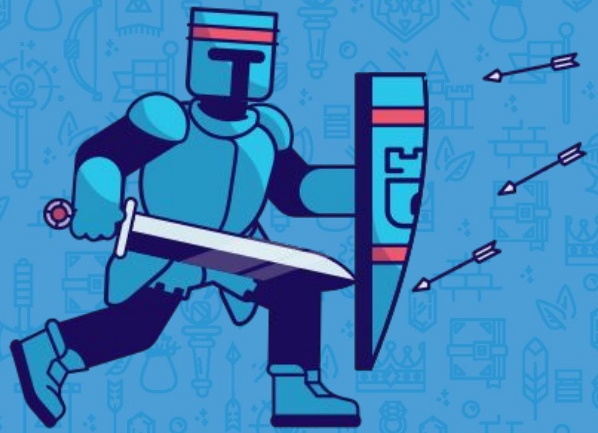
Rook Project

Travis Nielsen, Red Hat
Jared Watts, Upbound
Alexander Trost, Cloudical
Rook Maintainers

https://rook.io/
https://github.com/rook/rook



Agenda

- Introduction to Rook
- Architectural Overview
- Roadmap
- CNCF Project Graduation
- Storage Provider Deep Dives

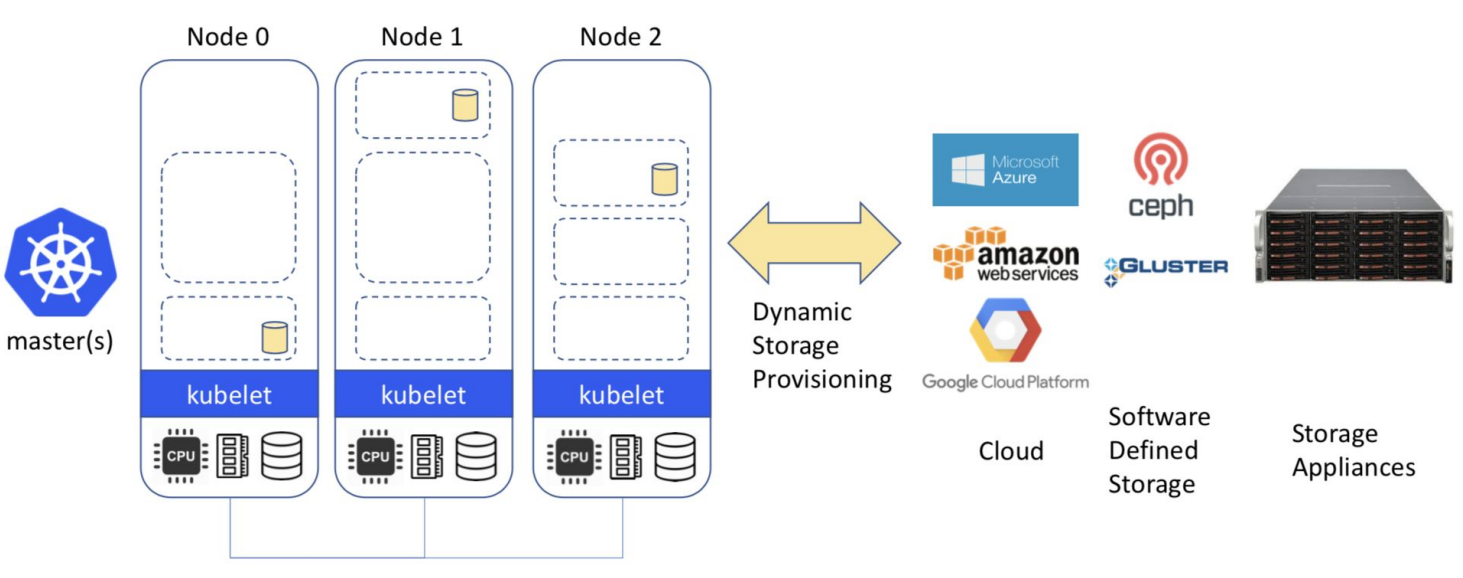
Schedule

Length	Presenter(s)	Description
15	Travis/Alexander	Intro to Rook
5	Jared	Path to Graduation
20	Sebastien	Deep Dive: Ceph
20	Ilya	Deep Dive: EdgeFS
10	Sid	Deep Dive: YugabyteDB
10	Yannis	Deep Dive: Cassandra

Kubernetes & Storage

Storage for Kubernetes

- Volume plugins allow external storage solutions to provide storage to your apps



