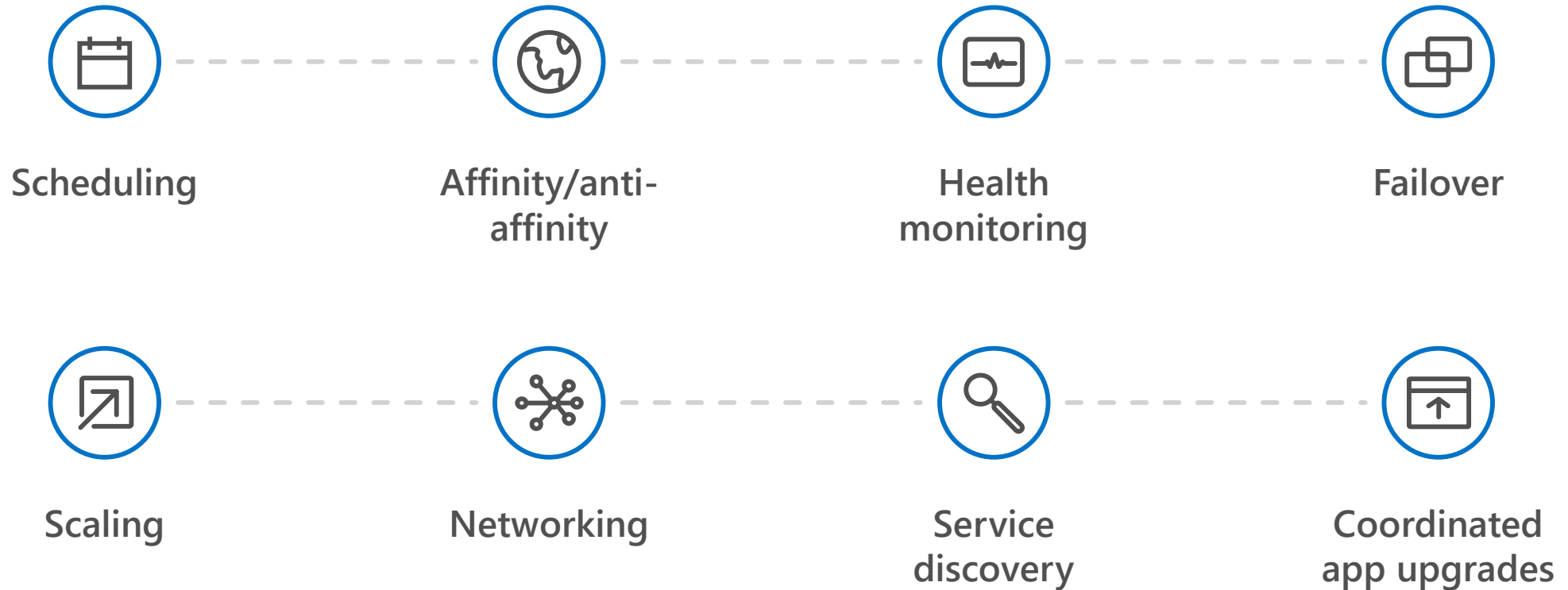


If we really want to run lots of containers



The elements of **orchestration**



Kubernetes: the industry leading orchestrator



Portable

Public, private, hybrid,
multi-cloud

Extensible

Modular, pluggable,
hookable, composable

Self-healing

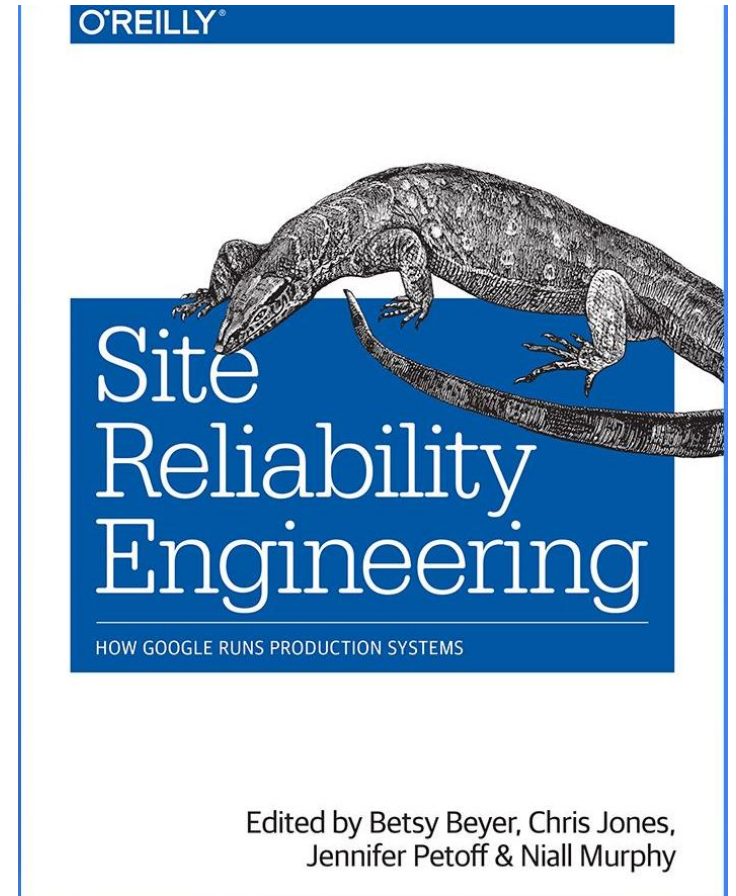
Auto-placement, auto-restart,
auto-replication, auto-scaling

