

RANCHER[®]

2.0

外からコンテナに繋ぎたい！

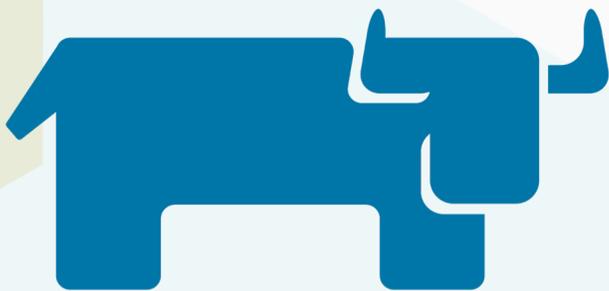
2018/4/27 Rancher Meetup KYOTO #1
@yamamoto_febc

Who?



Kazumichi Yamamoto (@yamamoto_febc)





RANCHER[®] 2.0

外からコンテナに繋ぐには？



kubernetes

クラスタ外から
コンテナに繋ぐには？



kubernetes

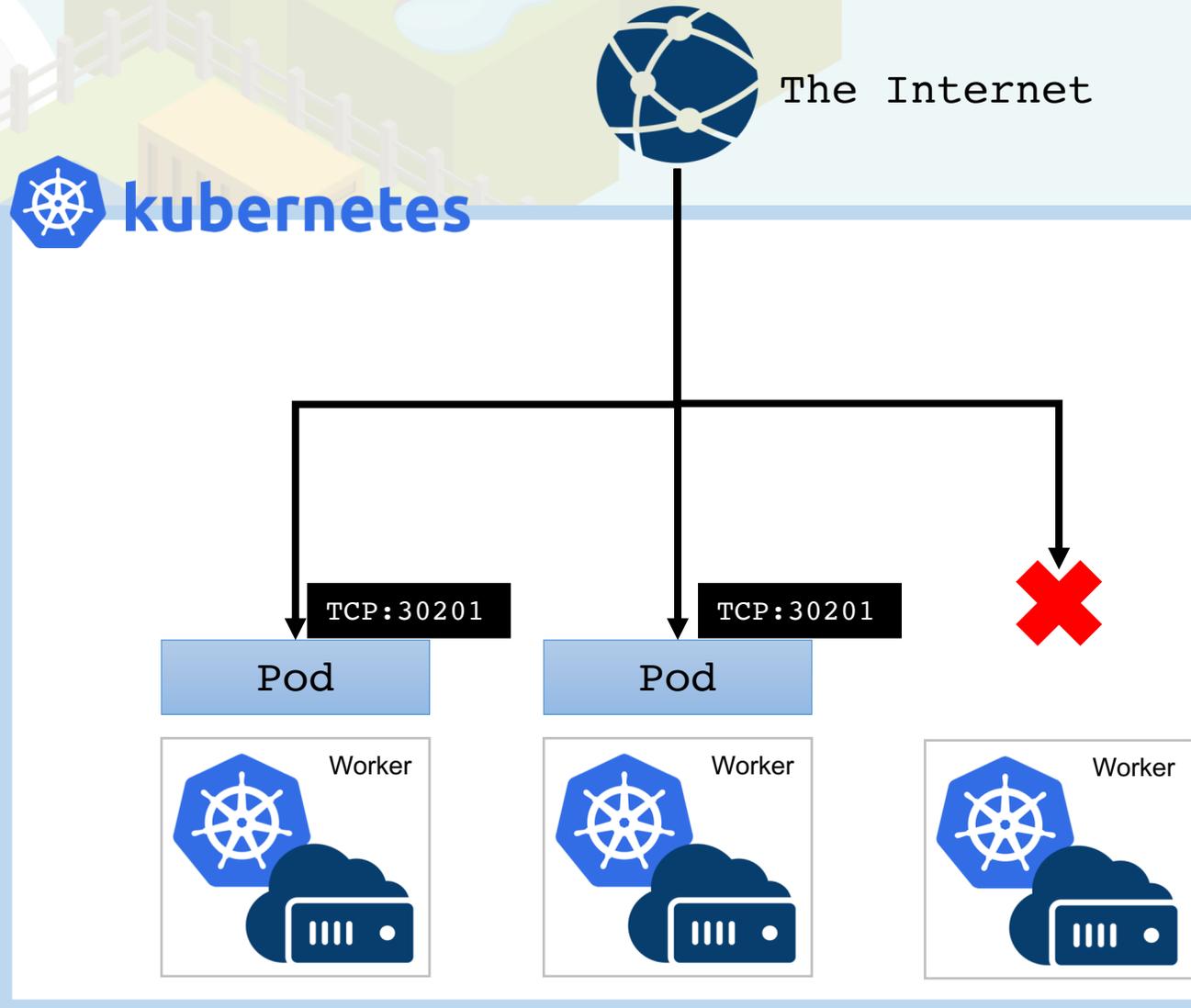
- **HostNetwork/HostPort**
- **Node Port**
- **LoadBalancer**
- **Ingress(GKE/NGINXなど)**

