Kubernetes, CNCF and Beyond

邓德源 才云科技

Agenda

- Look at how Kubernetes and CNCF runs the community, project update, etc
- There is no second agenda item

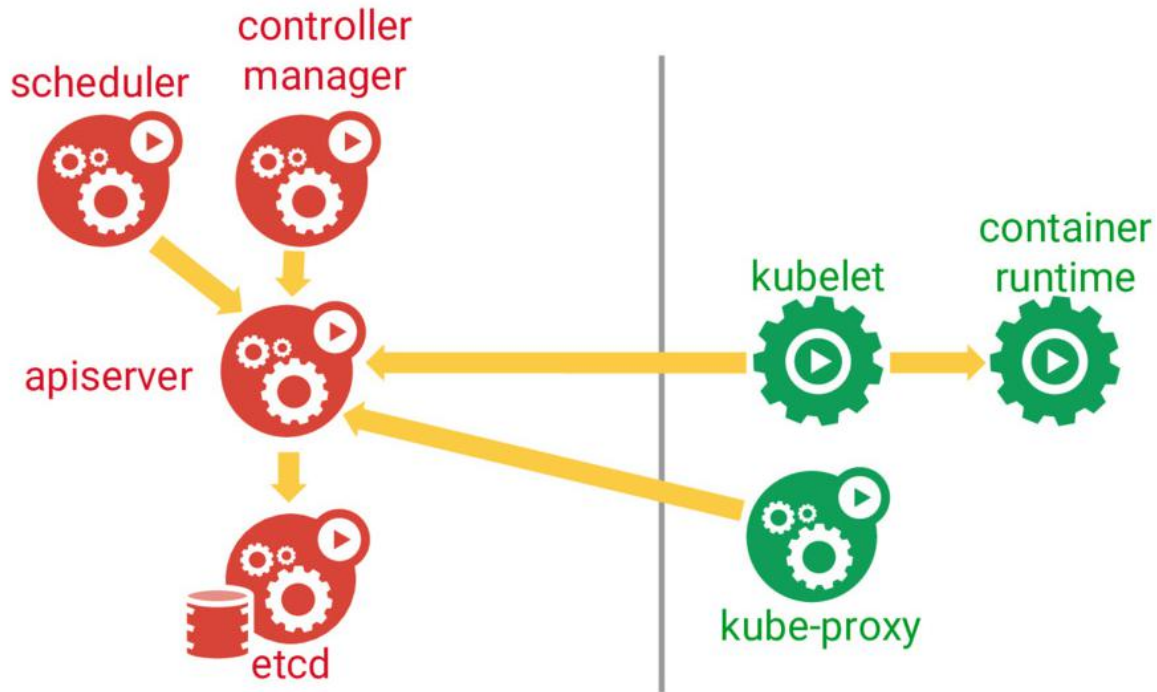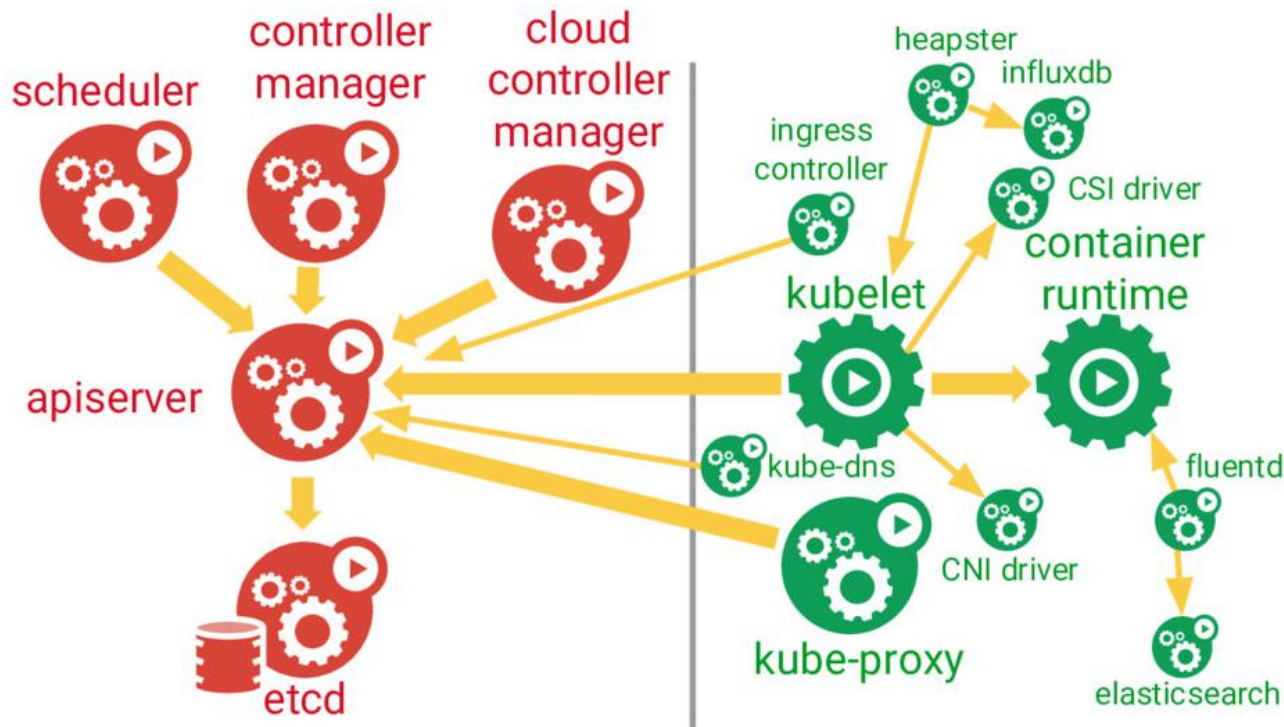The goal is to know where to find necessary information in the fast moving community

