

## NG309

【ドコモ・システムズ株式会社様】  
インフラエンジニア不足の課題への挑戦  
～Pivotal Container Service(PKS)の可能性

---

ドコモ・システムズ株式会社  
クラウド開発部

部長 宮本 英典 様

#vforumjp

POSSIBLE  
BEGINS  
WITH YOU

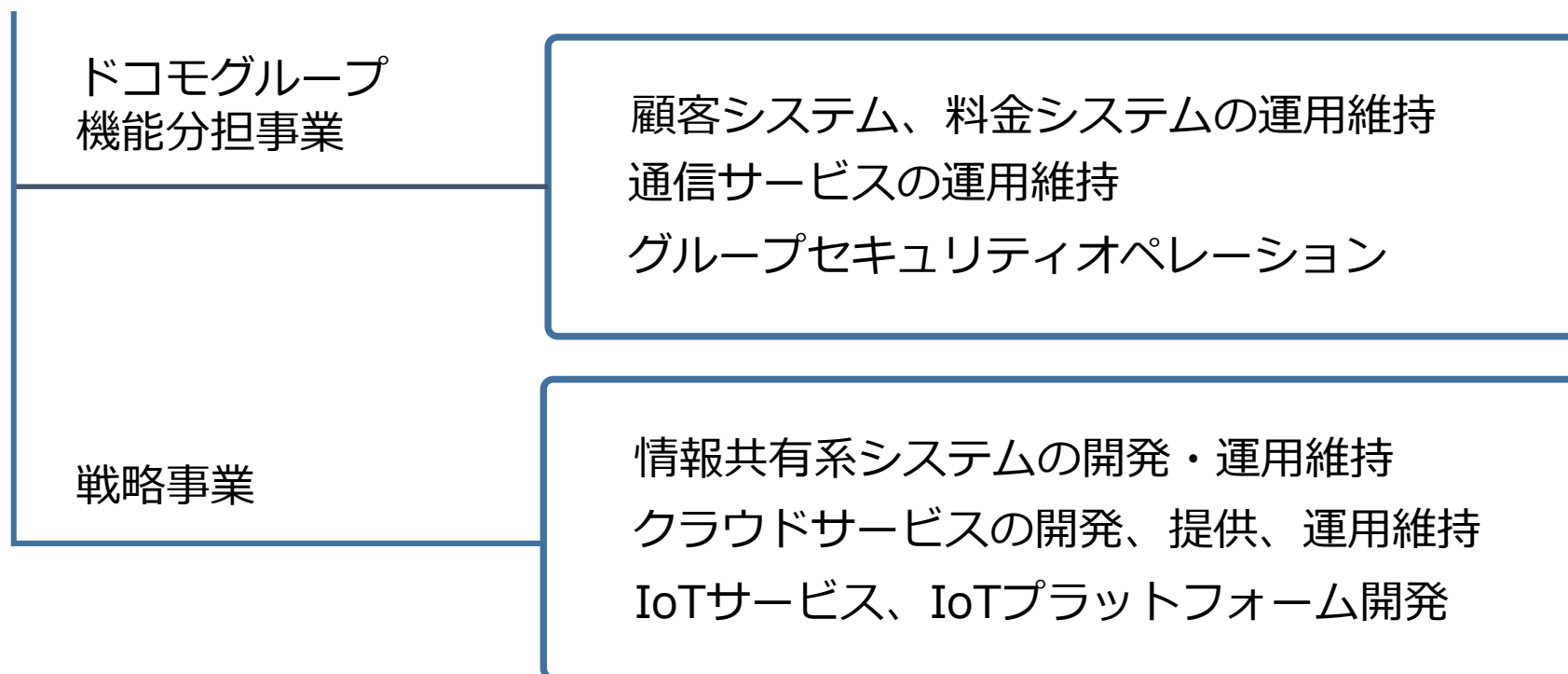
# インフラエンジニア不足の課題への挑戦 Pivotal Container Service (PKS) の可能性