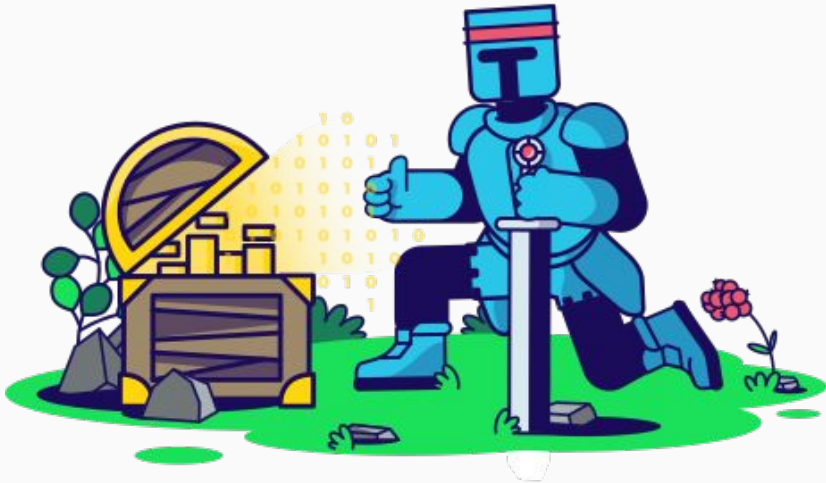
Storage Challenges

- Reliance on external storage
  - Not portable
  - Requires these services to be accessible
  - Deployment burden
- Reliance on cloud provider managed services
  - Vendor lock-in
- Day 2 operations - who is managing the storage?

What Is Rook?

What is Rook?

- Storage Operators for Kubernetes
  - Wherever K8s runs
- Automate Management
  - Deployment
  - Configuration
  - Upgrading

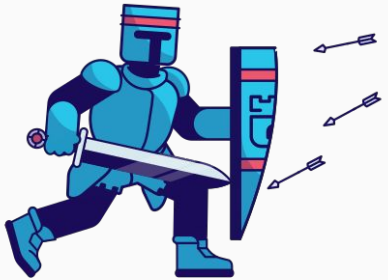


What is Rook?

- Open Source (Apache 2.0)
- Cloud-Native Computing Foundation (CNCF)
  - Incubation Project
- Extends Kubernetes with Operators and custom types
- Framework for many storage providers and solutions

Storage Providers

Storage Provider	Status	Joined Rook
Ceph	**Stable**	v0.1
CockroachDB	Alpha	v0.8
Minio	Alpha	v0.8
EdgeFS	**Stable**	v0.9
Cassandra	Alpha	v0.9
NFS	Alpha	v0.9
YugabyteDB	Alpha	v1.1
Apache Ozone	Alpha	v1.2*