From Dan Kohn, Kubecon + CloudnativeCon 2017, Austin

From Tim Hockin, Kubecon + CloudnativeCon 2017, Austin

From Tim Hockin, Kubecon + CloudnativeCon 2017, Austin

Definition

Special interest groups are for discussing subtopics within the broader Kubernetes project among interested contributors.

Apps	Scalability	Autoscaling	Storage	UI
Service Catalog	Federation	Instrumentation	Contributor Experience	Scheduling
Node	AWS	Cluster life cycle	API Machinery	Windows
Auth	OpenStack	Cluster Ops	PM	Docs
Network	Bare Metal			Testing
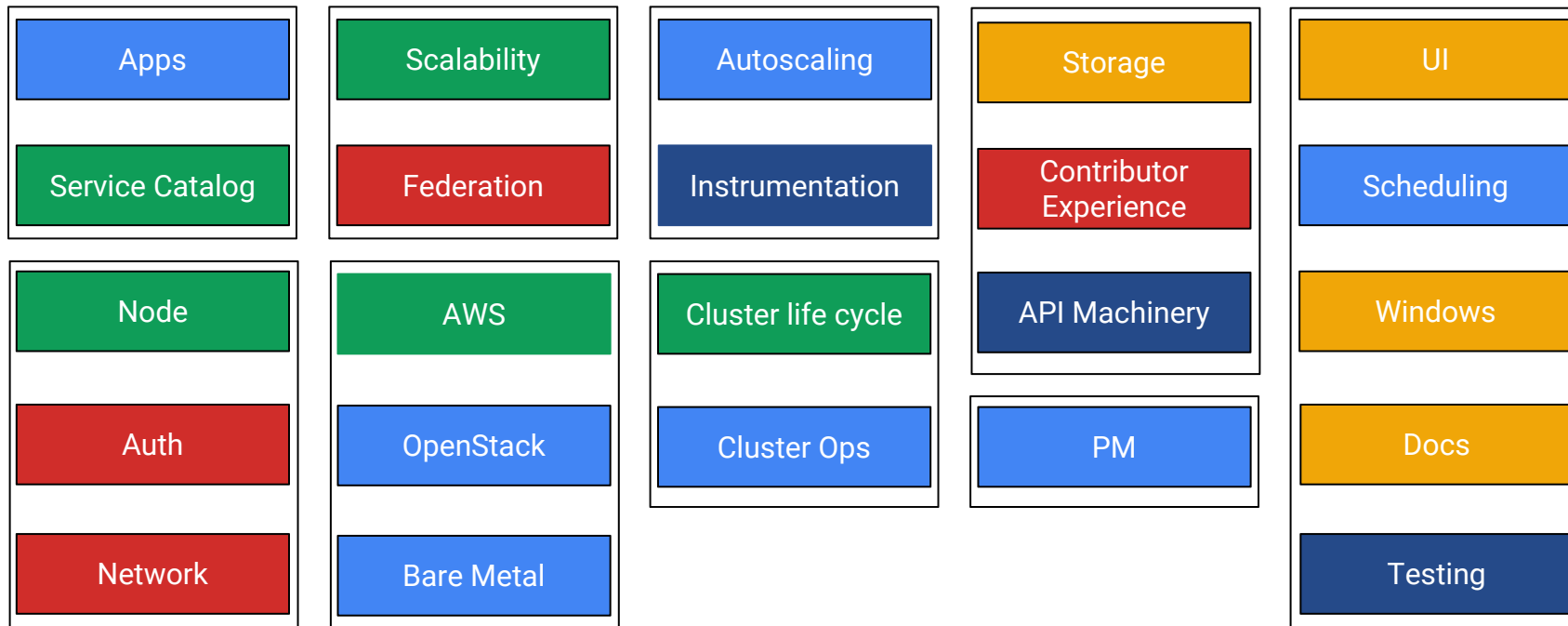
Problem

- Need collaboration between different SIGs

- How to manage ever-growing SIGs
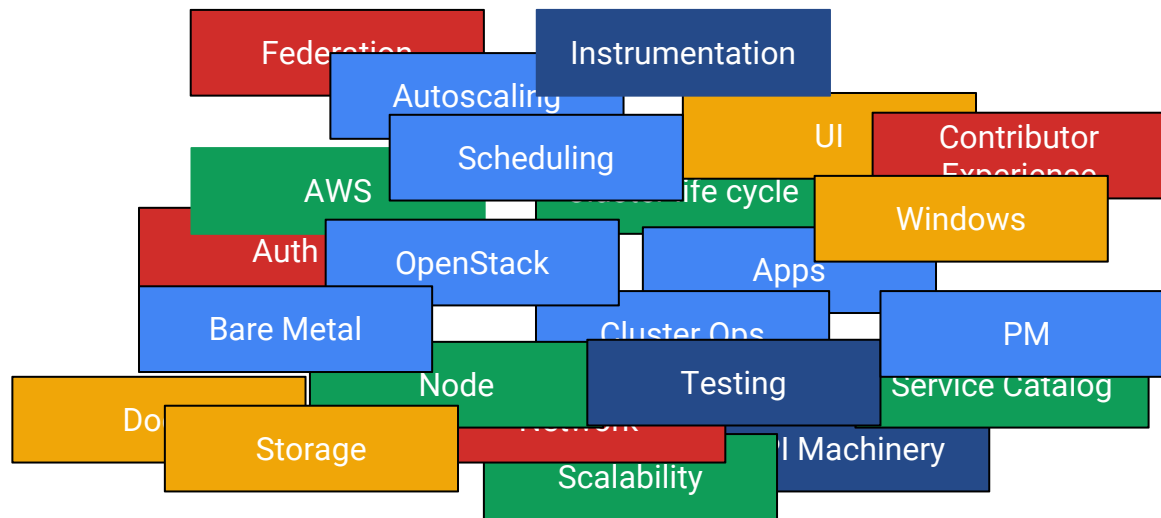
Solution

