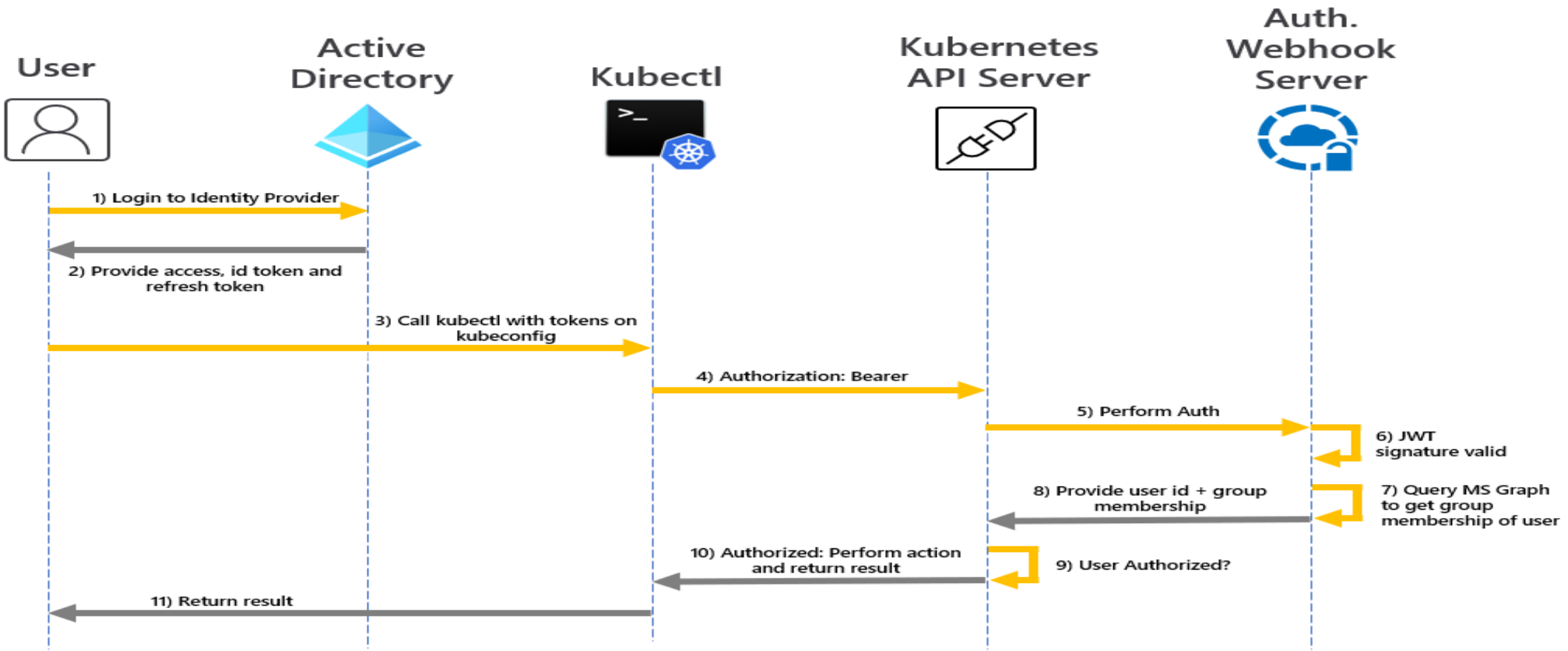


Storage in K8S/AKS