2018年11月14日  
ドコモ・システムズ株式会社  
クラウド開発部  
宮本英典

# ドコモ・システムズ株式会社

企業理念： **「自ら磨き上げるITスキル・自ら蓄積するノウハウ」を駆使し、  
お客さま企業およびドコモグループの経営に貢献します。**

企業情報： 資本金 113.8億円      社員数 777名（2018年3月31日現在）



# 弊社サービスご紹介

働き方改革、BYOD、2020通勤対策・セキュリティ対応



リモートでも臨場感が伝わる。音質と資料共有にこだわったWebミーティング。



仮想デスクトップサービス。在宅勤務、テレワークでも社内と同じPC環境。



セキュリティゲートウェイ。安心のインターネットアクセス。

# 何の話をするか

- PKS/Kubernetesを使う動機
- PKSの価値
- どう取り組んでいるか
- コンテナはどのように使えそうか

# 何の話はしないか

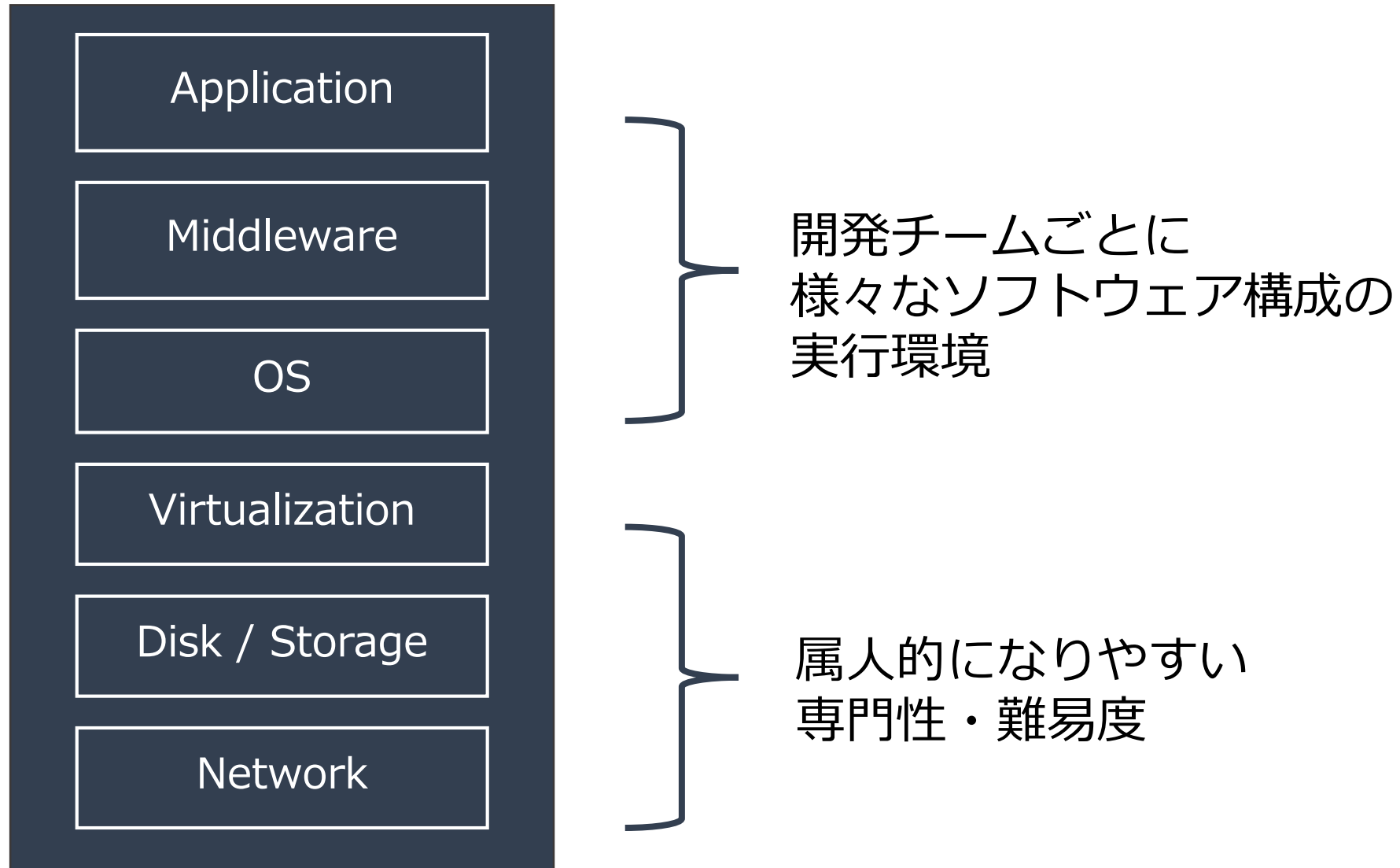
- Dockerの仕組み、操作
- Kubernetesの仕組み、操作
- マネージドKubernetesサービスの比較