- Stricter SIG creation process

- Use timed Working Group to manage cross SIG collaboration

- Creation of SIG-PM, SIG-architecture, etc to govern project lifecycle

Goal

- SIG PM facilitates aspects of the Kubernetes project lifecycle

Activities

- Collecting and generalizing user feedback to help drive project direction and priorities
- Supporting collaboration across the community by working to improve the communication of roadmap
- Supporting the continued effort to improve the stability and extensibility of Kubernetes Project
- Supporting the marketing and promotion of the Kubernetes Project through the CNCF
- Working with the Kubernetes Release Team to continually ensure a high quality release
- Supporting the Kubernetes ecosystem through the Kubernetes Incubator

Apps

- Workloads API (planned v1 in 1.9)

Scalability

- 5k nodes deployment

Networking

- Network Policy v1

Node

- CRI
- GPU Support

Auth

- RBAC

Cloud Providers

- Out-of-core movement

Cluster Lifecycle

- kubeadm enhancements

Autoscaling & Instrumentation

- Custom metrics for HPA

Storage

- CSI

Problems on current feature-based proposal process

- Information is spread across multiple places

- No living documents governing a 'feature'

- Hard to collaborate across different SIGs

- Duplicate work around the community

- Release burden with feature planning

Proposals -> KEP (Kubernetes Enhancement Proposal)

- Steel thread: the minimum size required to implement the main elements of a design
- Focus on enhancement and adding value, not just feature-by-feature process
- Not just Kubernetes itself, but across the whole Kubernetes ecosystem

A bit more information

- Formal design proposal for KEP
- An example KEP for SIG cluster lifecycle planning, facilitated from SIG-PM

```
├── 0000-kep-template.md
├── CODEOWNERS
├── index.md
├── sig-architecture
│   ├── deferred
│   ├── orphaned
│   └── retired
├── sig-network
│   ├── deferred
│   ├── kube-dns
│   ├── orphaned
│   └── retired
├── sig-node
│   ├── deferred
│   ├── kubelet
│   ├── orphaned
│   └── retired
├── sig-release
│   ├── deferred
│   ├── orphaned
│   └── retired
├── sig-storage
│   ├── deferred
│   ├── orphaned
│   └── retired
├── unsorted-to-be-used-by-newcomers-only
└── wg-resource-management
    ├── deferred
    ├── orphaned
    └── retired
```

Reference

- SIG-Product-Management README

- Kubecon SIG-PM slide

- Kubecon contributor summit nodes

- Kubernetes Enhancement Proposal Template, and an Example

Goal

- Maintains and evolves the design principles of Kubernetes

Activities

- Establishing and documenting design principles
- Maintaining, evolving, and enforcing the deprecation policy
- Documenting and evolving the system architecture
- Driving improvement of overall code organization, including github orgs and repositories
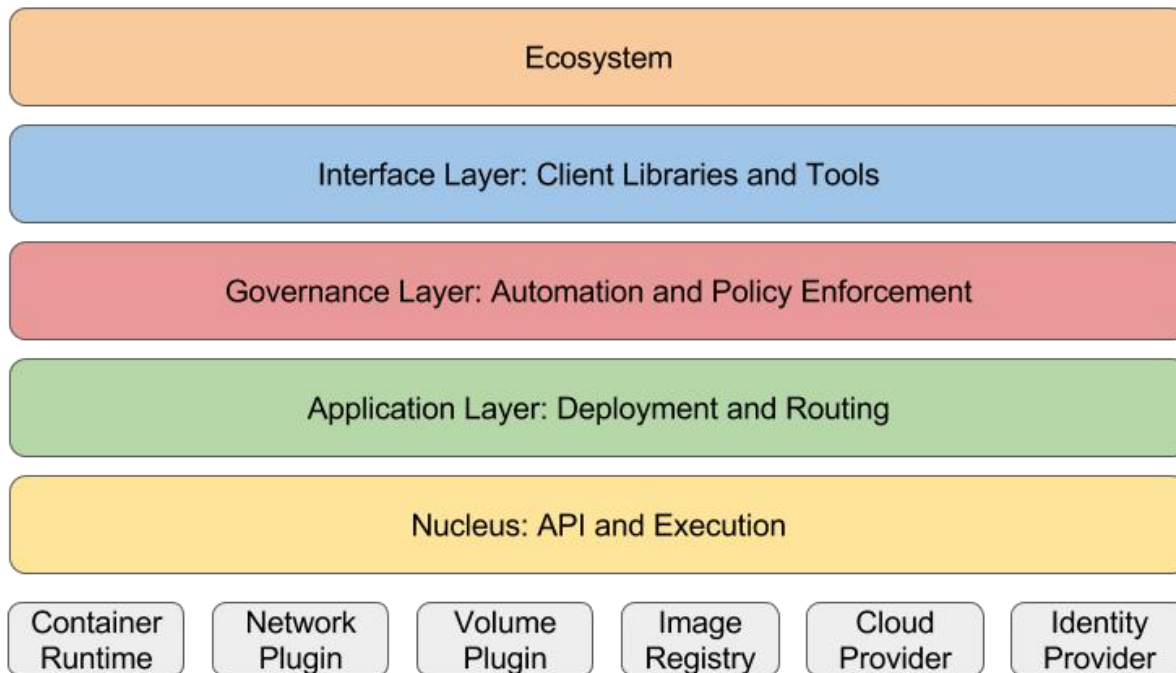
SIG-architecture