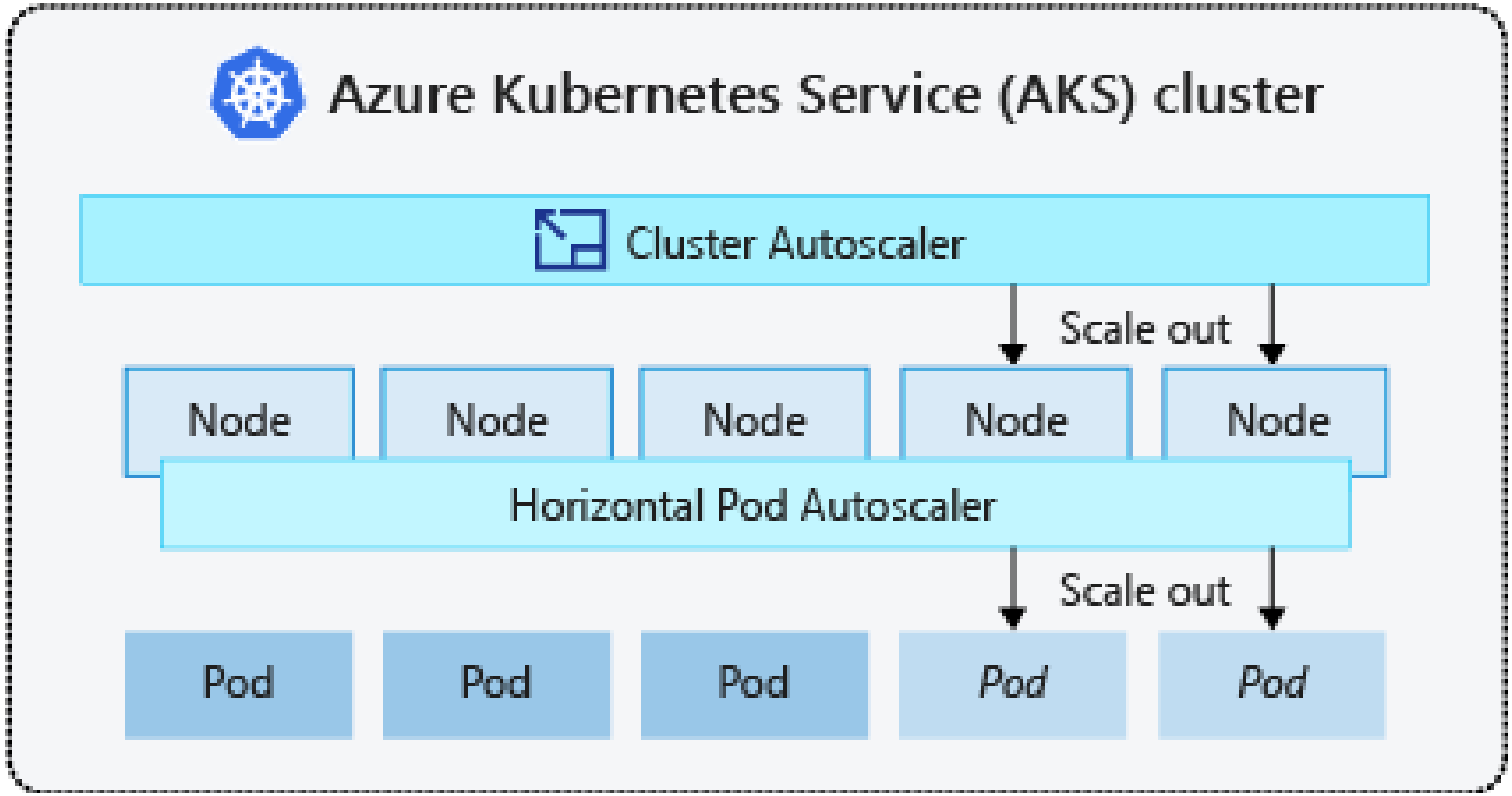
https://docs.microsoft.com/en-us/azure/aks/concepts-storage