# インフラは複雑になりすぎた

- TCP/IP, VLAN, VxLAN
- HTTP, Firewall, IPS, WAF
- DNS, LDAP, RADIUS
- RAID, LU, SSD, thin provision
- VSAN, NSX
- SLB, HA cluster
- Apache httpd, Nginx, Tomcat
- Oracle RDBMS, MySQL, Post
- Hadoop, Cassandra
- Docker
- Ansible, Chef

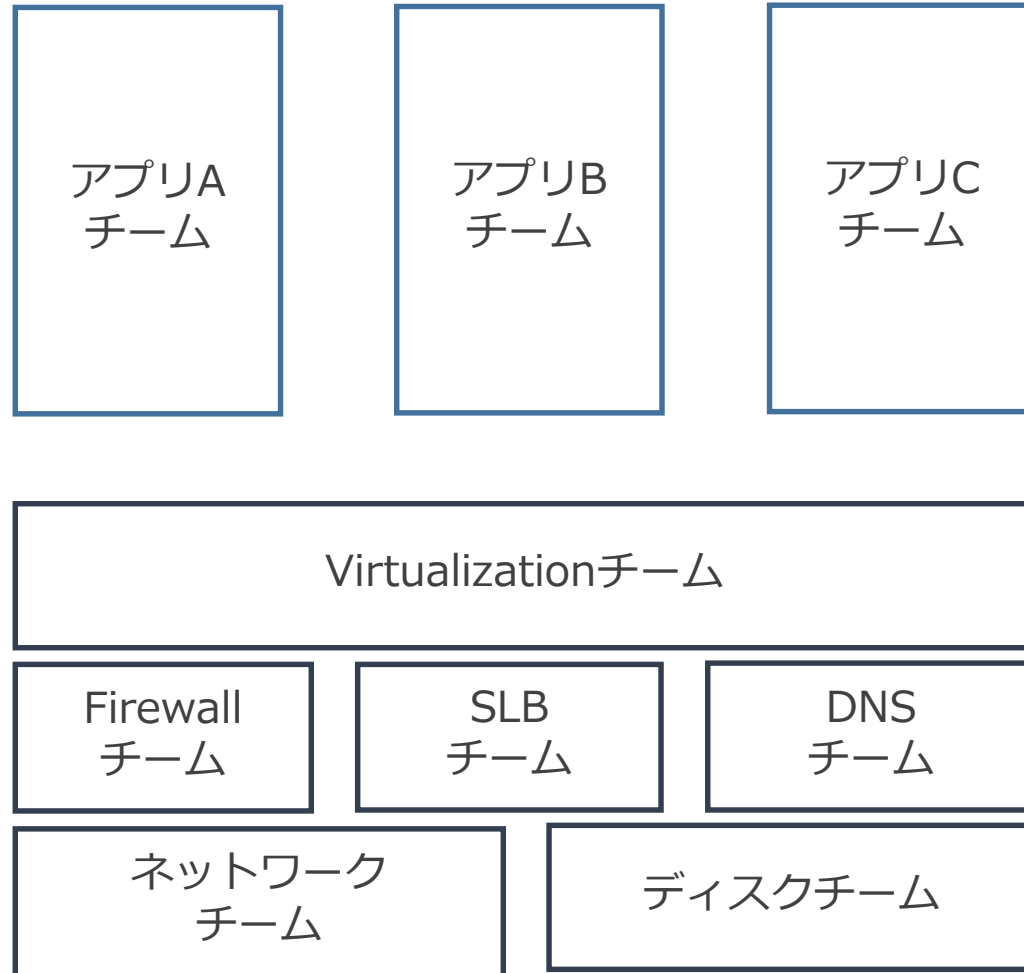
**すべてを理解している  
人はいない**

# インフラ → アプリの実行環境





# アプリ vs インフラチーム



こんなバトルとトラブルが起きる

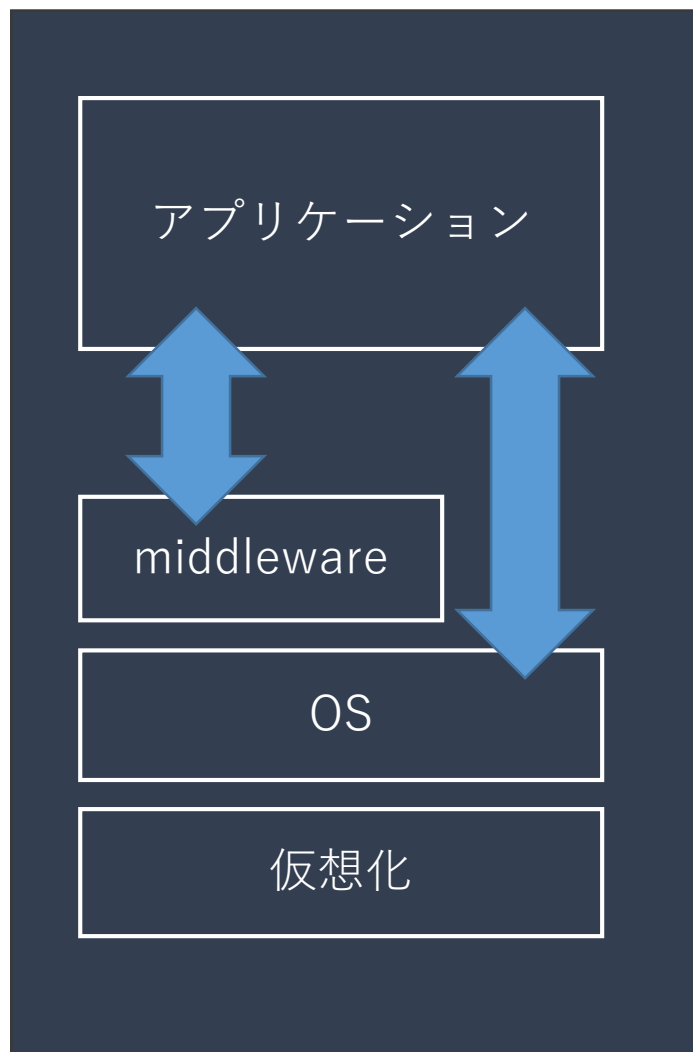
新しいアプリをデプロイしたが、  
Firewallのポートがあいていないよ

Firewallの設定変更は  
週に1回だけなんだけどな

アプリ起動失敗したの？  