Governance Layer: Automation and Policy Enforcement

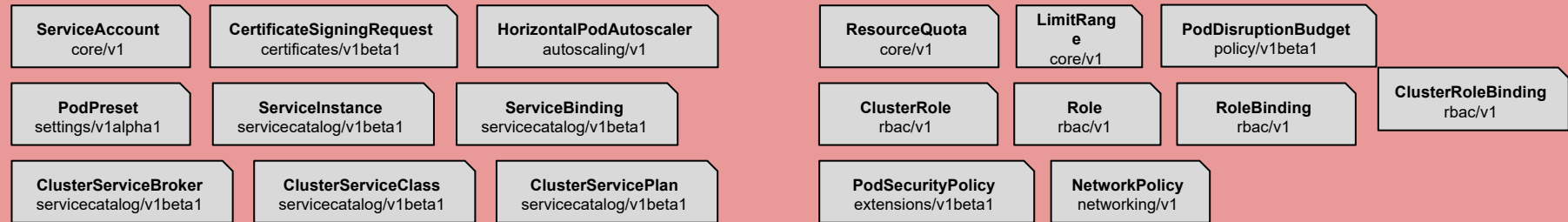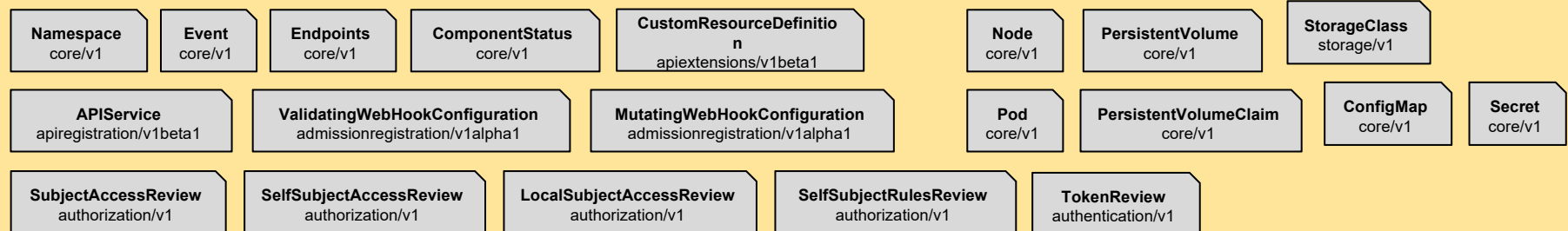|					
ServiceAccount core/v1	**CertificateSigningRequest** certificates/v1beta1	**HorizontalPodAutoscaler** autoscaling/v1	**ResourceQuota** core/v1	**LimitRange** core/v1	**PodDisruptionBudget** policy/v1beta1
PodPreset settings/v1alpha1	**ServiceInstance** servicecatalog/v1beta1	**ServiceBinding** servicecatalog/v1beta1	**ClusterRole** rbac/v1	**Role** rbac/v1	**RoleBinding** rbac/v1 **ClusterRoleBinding** rbac/v1
ClusterServiceBroker servicecatalog/v1beta1	**ClusterServiceClass** servicecatalog/v1beta1	**ClusterServicePlan** servicecatalog/v1beta1	**PodSecurityPolicy** extensions/v1beta1	**NetworkPolicy** networking/v1	