Authentication & Authorization





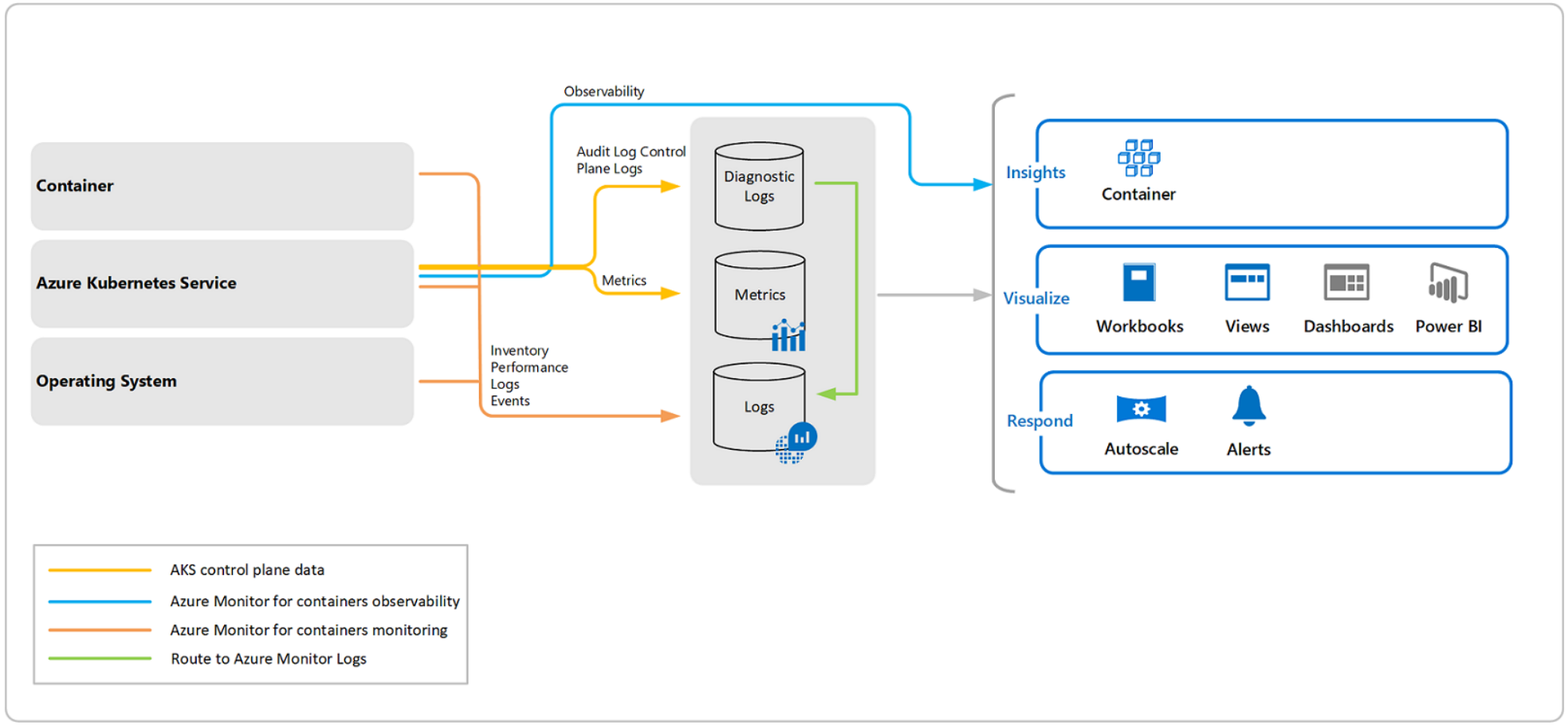
https://docs.microsoft.com/en-us/azure/aks/concepts-identity



Scaling options for applications in Azure Kubernetes Service (AKS)

