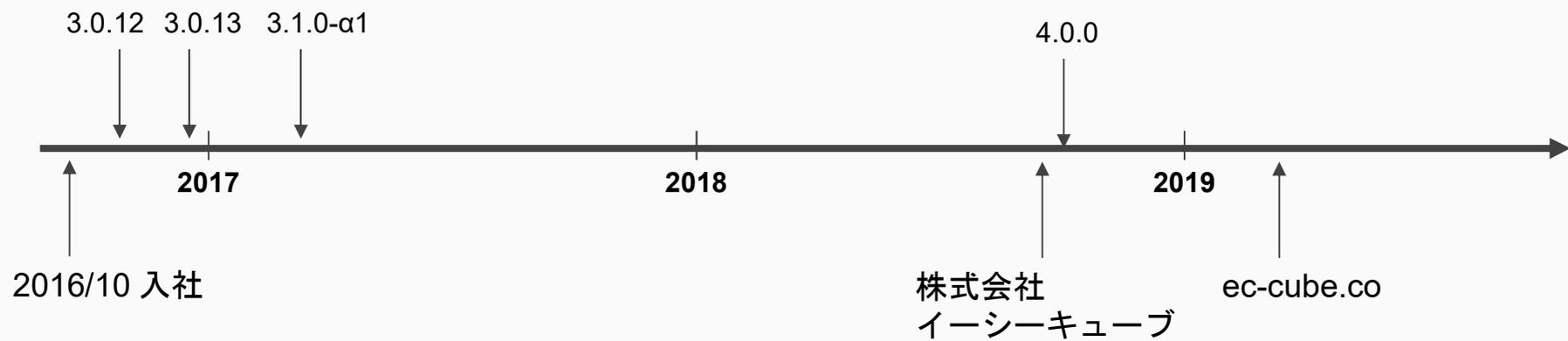


# EC-CUBE4の アーキテクチャとこれから

株式会社イーシーキューブ  
CTO 奥清隆

# 自己紹介

奥清隆 (おく きよたか)



3 から 4 へ

## 3 -> 4

- 当初は少し変更するつもりだった
  - 安定した3系のコードをベース
  - 大きな変更はしないが、マイナーバージョンアップではできない互換性を損なう変更

## The end of Silex (2018-01-12)

“For all these reasons, I would say that Silex is not needed anymore. So, we've decided to not support Symfony 4 in Silex, or at least not add the new features added in 3.4. The current stable version of Silex is still maintained for bugs and security issues. But its end of life is set to **June 2018**.”

<https://symfony.com/blog/the-end-of-silex>

# Silex -> Symfony

EC-CUBE 3.0.x

Silex 1.3.6

Symfony Components

Pimple 1.1

PHP

# Silex -> Symfony

EC-CUBE 4

Symfony 3.4

PHP

# Symfony

- Twig/Doctrineなどのフレームワークは引き続き採用
- アノテーションにより煩雑になりがちな設定を簡潔に
- できる限りSymfonyの機能をそのまま利用し学習コストを抑える