vCenter HAの切り替えは正常だよ

サービスがスローなのは  
DBMSのディスク負荷のせいです

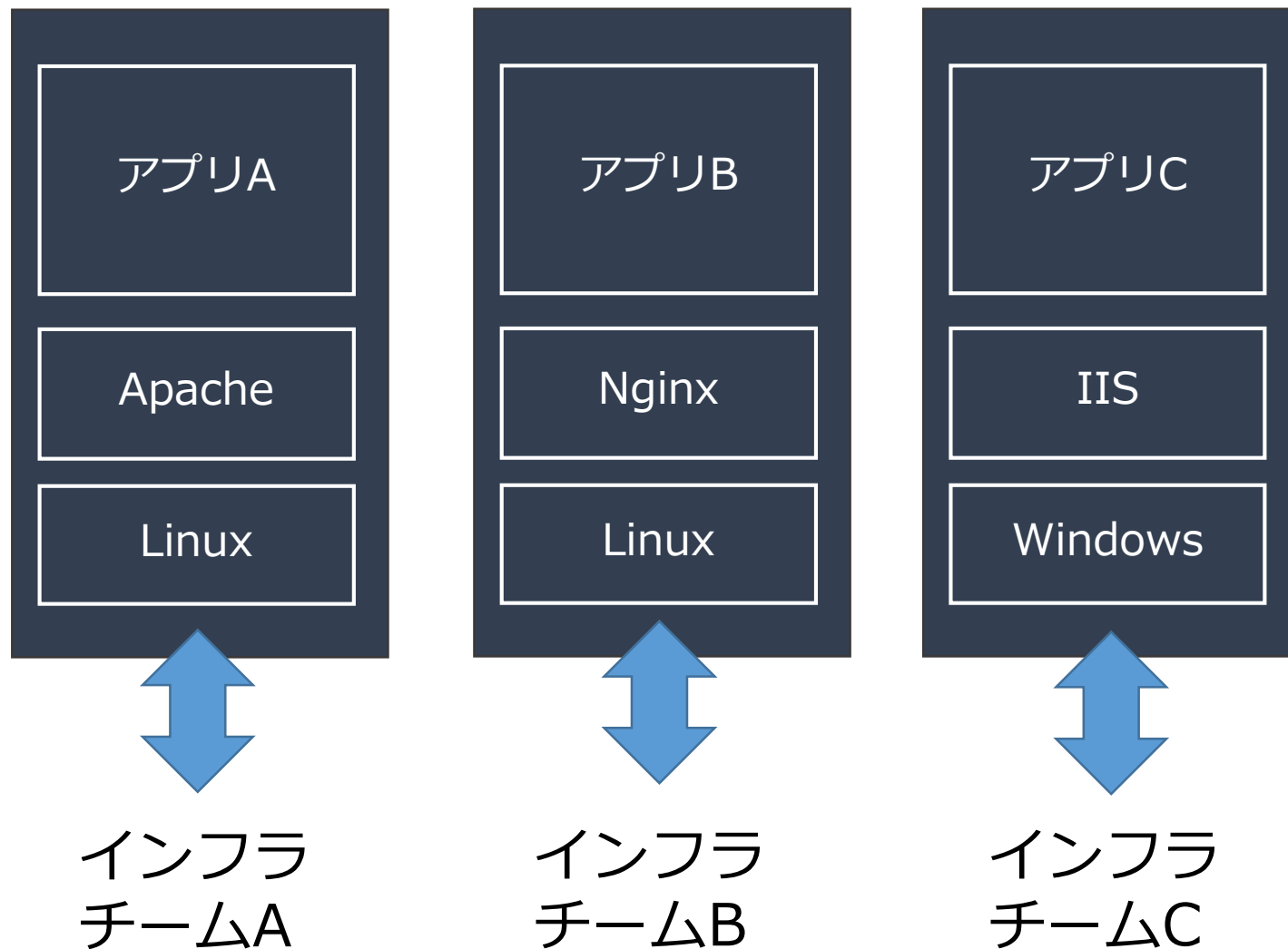
# アプリは実行環境と密結合



- ✓ ミドルウェア固有のネイティブAPIを呼び出している
- ✓ OSやミドルウェアのバージョンアップの都度、アプリケーションの影響調査が大変
- ✓ 外部サービスのIPアドレスがハードコーディングされている
- ✓ 自サーバのIPアドレスで開発環境か商用環境かを判定している