History of Kubernetes, part 1

- 1979 chroot
- 1982 BSD
- 2000 FreeBSD Jails
- 2005 Solaris Zones/Containers
- 2006-2007 Linux Cgroups and namespaces
- 2006 AWS begins selling VMs/IaaS
- 2008 Brendan Burns joins Googles Websearch infrastructure team

Short intermission, Google Websearch

- No SSH (after a while)
- No Imperative config (No changing running systems)
- Errors very costly
- Borg developed to handle this, consists of two parts:
 - Global Work Queue
 - Babysitter
- Growth, growth & growth
- Utilization becomes big problem



History of Kubernetes, part 2

- Google websearch leaves Seattle, people moved to IaaS
- Ex-websearches want create Borg for IaaS, but difficult:
 - Declarative config
 - Server side deployment
 - Health maintenance
- But Docker (2013) becomes popular, standardizes:
 - Image format and build tools
 - Image distribution
 - Container runtime
- Kubernetes project is started !

Children's Guide to Kubernetes:

<https://azure.microsoft.com/en-us/resources/videos/the-illustrated-children-s-guide-to-kubernetes/>



Kubernetes summary

- Nodes

- The vm's you run kubernetes on. At least one has to have master role (Won't see the master)

- Pods

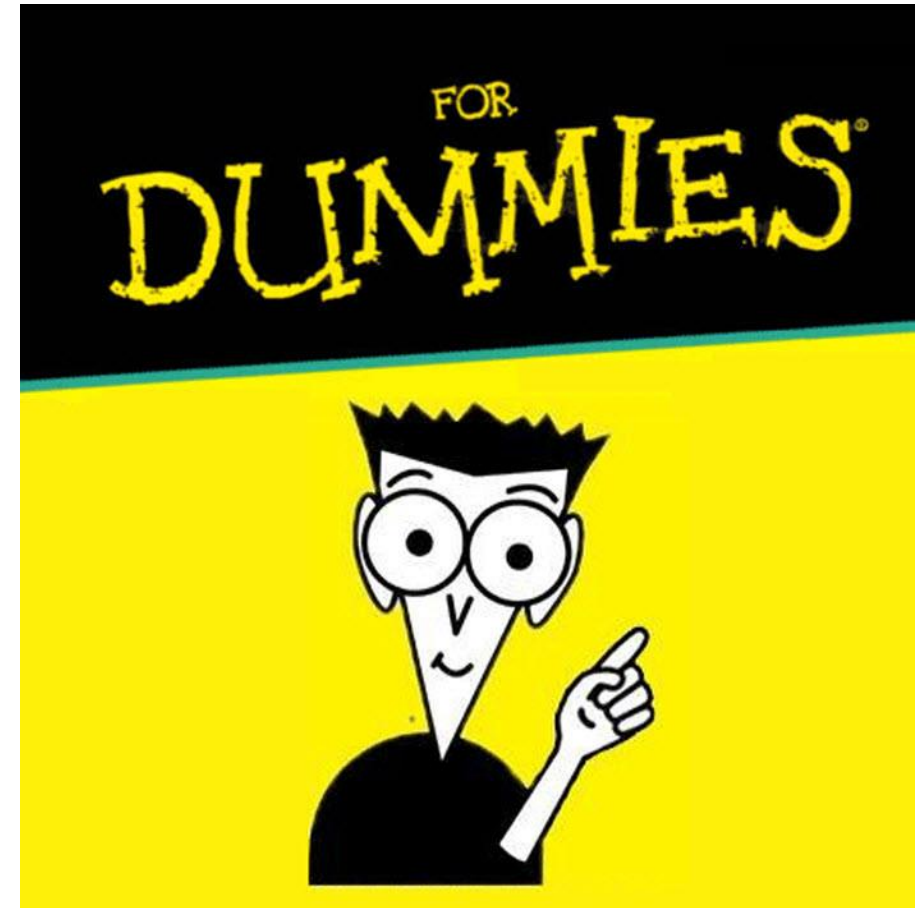
- The scheduling unit of k8s. One or many containers that share an ip

