Crossplane

Introduction & Deep Dive

https://crossplane.io

Jared Watts, Steven Borrelli, Yury Tsarev, Christopher Haar

What is Crossplane?

- **Framework** for building cloud native **control planes**
  - No need to write any code
- Cloud providers have been building control planes for years
  - K8s is a control plane for containers - but there's more than that
  - Crossplane helps you build your own - with your own **opinions**
- Extensible backend to manage **any infrastructure** in **any environment**
- Configurable frontend to expose **declarative APIs** (abstractions) for developer **self-service**

CNCF Project for the Community

- Crossplane is a neutral place for vendors and individuals to come together in enabling control planes
- Launched in Dec 2018 by creators of CNCF graduated Rook project
  - Accepted into Sandbox in June 2020
  - First major “stable” milestone v1.0 released in Dec 2020
  - Moved to Incubation September 2021
  - v1.8 released May 2022

Project and Community Stats

6,300+
Stars
2x Increase YoY

46,000+
Contributions
2x Increase YoY

750+
Contributors
5x Increase YoY

5,000+
Followers
2x increase YoY

5,200+
Members
5x increase YoY

25M+
Pulls
10x increase YoY

The Basics

Managed Resources

Managed Resources Example: AWS

Certificates

SQS

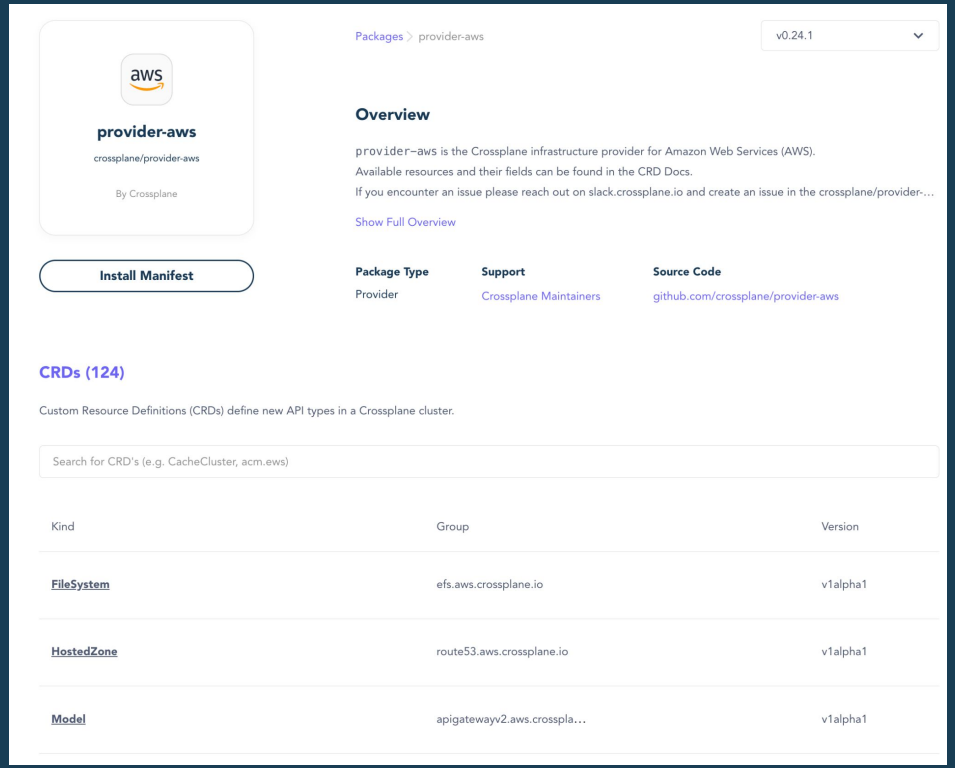
Caches

Kubernetes Clusters

Databases

Networking

....