Application Layer: Deployment and Routing

Deployment apps/v1	**StatefulSet** apps/v1	**DaemonSet** apps/v1	**ControllerRevision** apps/v1	**ReplicaSet** apps/v1	**ReplicationController** core/v1	**Service** core/v1	**Ingress** extensions/v1beta1
Job batch/v1	**CronJob** batch/v2alpha1	**PriorityClass** scheduling/v1alpha1					

Nucleus: API and Execution

Namespace core/v1	**Event** core/v1	**Endpoints** core/v1	**ComponentStatus** core/v1	**CustomResourceDefinition** apiextensions/v1beta1	**Node** core/v1	**PersistentVolume** core/v1 **StorageClass** storage/v1
APIService apiregistration/v1beta1	**ValidatingWebHookConfiguration** admissionregistration/v1alpha1	**MutatingWebHookConfiguration** admissionregistration/v1alpha1			**Pod** core/v1	**PersistentVolumeClaim** core/v1 **ConfigMap** core/v1 **Secret** core/v1
SubjectAccessReview authorization/v1	**SelfSubjectAccessReview** authorization/v1	**LocalSubjectAccessReview** authorization/v1	**SelfSubjectRulesReview** authorization/v1	**TokenReview** authentication/v1		

Reference

- SIG-architecture README
- Architectural Roadmap
- Kubecon 2017 - What is Kubernetes
- Kubecon 2017 - Kernels and Distros

Goal

- Developing and sustaining a healthy community of contributors

Activities

- Metrics: developer statistics
- Consistency: consistent experience across all repositories
- Contributor Feedback: feedback channels for community members to improve experience
- Mentoring & Guidance: onboarding new contributors

Reference

- SIG contribx README

- Mentoring Program

- Contributor Experience 1.9 Planning

Goal

- Documentation, doc processes, and doc publishing for Kubernetes

Activities

- Discuss documentation and docs issues for kubernetes.io

- Plan docs releases for kubernetes

- Suggest improvements to user onboarding through better documentation on Kubernetes.io

- Identify and implement ways to get documentation feedback and metrics

- Help community contributors get involved in kubernetes documentation

Come and Join us!

- https://github.com/kubernetes/kubernetes-docs-cn

Goal

- Production of high quality Kubernetes releases on a reliable schedule

Activities

- Support SIG PM by providing tooling and processes for the generation of release notes
- Coordinate with SIG PM to communicate feature burndown progress during a release cycle

Where to find information

- Go to kubernetes [community repository](#)
  - ○ Information about all SIGs and their goal
  - ○ Look at *meeting agenda/docs*
  - ○ Join slack channel
  - ○ Subscribe to [google groups](#)
  - ○ Walkthrough [proposals](#) see what have been done and whatnot
- Go to kubernetes [features repository](#)
  - ○ Information about every release
  - ○ Github issues for tracking features

Mission

The Foundation's mission is to create and drive the adoption of a new computing paradigm that is optimized for modern distributed systems environments capable of scaling to tens of thousands of self healing multi-tenant nodes.