※ External IPsやkube proxy等もありますが省略

HostNetwork/HostPort

- ✓ Podを起動したノードのネットワーク/ポートを直接使う

HostNetwork/HostPort



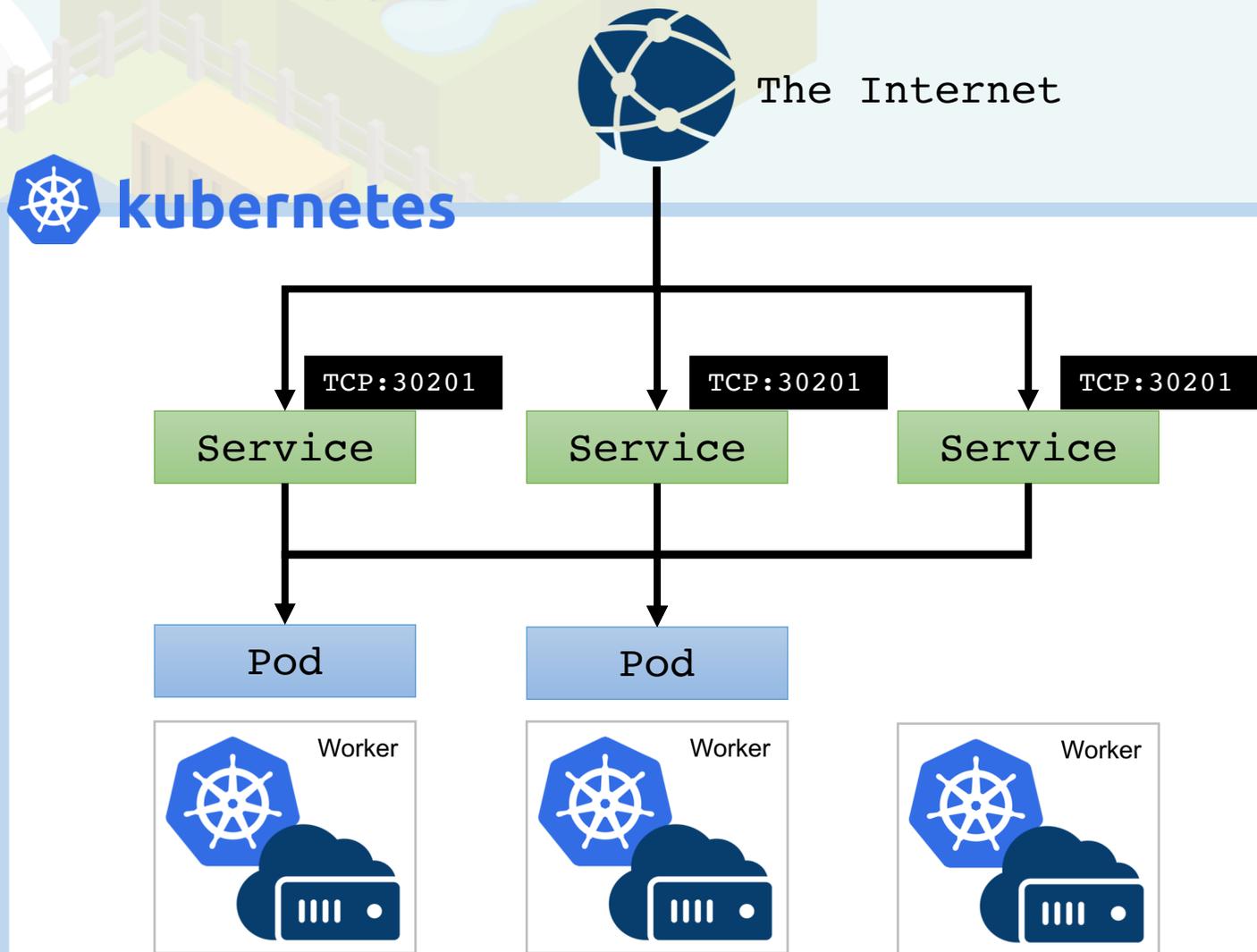
Podが起動しているworkerへの通信のみ可能

どのノードでPodが起動しているか知らないといけない

NodePort

- ✓ 特定ポートへの着信を Podに振り分ける **Service** (kube-proxyによるiptables)