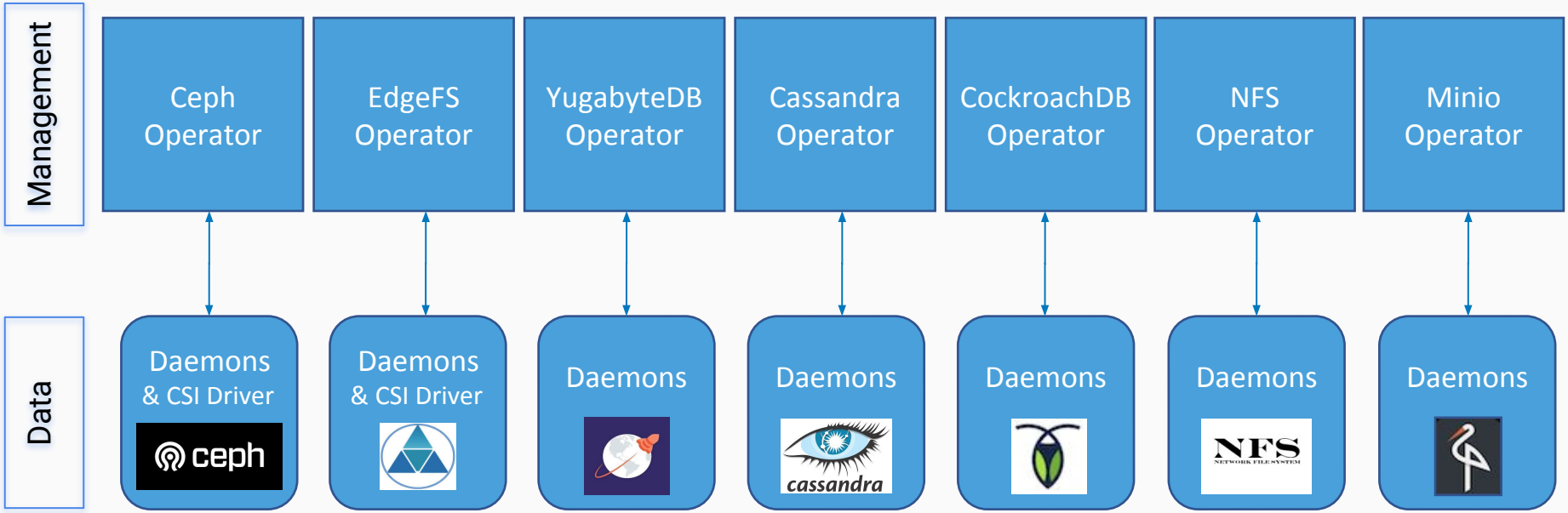


Architecture

Architectural Layers

- Orchestration
  - The operator owns the **management** of the storage provider
- Storage Provisioning
  - CSI driver **connects** client pods to the storage
- **Data** layer: Storage Provider
  - Block/File/Object storage or databases

Rook Architecture (Runtime)



Rook Operators

- Deploy and manage a storage platform
  - Automates actions a human would normally do
- Defines *desired state* for the storage resource
  - Storage Cluster, Filesystem, Object Store, etc.
- The Operator runs reconciliation loops
  - Watches for changes in desired state
  - Watches for changes in the cluster
  - Applies changes to the cluster to make it match desired

Rook Operators

- The Operators leverages the full power of K8S
  - Services, ReplicaSets, DaemonSets, Secrets, …
- Manage storage systems at scale
  - Stateful upgrades
  - Health and monitoring tasks
- Not on the data path – can be offline for minutes

Custom Resource Definitions (CRDs)

- Teaches Kubernetes about new first-class objects
- Custom Resource Definition (CRDs) are arbitrary types that extend the Kubernetes API
  - look just like any other built-in object (e.g. Pod)
  - Enabled native `kubectl` experience
- A means for user to describe their desired state

Framework for Storage Solutions

- Rook is more than just a collection of Operators and CRDs
- **Framework** for storage providers to integrate their solutions into cloud-native environments
  - Storage resource normalization
  - Operator patterns/plumbing
  - Common policies, specs, logic
  - Build/CI/Testing tools
- Community: Slack & GitHub

Storage Configuration Sequence

- Admin configures storage for K8s
  - Installs the storage provider (Rook operator/CRDs)
  - Creates storage class(es)
- Application requests storage with a PVC
  - Storage is mounted into the pod

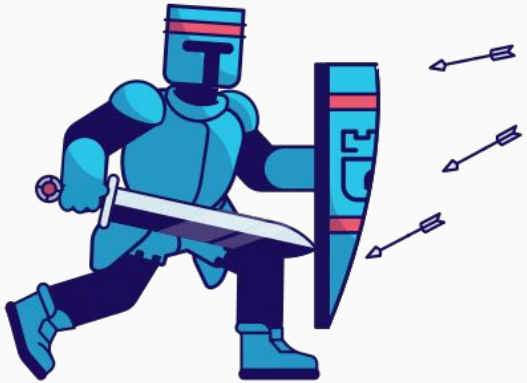
Roadmap

- Storage providers
  - Owner and community driven
  - Features depend on the storage provider
- Define API for storage operators

Getting Started with Rook

Website	https://rook.io
Documentation	https://rook.io/docs/rook/v1.1/
Blog	https://blog.rook.io/
Install v1.1	https://github.com/rook/rook/releases/