- ◆ 属人化したインフラ設計・維持
- ◆ OS・ミドルウェアのEOSLに合わせアプリの対応が必要

# アプリケーション



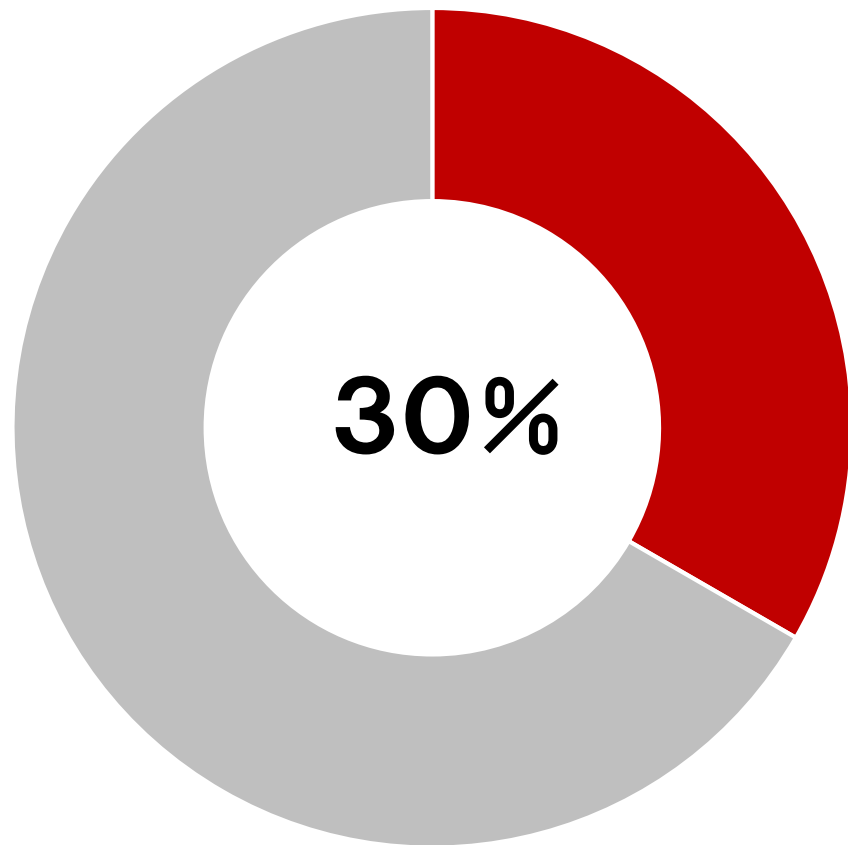
モノリシックなアプリ構造  
縦割りの開発チーム

- ✓ フレームワーク
- ✓ 可用性(冗長化)
- ✓ 負荷分散
- ✓ 性能スケール



アプリ毎に  
異なる方式設計  
異なるソフトウェア  
**異なるインフラチーム**

# アプリ開発：インフラ設計構築



■ インフラ ■ アプリ

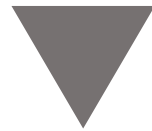
アプリケーションの開発&維持コストの  
30%以上がインフラ関連