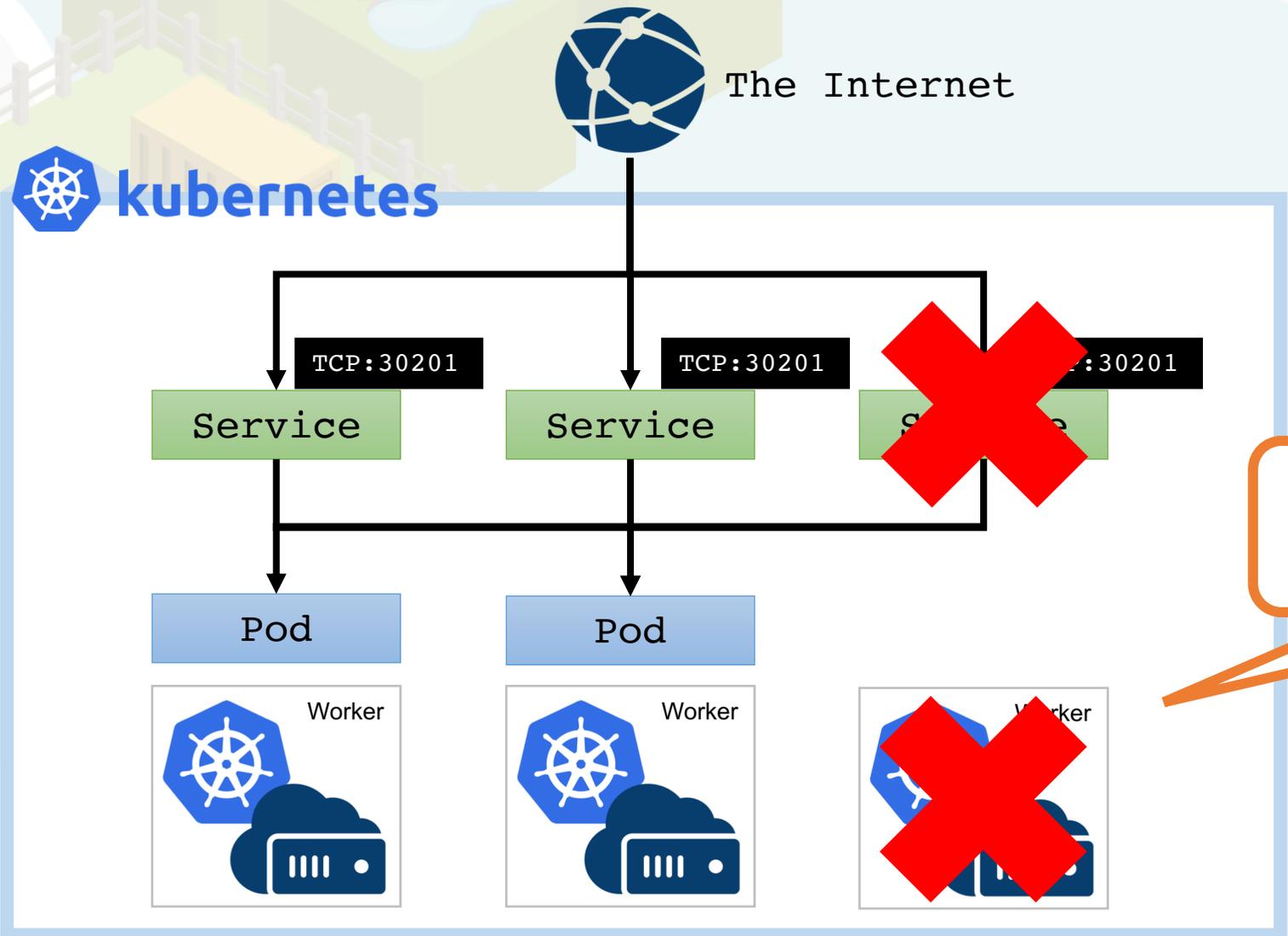
NodePort



各Workerで指定ポートへの着信をproxy(iptables)