# 擴張機構

# EC-CUBE4 = Symfony + 拡張機構

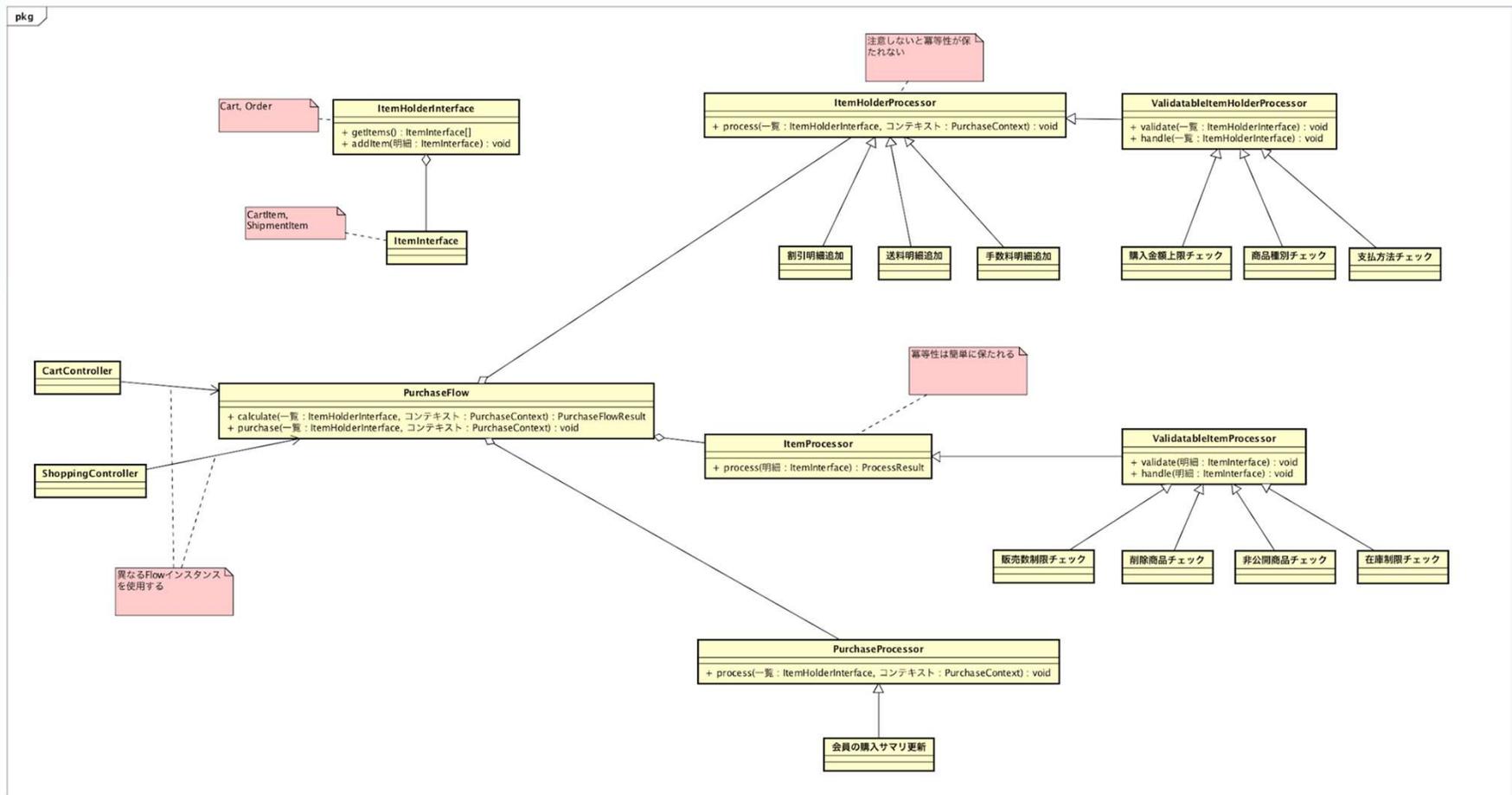
- Entity拡張や金額計算など、EC-CUBEでよくある拡張パターンやこれまで拡張しづらかった箇所を拡張機構として提供