Crossplane

https://marketplace.upbound.io/providers/crossplane/provider-aws

Managed Resources

```yaml
apiVersion: s3.aws.jet.crossplane.io/v1alpha2
kind: Bucket
metadata:
  name: crossplane-deepdive-demo-bucket
spec:
  forProvider:
    region: eu-west-1
    acl: private
    tags:
      Name: CrossplaneDeepDiveDemoBucket
```

Bucket overview

AWS Region	Amazon Resource Name (ARN)	Creation date
EU (Ireland) eu-west-1	arn:aws:s3:::crossplane-deepdive-demo-bucket	May 16, 2022, 13:33:42 (UTC-05:00)

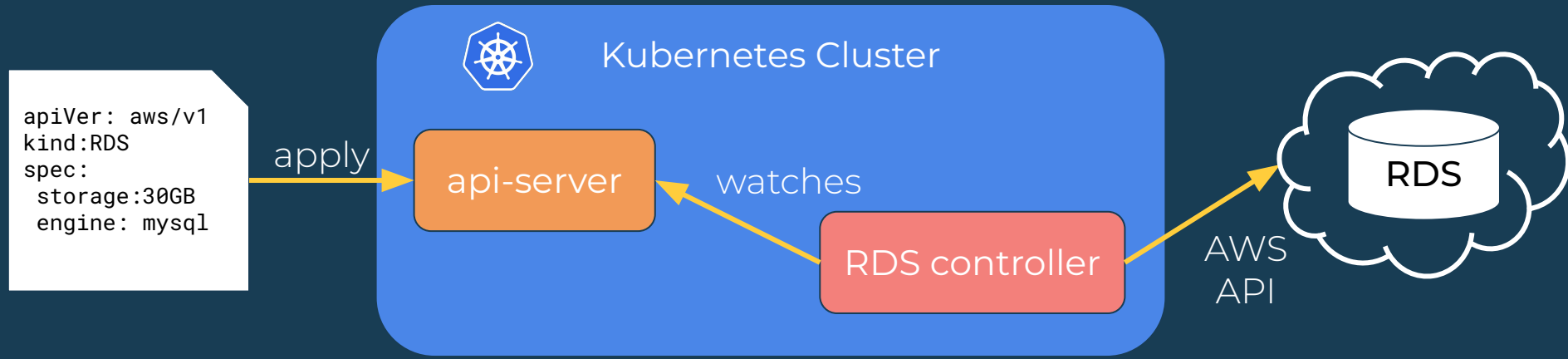
Tags (1)

Track storage cost or other criteria by tagging your bucket. Learn more

Key	Value
Name	CrossplaneDeepDiveDemoBucket

Managed Resource Reconciliation

- Controllers reconcile these CRDs with cloud provider and on-prem APIs (e.g., GCP, AWS, or any API really)



Managed Resources

```
Status:
  At Provider:
    Arn:  arn:aws:s3:::crossplane-deepdive-demo-bucket

Events:
  Type    Reason                  Age   From
Message
  ----    ------                  ----  ----
-------
  Normal  CreatedExternalResource  6m8s  managed/bucket.s3.aws.crossplane.io
Successfully requested creation of external resource
```

Status contains values returned from the remote API and the condition of the resources.