各ノードのIPアドレスをDNS登録してホスト名でアクセスすればIP/Podの状況を気にしなくてOK

DNSラウンドロビンだと、、、



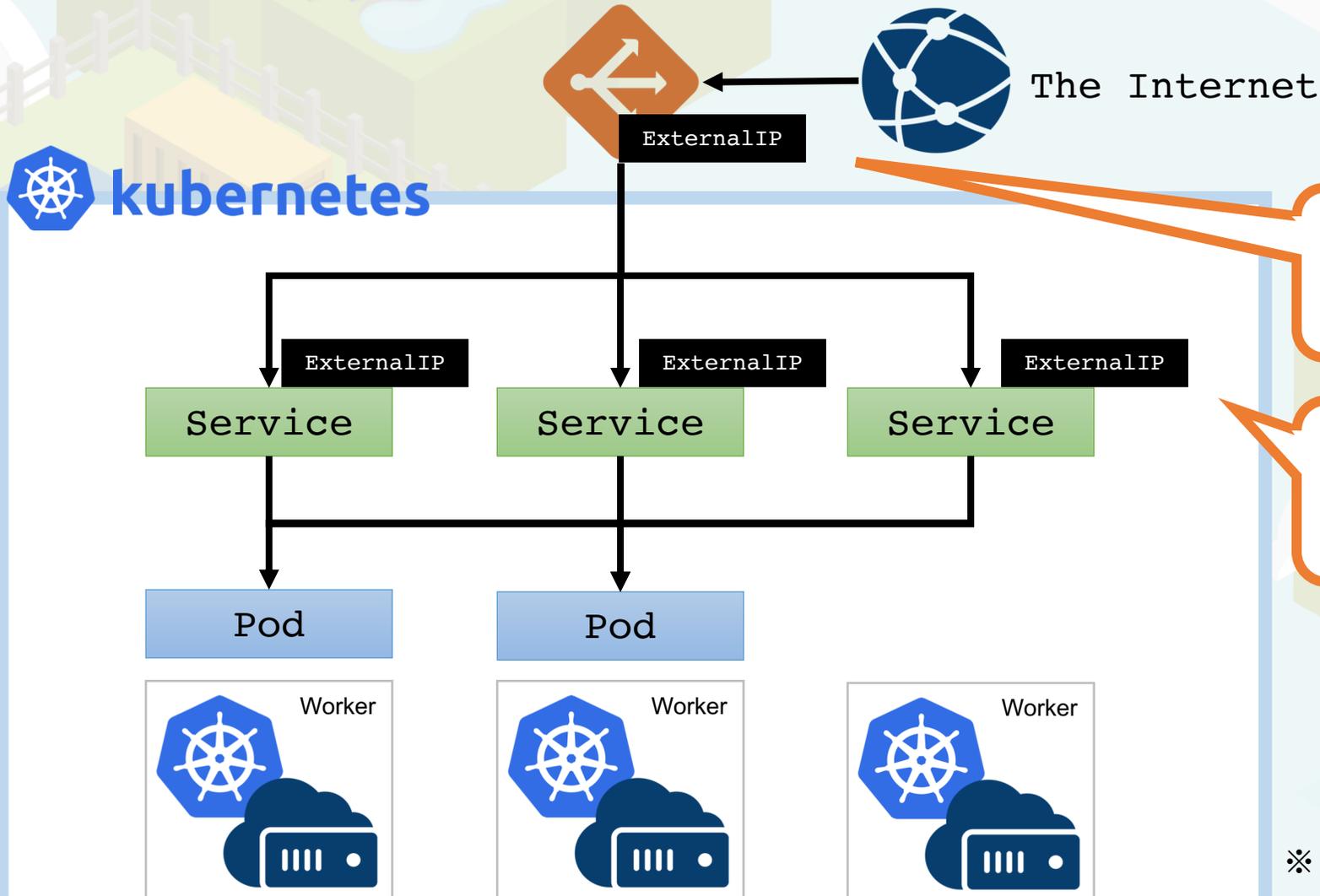
ノードのダウン時、
すぐに切り替えができない

LoadBalancer

An isometric illustration of a farm scene. In the foreground, there's a fenced-in area with a blue building and a yellow building. To the right, a large green field contains a red barn and a yellow tractor. In the background, there are more farm buildings, a windmill, and a pond. The sky is light blue with white clouds.

✓ **L4ロードバランサーで
Serviceへのアクセスを振り分け**

LoadBalancer

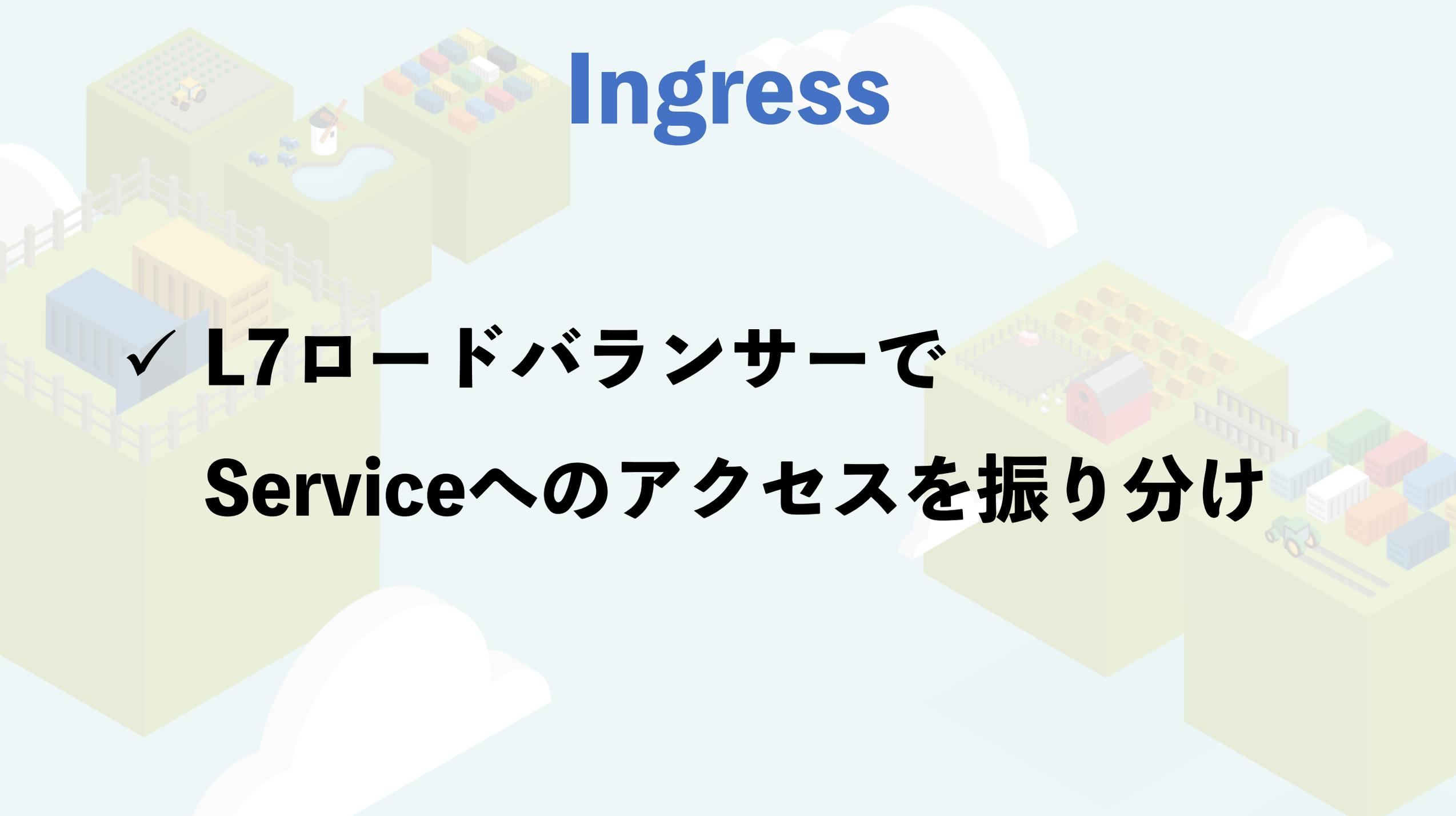


cloud-controller-managerが
ロードバランサを作成

L4ロードバランサ
(IPアドレス/ポート)