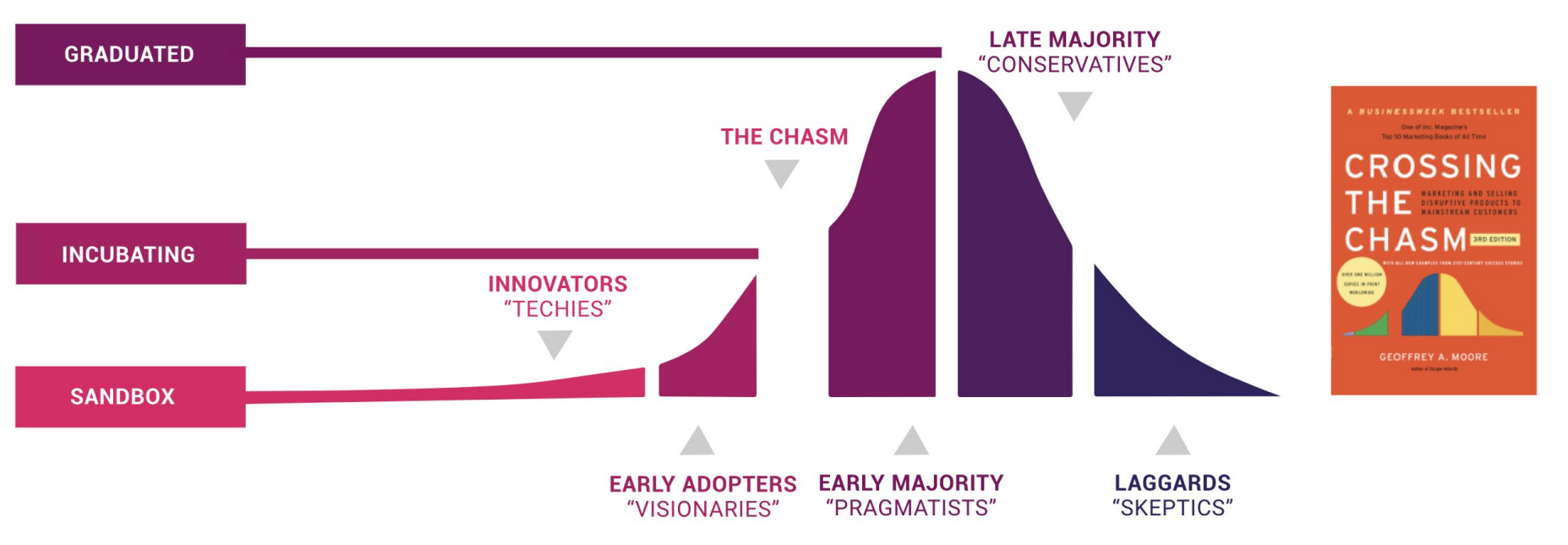
How to get involved?



- Contribute to Rook, review issues and PRs
  - https://github.com/rook/rook
- Slack - https://rook-io.slack.com/
  - #conferences
- Twitter - @rook_io
- Community Meetings
- Forums: https://groups.google.com/forum/#!forum/rook-dev

CNCF Project Graduation

CNCF Project Maturity Lifecycle

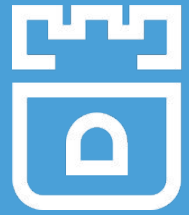


Courtesy https://www.cncf.io/projects/

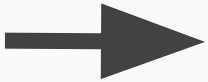
Rook Project Maturity

- **(Jan ‘18) Sandbox** - “Innovators”
  - Nurture early stage projects
  - Promote visibility and alignment with other CNCF projects
- **(Sept ‘18) Incubating** - “Early Adopters”
  - Starting to show early production adoption
  - Ceph orchestrator → Storage Orchestration Framework
- **(March ‘20) Graduation** - “Early Majority”
  - Sustainability and ready for more conservative adopters
  - Kubernetes, Prometheus, Envoy, CoreDNS, Jaeger, Vitess, containerd, Fluentd

Stats Since Incubating



- 13.3M+ container downloads
- 3,265+ GitHub stars
- 17 releases
- 85+ contributors
- 1,928 commits
- 390+ forks
- 1,810+ Twitter followers
- 800+ Slack members



- 96M+ container downloads
- 6,400+ Github Stars
- 40 releases
- 210+ Contributors
- 3,900+ commits
- 1,200+ forks
- 4,000+ Twitter followers
- 2,500+ Slack members

Security Audit and Process

- Independent third party security review
  - Performed by Trail of Bits https://www.trailofbits.com/, team behind core Kubernetes security audit
  - Dec 2nd kickoff
- Any critical vulnerabilities found must be addressed
- Security disclosure process defined
- Core Infrastructure Initiative (CII) badge
  - https://bestpractices.coreinfrastructure.org/en/projects/1599#
  - Currently at 80%

Call to Action: Production Testimonials

- We need your input on adoption of Rook in production deployments!
  - Size of cluster (nodes, storage, users), how long has Rook been in production for you, what problems has Rook helped you solve, etc.
- Link to survey will be posted on Slack, Twitter, README.md
  - DM Jared Watts (@jbw976) any time
- Happy to include your company in Adopters.md
- Confidential **if needed**

Storage Provider Deep Dives

Thank you!

https://github.com/rook/rook

https://rook.io/