Cloud native systems will have the following properties:

- Container packaged
- Dynamically managed
- Microservices oriented

CNCF's vision is to establish

- Standardized interfaces between subsystems

- A standard systems architecture describing the relationship between parts

- At least one standard reference implementation of each sub-system

- Thinking about adding extensible architecture that end users can extend, replace or change behavior in every layer of the stack for their purposes
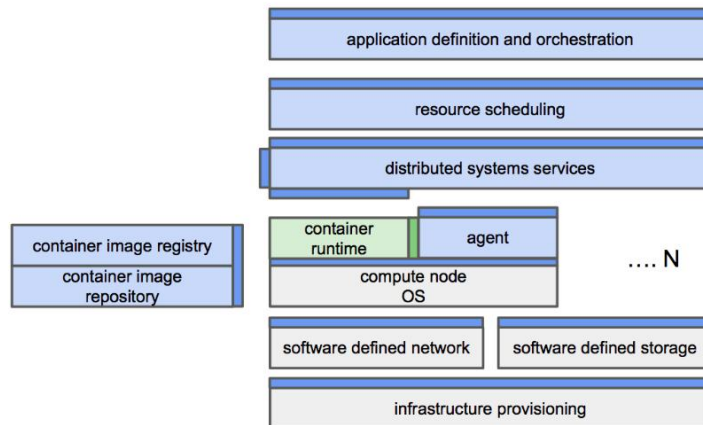
Kubernetes
Orchestration