# Entity拡張

- プラグインから既存Entityにプロパティを追加
- データベースにもカラムが追加される
- フォームにも出力可能

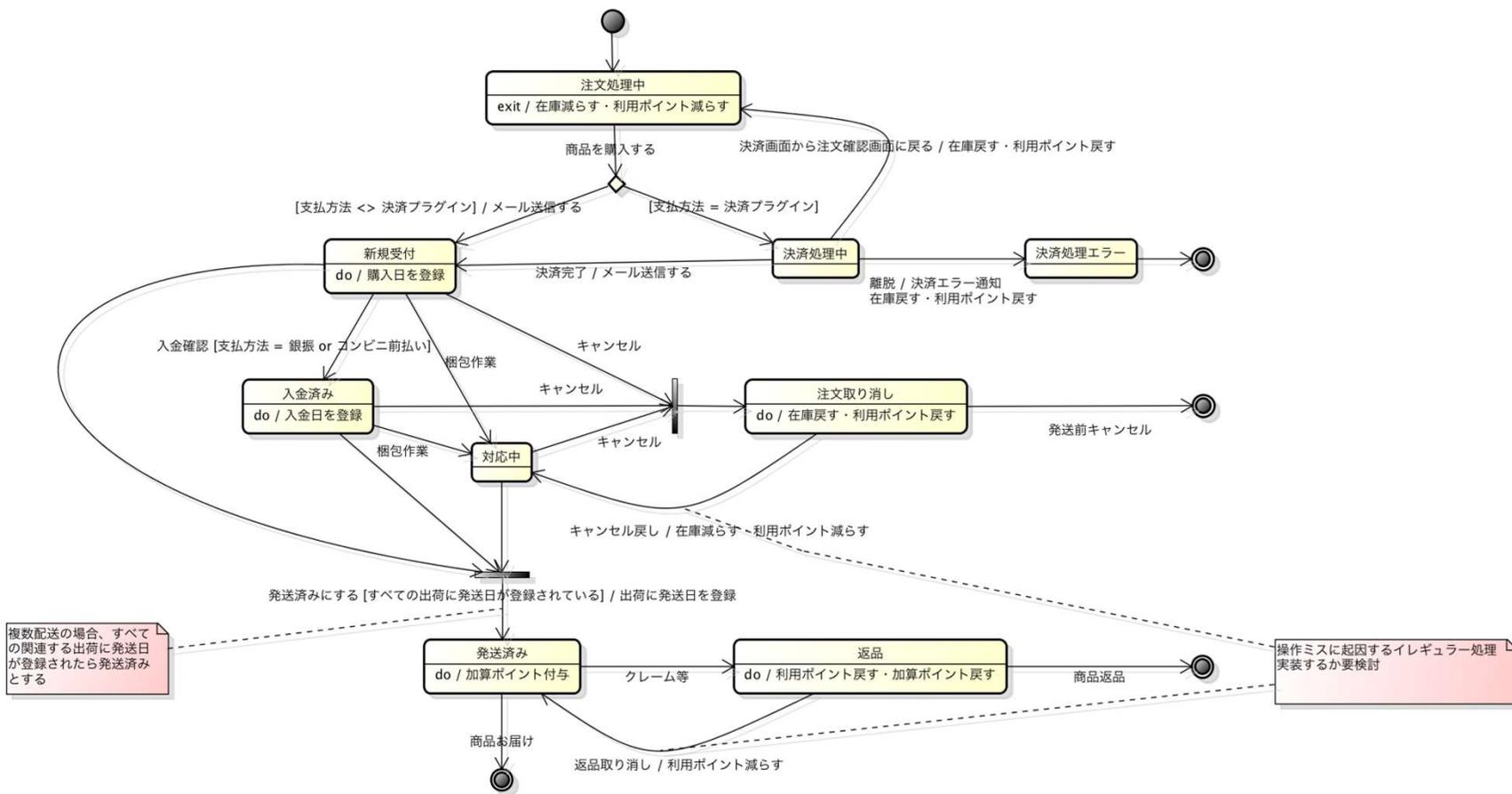
# PurchaseFlow

- フロント購入フローや管理画面受注管理でのカートや受注に関する処理をモデル化し、拡張しやすいように実装



# 受注ステートマシン

- 受注のステータスと状態遷移を定義
- 状態遷移時に発生する処理をカスタマイズ可能に
- Symfony Workflow Componentをベースに実装
  - <https://symfony.com/doc/3.4/components/workflow.html>