- ※ 実装はcloud-controller-managerごとに異なります
- ※ MetalLBのようなccmを使わない方法もあります

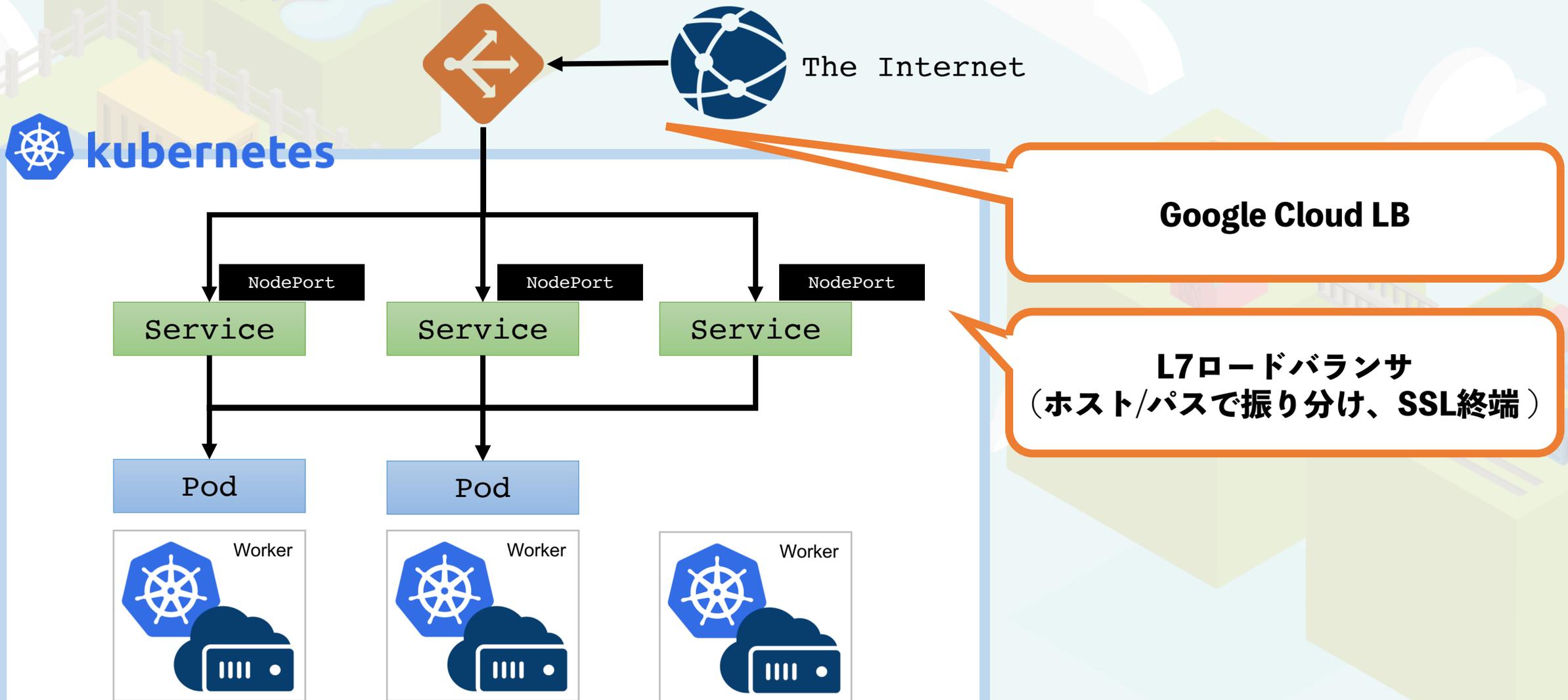
Ingress



✓ L7ロードバランサーで

Serviceへのアクセスを振り分け

Ingress(GKE)



Ingress

GKE以外の場合は？



**自分でIngress Controllerを
用意すればOK**

Ingress(NGINX)

 kubernetes



The Internet



Ingress

Service

Service

Service

Pod

Pod



Worker



Worker



Worker



実体はPod+Service(など)

L7ロードバランサ
(ホスト/パスで振り分け、SSL終端)

Ingress(NGINX)