Prometheus
Monitoring

OpenTracing
Distributed Tracing API

Fluentd
Logging

linkerd
Service Mesh

gRPC
Remote Procedure Call

CoreDNS
Service Discovery

containerd
Container Runtime

rkt
Container Runtime

CNI
Networking API

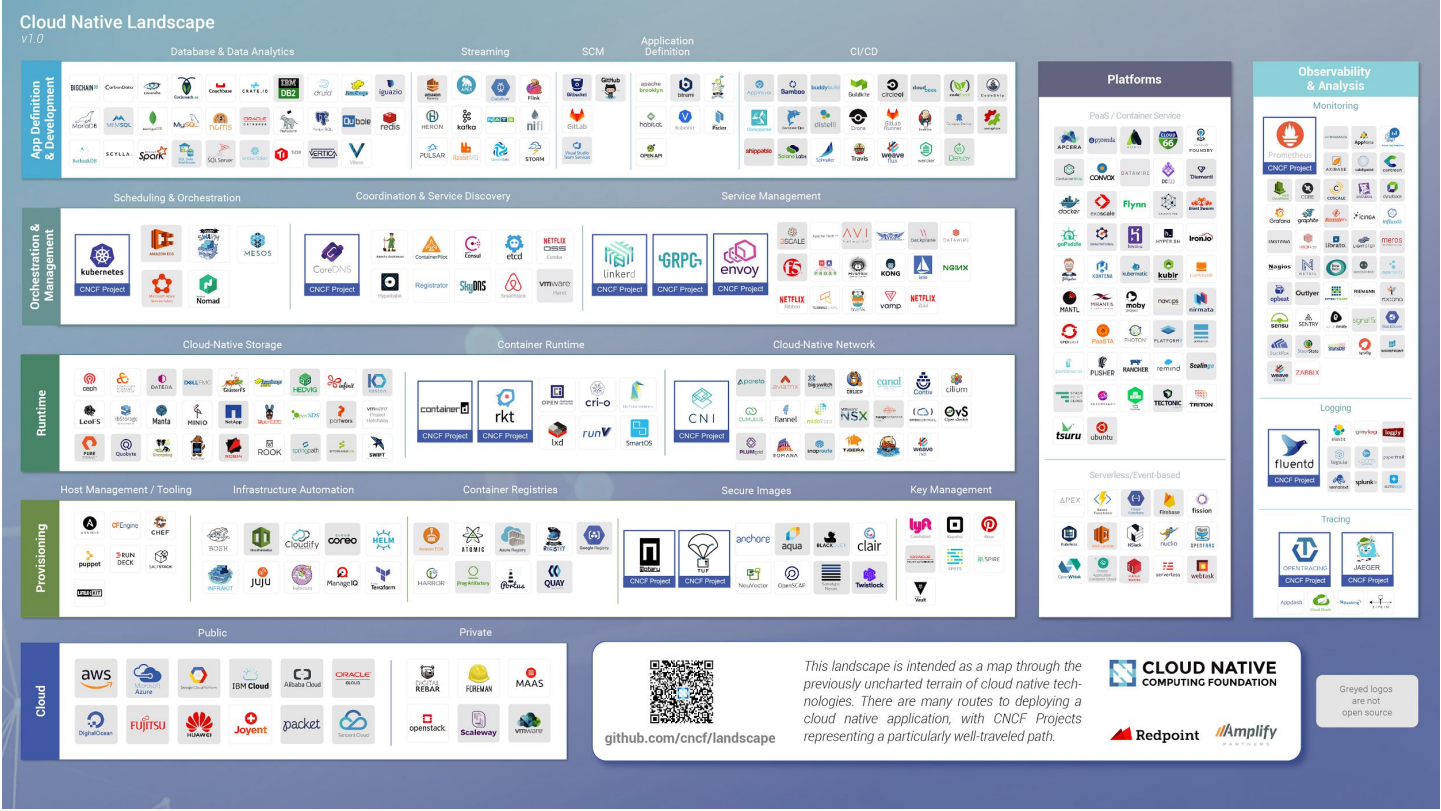# Envoy
Service Mesh

Jaeger
Distributed Tracing

Notary
Security

TUF
Software Update Spec

2 Inception
12 Incubating
0 Graduation

Cloud Native Landscape
v1.0

There are few projects discussed:

- minio - open source object storage server with Amazon S3 compatible API

- nats - a simple, high performance open source messaging system

- CockroachDB - a cloud-native SQL database for building global, scalable cloud services

- Traefik - a modern HTTP reverse proxy and load balancer made to deploy microservices with ease

- Gitlab - integrated product for modern software development

Six existing project proposals to review:

- Istio - an open platform to connect, manage, and secure microservices

- OPA - general-purpose policy engine that enables unified, context-aware policy enforcement

- SPIFFE - standards for identifying and securing communications between web-based services

- Vitess - database clustering system for horizontal scaling of MySQL

- Rook - orchestrator for distributed storage systems running in cloud native environments

- InfraKit - toolkit for infrastructure orchestration

- Governing Board
  - https://www.cncf.io/people/governing-board/
- Ambassadors
  - https://www.cncf.io/people/ambassadors/