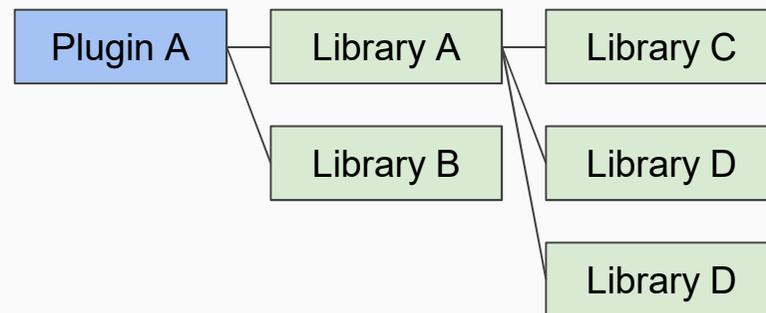
カスタマイズ

# カスタマイズ

- プラグイン
- カスタマイズディレクトリ

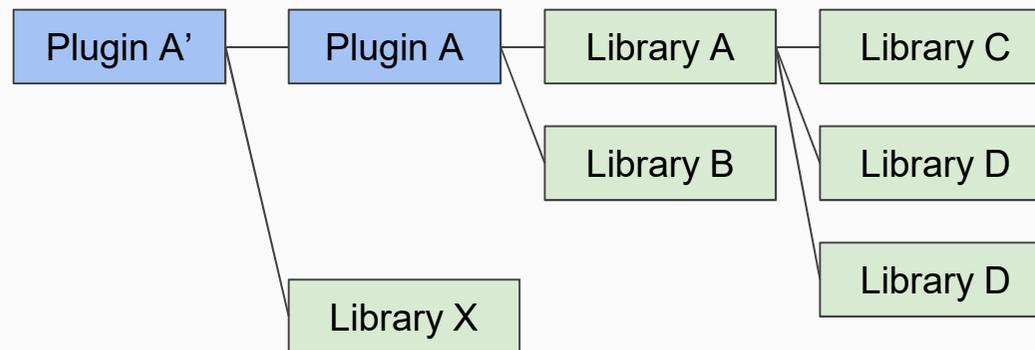
# プラグイン

- Composerによる依存管理



# プラグイン

- Composerによる依存管理



# カスタマイズディレクトリ

- 本体コードを直接変更せずにカスタマイズ
- プラグイン化も可能

## src/Eccube

Controller  
Entity  
Form  
Repository  
Resource  
Service  
...

本体コード

## app/Customize

Controller  
Entity  
Form  
Repository  
Resource  
Service

カスタマイズディレクトリ

## app/Plugin/MyPlugin

Controller  
Entity  
Form  
Repository  
Resource  
Service

プラグイン

実行環境

# 実行環境

- EC-CUBE4対応レンタルサーバー
  - [https://www.ec-cube.net/partner/search\\_rental03.php](https://www.ec-cube.net/partner/search_rental03.php)

# ベンチマーク

Time Per Request※1 (1リクエストあたりの平均処理時間)

	アイテム数※2	EC-CUBE3.0.16		EC-CUBE4.0
トップ	4点	0.07 s	▶	0.03 s
	90,000点	0.11 s	▶	0.05 s
商品一覧	4点	0.08 s	▶	0.04 s
	90,000点	0.33 s	▶	0.16 s

さくらのクラウド (1vCPU/1GB/SSD)での測定結果

# ベンチマークコード

- <https://github.com/EC-CUBE/eccube-bench>
- さくらのクラウド上にサーバーを起動しEC-CUBEをインストールしApache Benchを実行