Managed Resources Generate K8s Events

Demo

Managed Resource provisioning

Building Your Control Plane

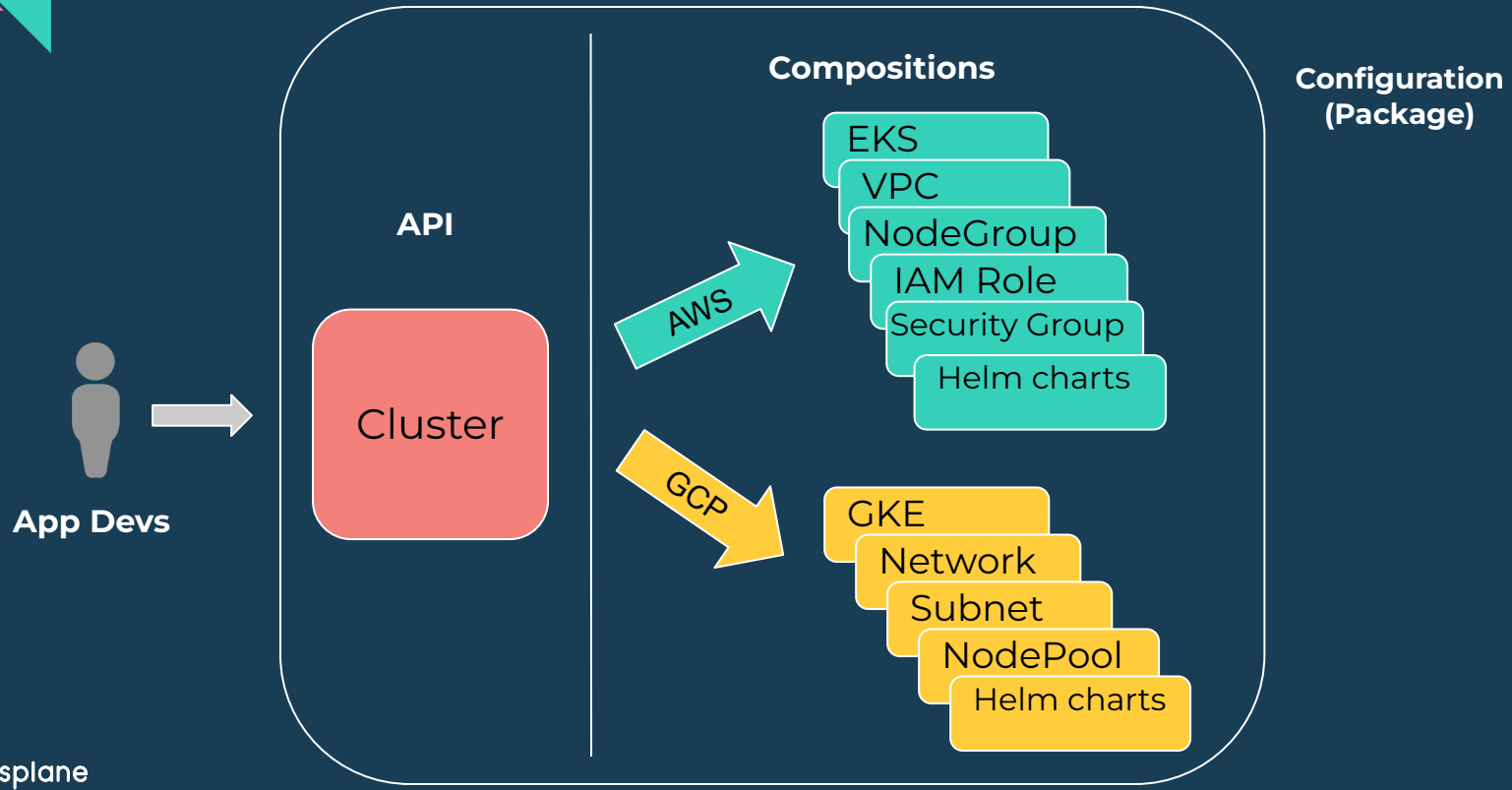
Composition

Crossplane

Build your own Platform API

- Assemble granular resources from multiple vendors/clouds
- Expose as higher level self-service API for your app teams
  - **Compose** GKE, NodePool, Network, Subnetwork, Helm charts
  - **Offer** as a single Cluster resource (API) with limited config for developers to self-service
- Hide infrastructure complexity and include policy guardrails
- All with K8s API - compatible with kubectl, GitOps, etc.
- **No code** required, it's all **declarative**