- End-user Community
  - https://www.cncf.io/people/end-user-community/
- TOC
  - https://www.cncf.io/people/technical-oversight-committee/
  - https://github.com/cncf/toc/ <- Places to find CNCF projects information

Goal

Provide assurance that CKAs have the skills, knowledge, and competency to perform the responsibilities of Kubernetes administrators.

Domain

- Application Lifecycle Management 8%
- Installation, Configuration & Validation 12%
- Core Concepts 19%
- Networking 11%
- Scheduling 5%
- Security 12%
- Cluster Maintenance 11%
- Logging / Monitoring 5%
- Storage 7%
- Troubleshooting 10%

An example scenario:

You are given a cluster of 1 master and 3 nodes, where one of the node is unhealthy. You have root access to all machines in the cluster. You are tasked to find the root cause of the problem and fix it.

Goal

Provide assurance that CKADs have the skills, knowledge, and competency to perform the responsibilities of Kubernetes application development.

Domain (draft)

- Core Concepts 13%
- Pod Design 20%
- Service & Networking 13%
- Configuration 18%
- Observability 18%
- State Persistence 8%
- Multi-Container Pods 10%

- FOSDEM 2018 Europe, Feb 3-4, 2018

- Open Source Leadership Summit, Mar 6-8, 2018

- Open Networking Summit, Mar 28-29, 2018

- KubeCon + CloudNativeCon Europe 2018, May 2-4, 2018

- Open Source Summit Japan, Jun 20-22, 2018

- KubeCon + CloudNativeCon, Shanghai China, Nov 14-15, 2018

- KubeCon + CloudNativeCon North America, Dec 11-13, 2018