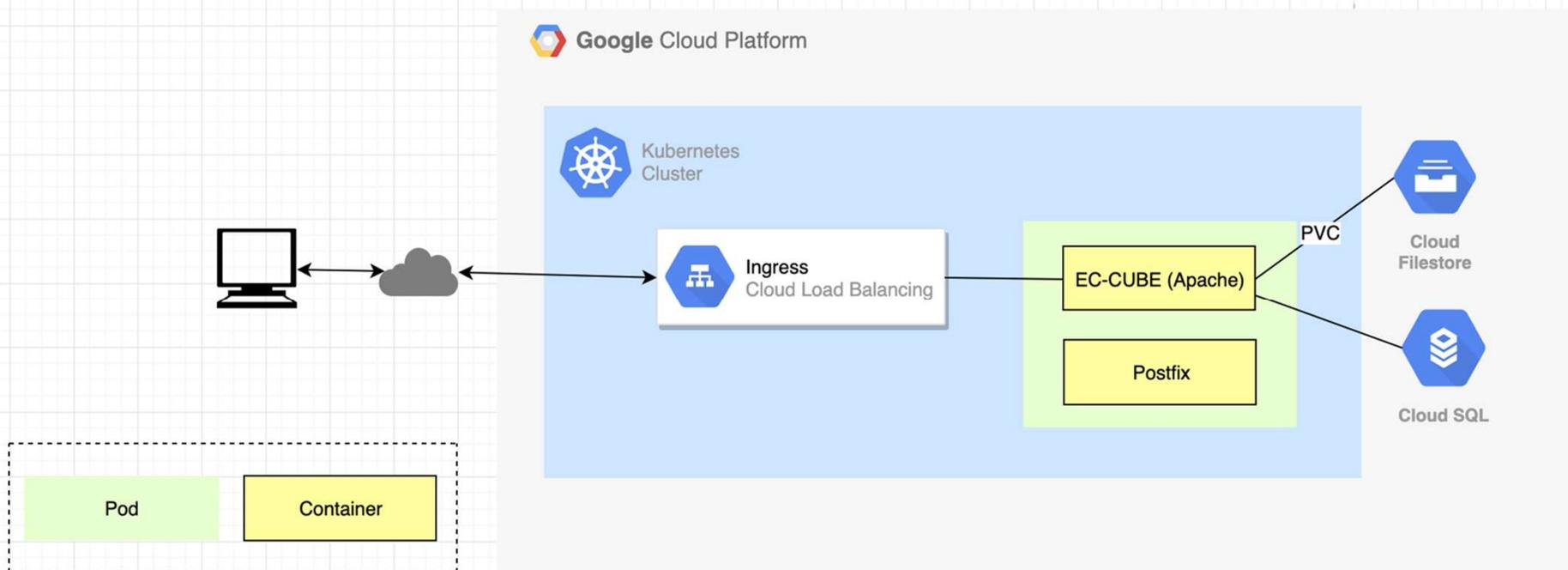
# Kubernetesでの運用 - ec-cube.co

- ハイパフォーマンス
- スケーラブル
- セルフヒーリング
- ローリングアップデート

# ec-cube.co

- 共創型クラウドECプラットフォーム
- サーバー管理不要
- OSS版EC-CUBEと同じコード
- 自動アップデート

# Kubernetesでの運用



# パフォーマンス

- 共有Volumeに置くファイル数を減らす
- リソースリミットを設定する
- 可能ならキャッシュディレクトリをメモリに乗せる

## 共有するファイル数を減らす

```
/var/www/ec-cube/  
|-- app  
| |-- Customize  
| |-- DoctrineMigrations  
| |-- Plugin  
| |-- PluginData  
| |-- config  
| |-- proxy  
| `-- template  
|-- bin  
|-- composer.json  
|-- composer.lock  
|-- html  
|-- index.php  
|-- maintenance.php  
|-- src  
|-- symfony.lock  
|-- var  
`-- vendor
```

## 共有するファイル数を減らす

```
/var/www/ec-cube/  
|-- app  
| |-- Customize -> /mnt/ec-cube/app/Customize  
| |-- DoctrineMigrations  
| |-- Plugin -> /mnt/ec-cube/app/Plugin  
| |-- PluginData -> /mnt/ec-cube/app/PluginData  
| |-- config  
| |-- proxy -> /mnt/ec-cube/app/proxy  
| `-- template -> /mnt/ec-cube/app/template  
|-- bin  
|-- composer.json -> /mnt/ec-cube/composer.json  
|-- composer.lock -> /mnt/ec-cube/composer.lock  
|-- html -> /mnt/ec-cube/html  
|-- index.php  
|-- maintenance.php  
|-- src  
|-- symfony.lock -> /mnt/ec-cube/symfony.lock  
|-- var  
`-- vendor
```

# リソース制限とキャッシュ

...

resources:

**limits:**

**cpu: "1000m"**

**memory: "1024M"**

...

volumeMounts:

- name: cache-dir

mountPath: "/var/www/ec-cube/var/cache/"

...

volumes:

- name: cache-dir

emptyDir:

**medium: Memory**

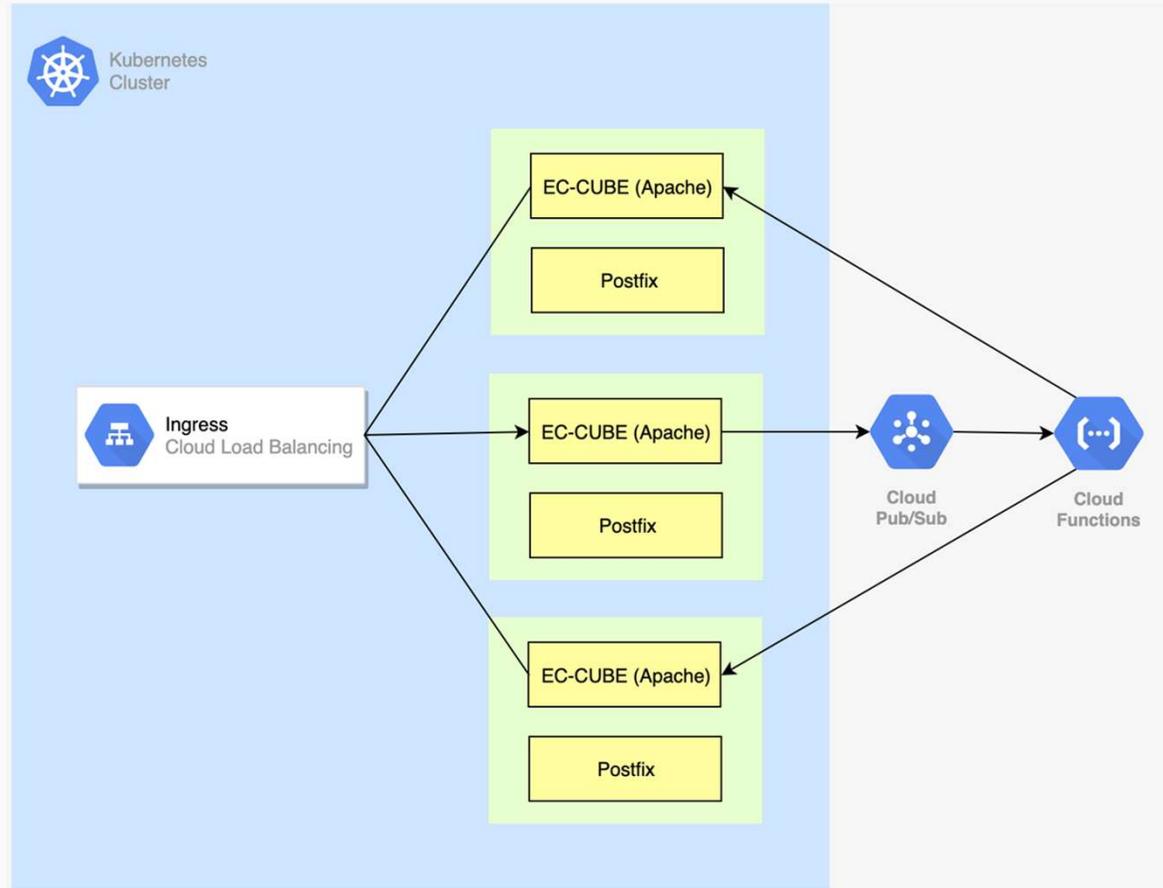
# スケール

- 水平Pod自動スケール
- ノード自動スケール

# キャッシュ/プラグイン

- キャッシュクリアやプラグインインストール時に全Podで同期させる仕組み

# Pub/Sub -> Functions



これから

# これから

- API
  - まずは在庫・受注連携API
- 本体のアーキテクチャは4系では変わらない
- コミュニティやユーザーのフィードバックを