Platform API Composition Visualized



Crossplane

Composite Resources

First we create Composite Resource Definition(XRD) to declare our custom platform API

```yaml
apiVersion: apiextensions.crossplane.io/v1
kind: CompositeResourceDefinition
metadata:
  name: xpostgresqlinstances.database.example.org
spec:
  group: database.example.org
  names:
    kind: XPostgreSQLInstance
    plural: xpostgresqlinstances
  versions:
  - name: v1alpha1
    served: true
    referenceable: true
    schema:
      openAPIV3Schema:
        type: object
        properties:
```

Custom API Group

Standard openAPIV3 Schema

Crossplane

Compositions

```yaml
apiVersion: apiextensions.crossplane.io/v1
kind: Composition
metadata:
  name: xpostgresqlinstances.aws.database.example.org
spec:
  writeConnectionSecretsToNamespace: crossplane-system
  compositeTypeRef:
    apiVersion: database.example.org/v1alpha1
    kind: XPostgreSQLInstance
  resources:
    - name: parametergroup
      base:
        apiVersion: rds.aws.jet.crossplane.io/v1alpha2
        kind: ParameterGroup
```

Then we define Composition which implements XRD

XRD reference

List of Managed Resources to Compose

Crossplane

Patches

Patches enable propagation of data from Composite Resource (XR) down to composed Managed Resources (MR)

```
patches:
  - fromFieldPath: "spec.nodes.count"
    toFieldPath: "spec.forProvider.scalingConfig.desiredSize"
  - fromFieldPath: "spec.nodes.size"
    toFieldPath: "spec.forProvider.instanceTypes[0]"
    transforms:
      - type: map
        map:
          small: t3.small
          medium: t3.medium
          large: t3.large
```