 kubernetes

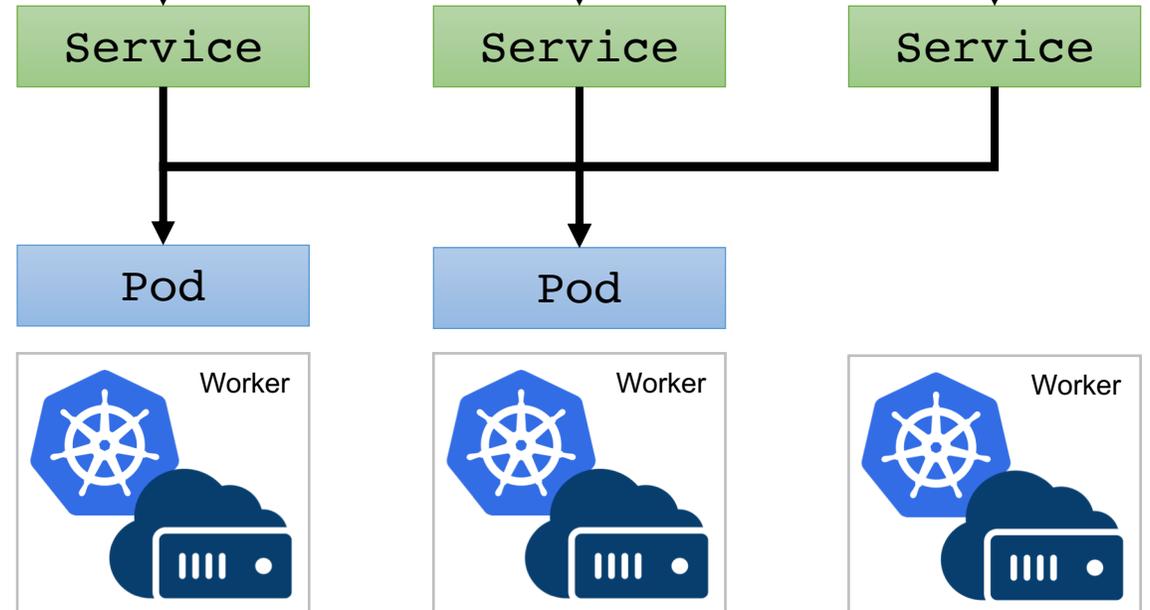
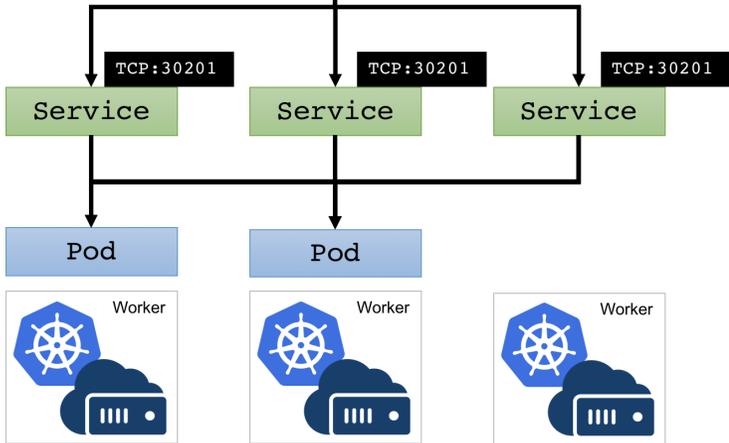


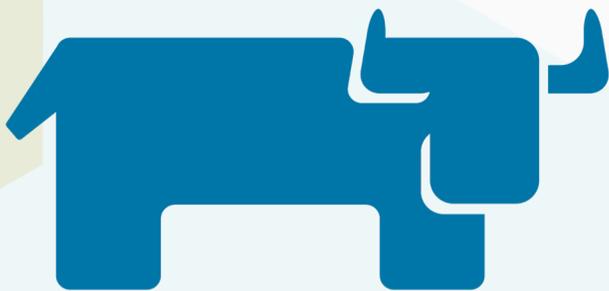
The Internet



Ingress

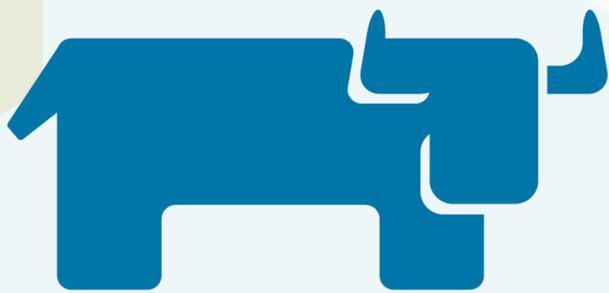
 kubernetes





RANCHER[®] 2.0

Rancherでは？



RANCHER[®]

Web UIから設定できます



Workloadsのデプロイ

local | Default | Workloads | Catalog Apps | Resources | Members

Deploy Workload

Name * [Add a Description](#) Workload Type [More options](#)

nginx Scalable deployment of 1 pod

Docker Image * nginx:latest Namespace * default

Port Mapping

Publish the container port *	Protocol	As a	On listening port *
e.g. 8080	TCP	<ul style="list-style-type: none">✓ NodePort (On every node)HostPort (Nodes running a pod)Cluster IP (Internal only)Layer-4 Load Balancer	Random

[+ Add Port](#)

[Expand All](#)

- Environment Variables**
Set the environment that will be visible to the container, including injecting values from other resources like Secrets.
- Node Scheduling**
Configure what nodes the pods can be deployed to.
- Health Check**
Periodically make a request to the container to see if it is alive and responding correctly.

- ✓ NodePort (On every node)
- HostPort (Nodes running a pod)
- Cluster IP (Internal only)
- Layer-4 Load Balancer