- Deployments

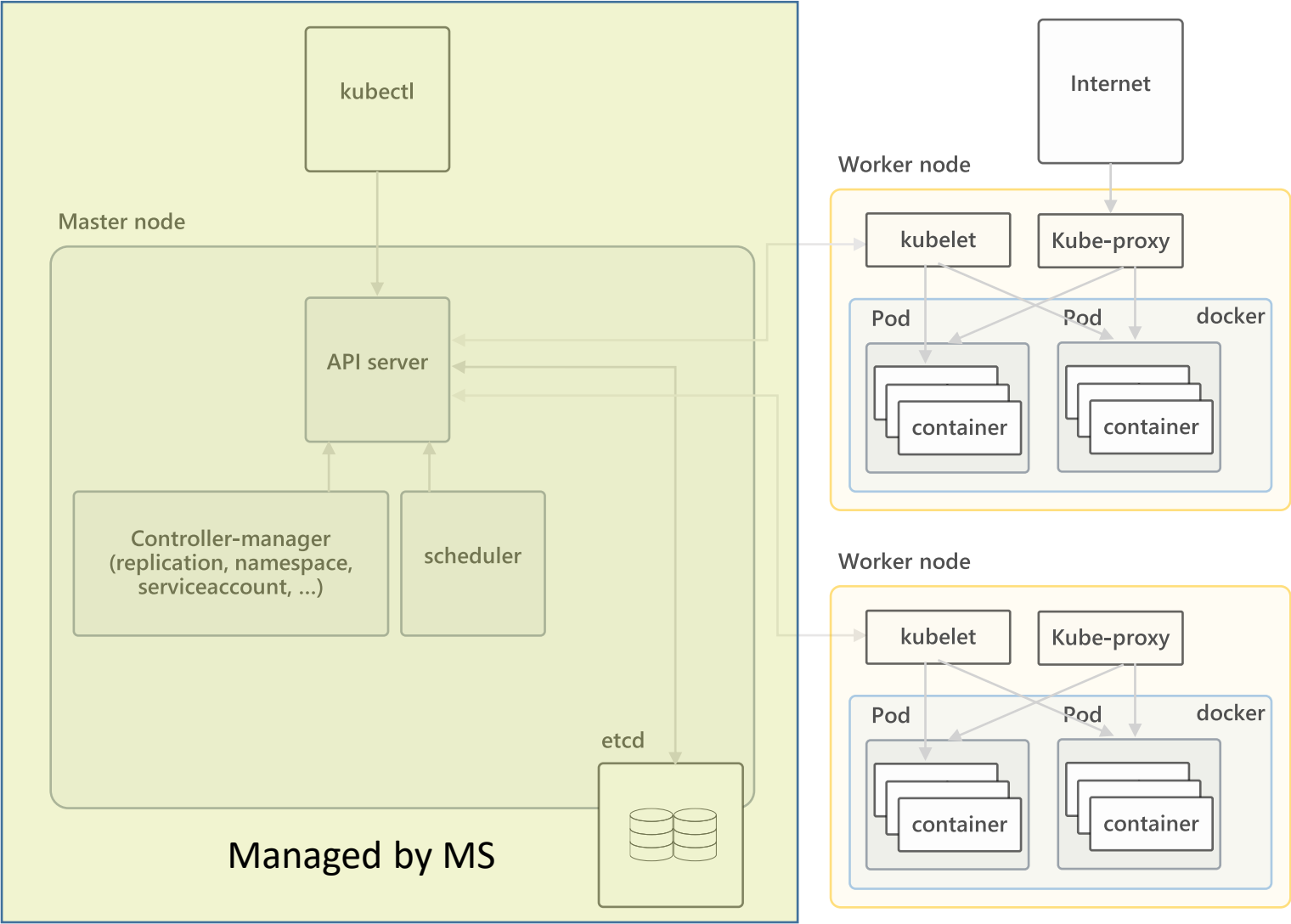
- A set containers with characteristics, like scale, version etc.

- Services

- A method of providing access to pods wherever they move



Kubernetes Architecture



How Azure Kubernetes Service (AKS) works

Automated upgrades, patches
High reliability and availability
Easy and secure cluster scaling
Self-healing
API server monitoring
Control plane at no charge