Copy of value from XR spec down to MR spec

Map transform to manipulate the config data

Demo

Creation of XRD, Composition and Claim

Extending Crossplane

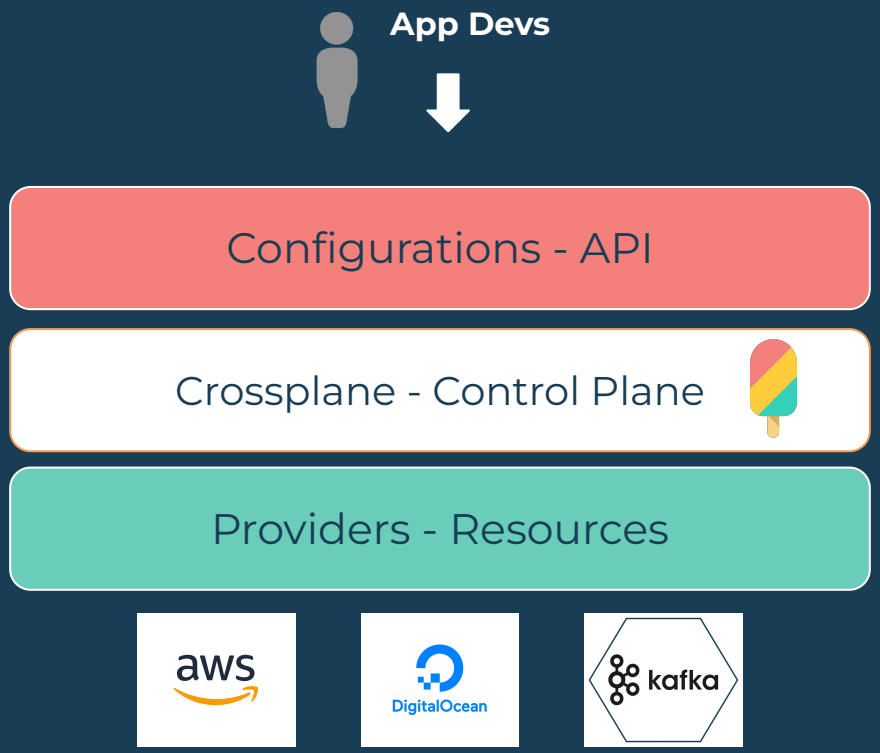
Providers & Configurations

Crossplane

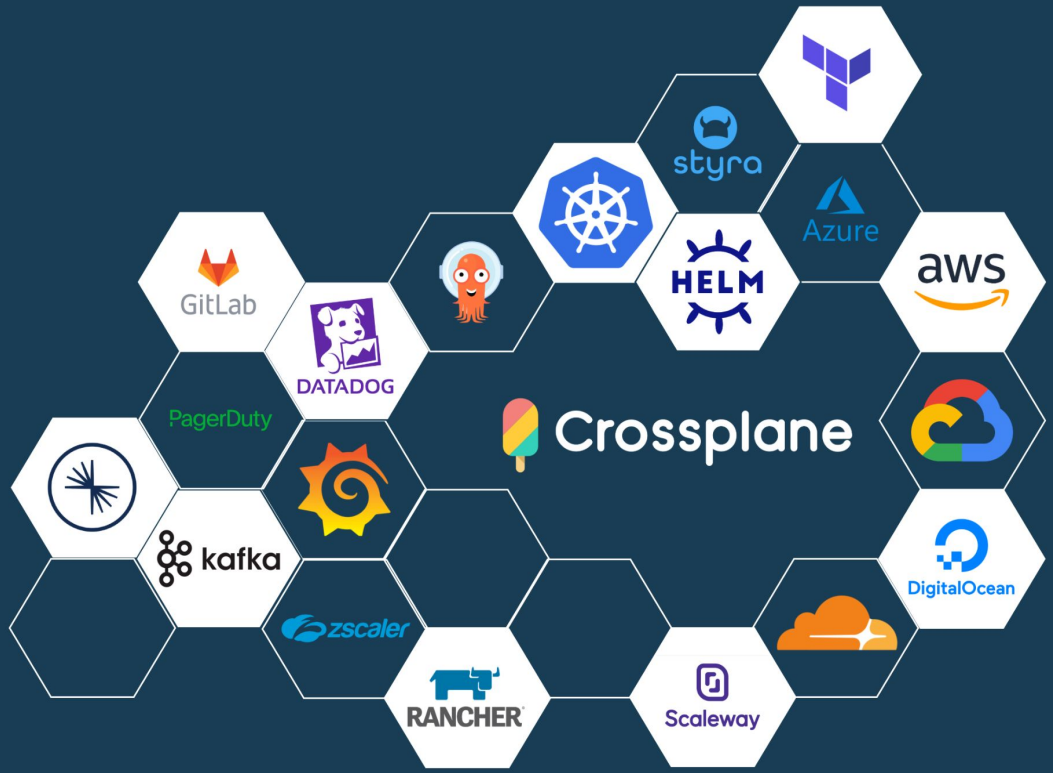
Extension Points

- We've said before that Crossplane is highly extensible - a framework to build a universal control plane
- Backend - **Providers**
  - You can build a provider to manage **anything** with an API
  - CRUD operations for cloud resources, on-prem services, etc.
- Frontend - **Configurations**
  - Compose resources from providers
  - Define your control plane's declarative APIs and abstractions
  - These are what your devs see - it's how they consume the offerings of your control plane

Extensions Visualized



Crossplane Provider Ecosystem



Control Plane Internal Stack

Controller · Controller · Controller · Controller	Custom Logic
Crossplane Runtime	Manage External APIs Create/Update/Delete
Controller Runtime	Event, Watch, Request, Reconciliation
Kubernetes API Machineryr	CRDs, OpenAPI, Persistence (etcd)
Kubernetes Runtime	Run Workloads, Ingress, RBAC

Demo

Adding new resources to a provider

Get Involved

- Website: https://crossplane.io/
- Docs: https://crossplane.io/docs
- GitHub: https://github.com/crossplane/crossplane
- Slack: https://slack.crossplane.io/
- Blog: https://blog.crossplane.io/
- Twitter: https://twitter.com/crossplane_io
- Youtube: Crossplane Youtube

Questions?

Thank you!