HostPort



HostPort (Nodes running a pod) ▾

Port Mapping

Publish the container port *

80

Protocol

TCP ▾

HostPort (Nodes running a pod) ▾

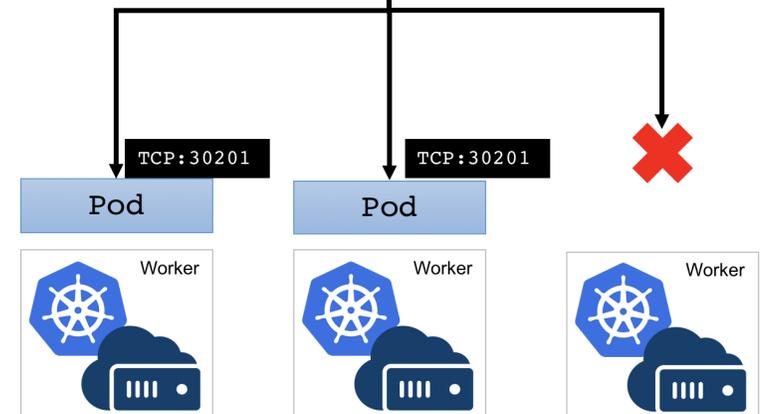
On listening port *

30201

+ Add Port



The Internet





NodePort



NodePort (On every node) ▾

Port Mapping

Publish the container port *

80

Protocol

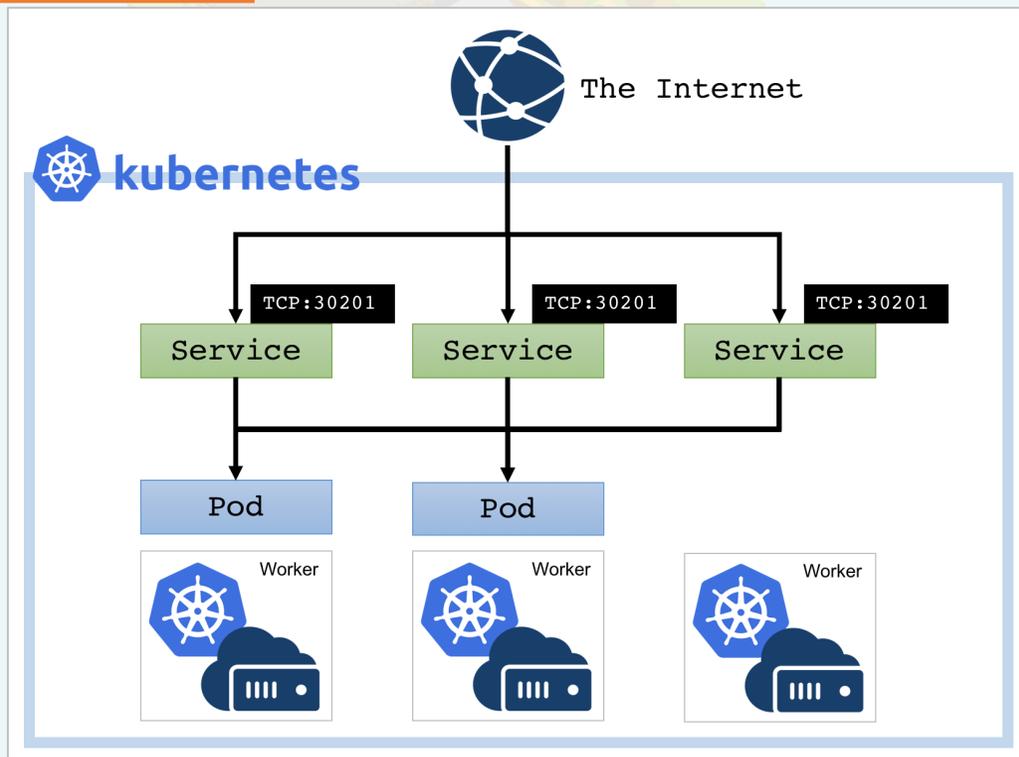
TCP ▾

NodePort (On every node) ▾

On listening port *

Random ✎

+ Add Port





LoadBalancer



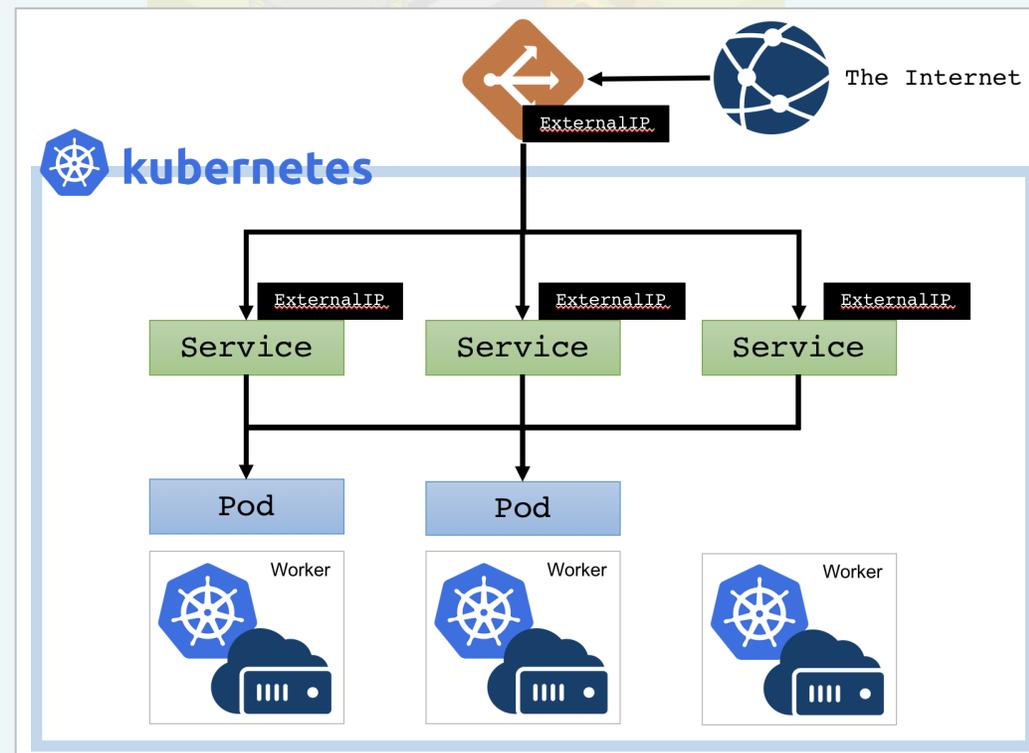
Layer-4 Load Balancer

Port Mapping

Publish the container port *

80	Protocol	As a	On listening port *
	TCP	Layer-4 Load Balancer	80