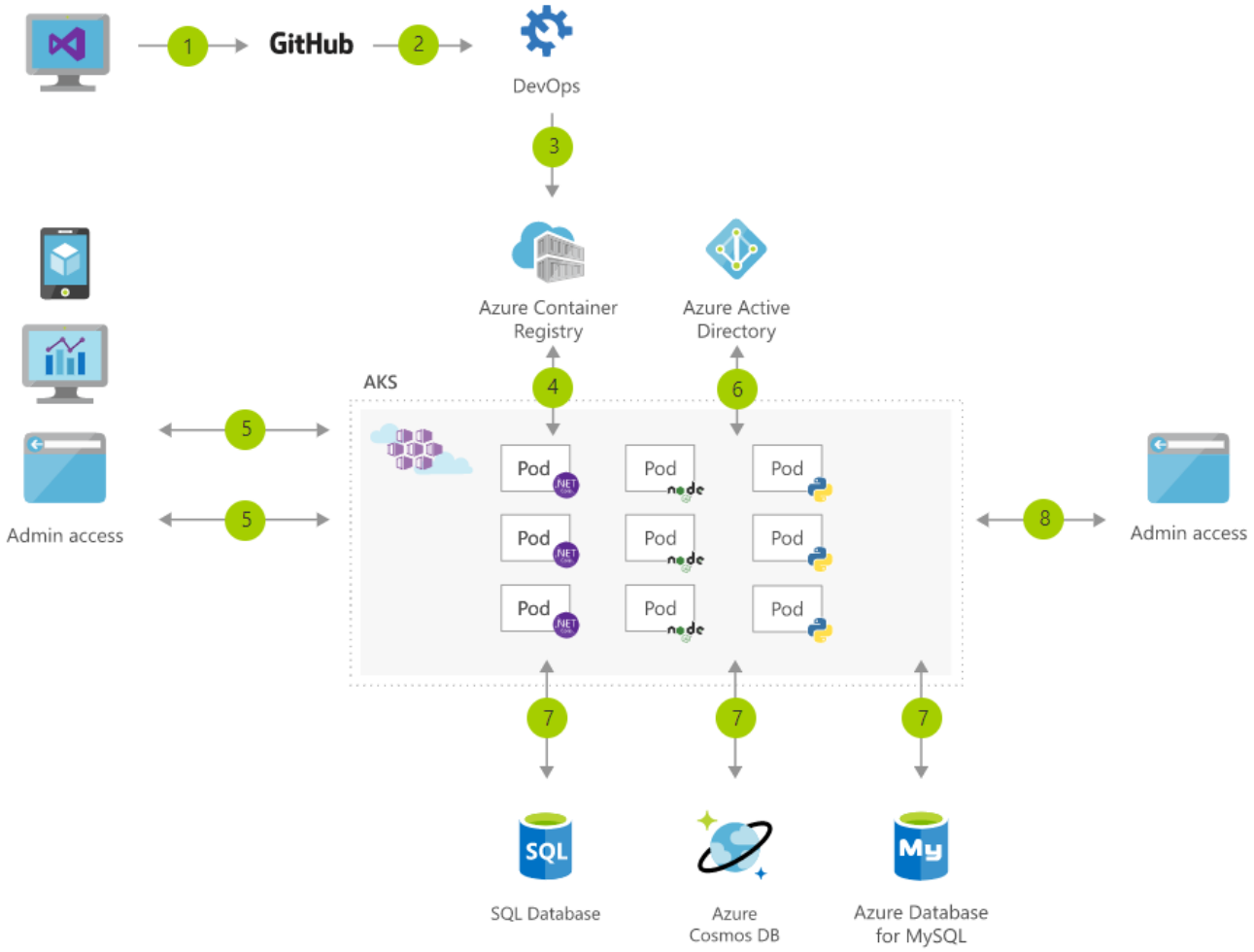
https://docs.microsoft.com/en-us/azure/aks/concepts-scale

Monitoring in AKS (Azure Monitor)



https://docs.microsoft.com/en-us/azure/azure-monitor/insights/container-insights-overview

Deploying with AKS (DevOps)



1. Use an IDE, such as Visual Studio, to commit changes to GitHub.
2. GitHub triggers a new build on Azure DevOps
3. Azure DevOps packages microservices as containers and pushes them to the Azure Container Registry
4. Containers are deployed to AKS cluster
5. Azure Active Directory is used to secure access to the resources
6. Users access services via apps and websites
7. Administrators access the apps via a separate admin portal
8. Microservices use databases to store and retrieve information

https://azure.microsoft.com/en-in/solutions/architecture/microservices-with-aks/

Resources

- K8s Learning Path
- Docker on Azure -
- Containerize your apps with Docker & Kubernetes
- Learning path for Containers & Kubernetes on MS Learn
- Git Hub Repo (Labs)
- K8S Cheat Sheet
- AKS Best Practices
- Kubernetes in the Cloud Adoption Framework - Cloud Adoption Framework