+ Add Port



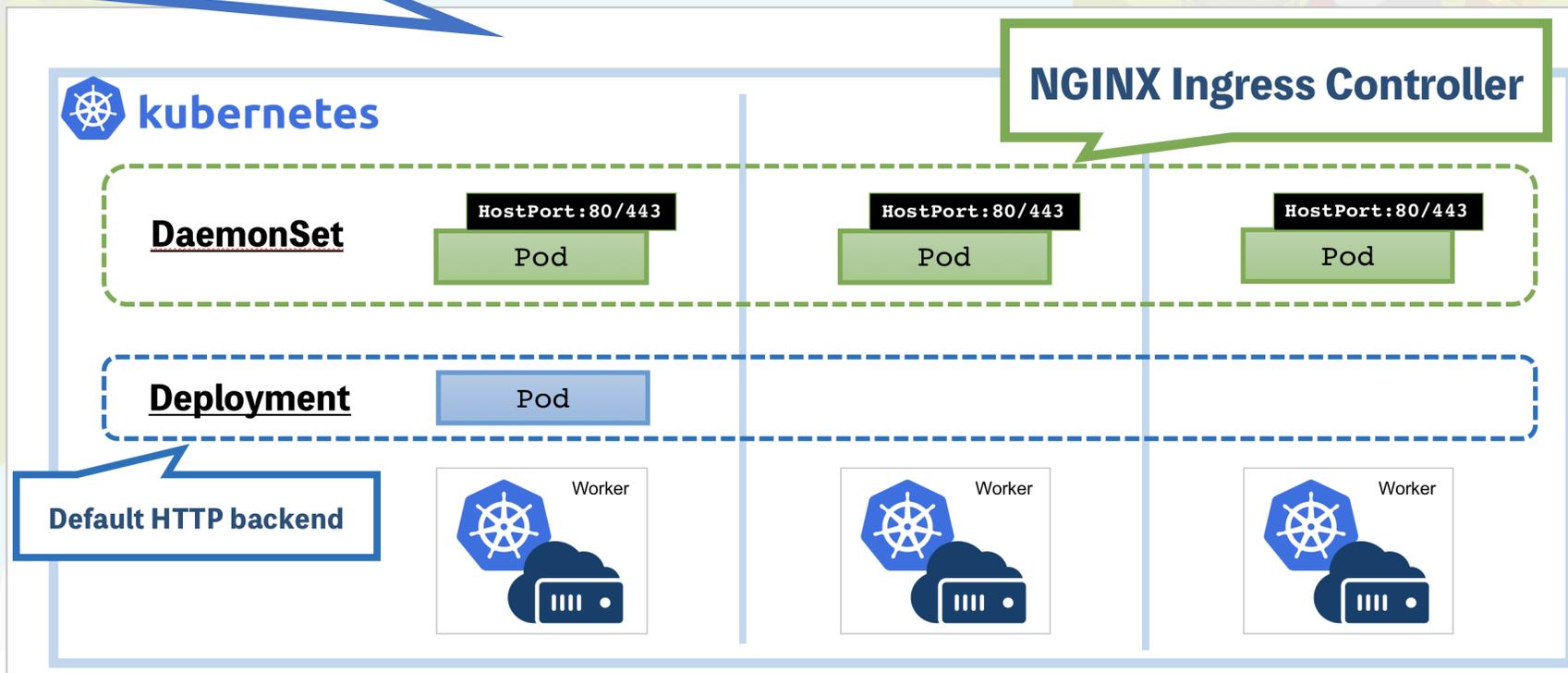
※ cloud-controller-managerが必要

※ RKEではcloud-controller-manager=externalは指定できません(v0.1.5時点)



Ingress Controller

いい感じにセットアップ済み





Ingress

local Default Workloads Catalog Apps Resources Members

Add Ingress

Name: e.g. website Add a Description Namespace: default Add to a new namespace

Rules

Request Host: e.g. example.com

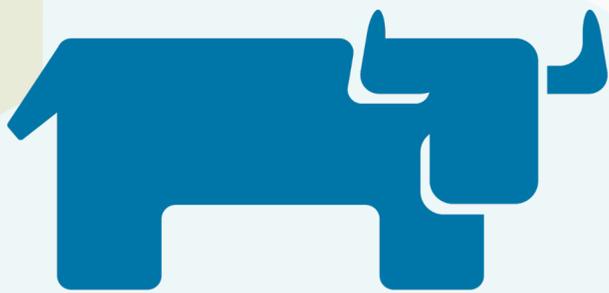
Target Backend: Set this rule as default backend

Path: e.g. /foo Target: Choose a Workload... Port: e.g. 80

宛先ホスト名

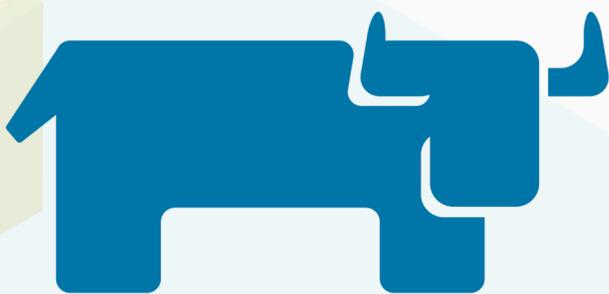
パス

転送先



RANCHER[®]

手軽に使えるよ！



RANCHER[®]

2.0

Enjoy!!