サイロなアプリ実行環境による分割損と  
インフラ要員の外注コスト増

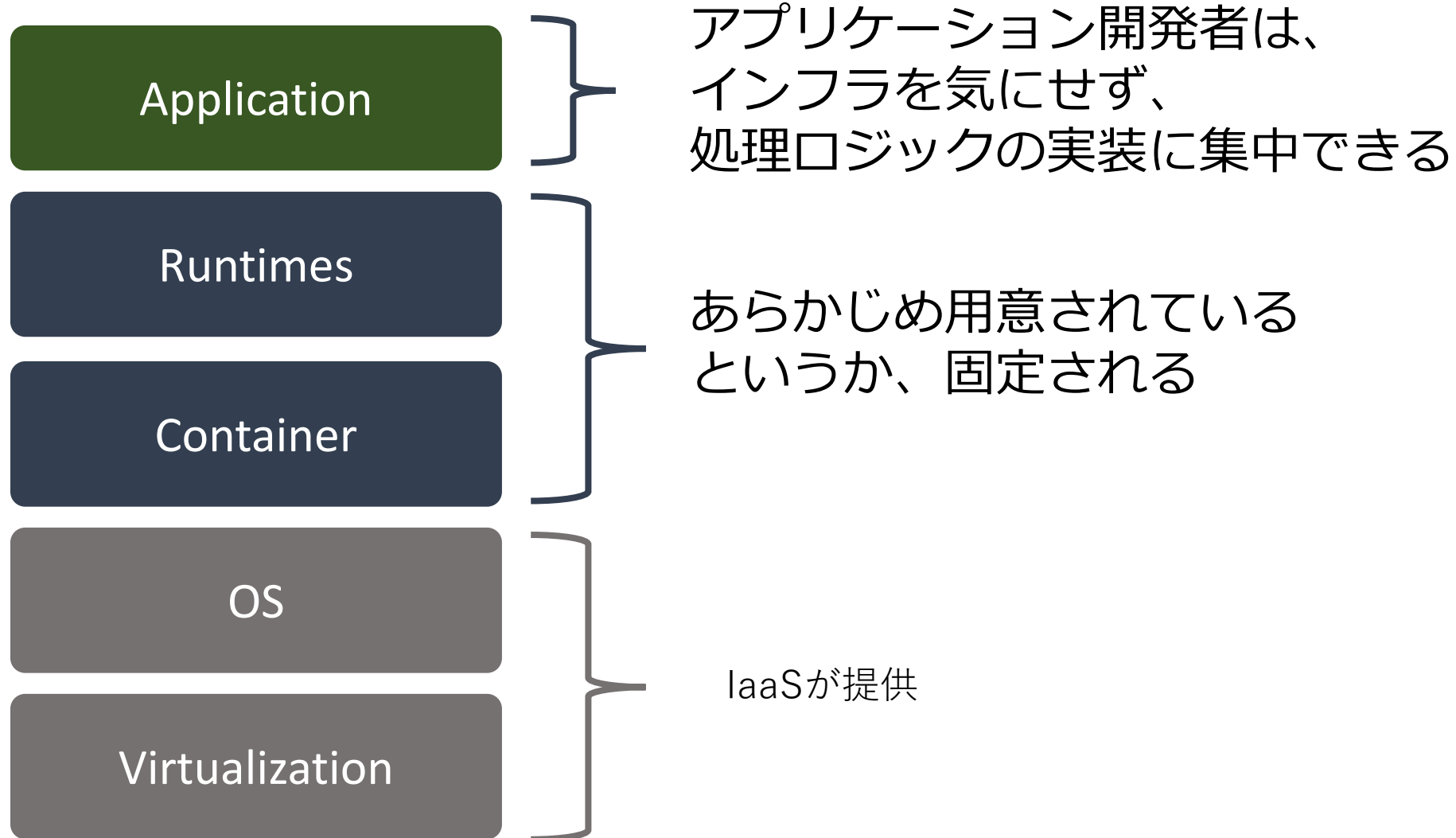
# アイデア

実行環境を固定すれば、  
環境設計のバリエーションがなくなり、  
インフラ稼働を集約できるのでは！



Pivotal  
**Cloud Foundry®**

# PCFあらため、Pivotal Application Service



```
$ rails new  
$ cf push
```



Yay! You're on Rails!



Pivotal  
**Application Service™**

# PCF(PAS)で作ってみよう

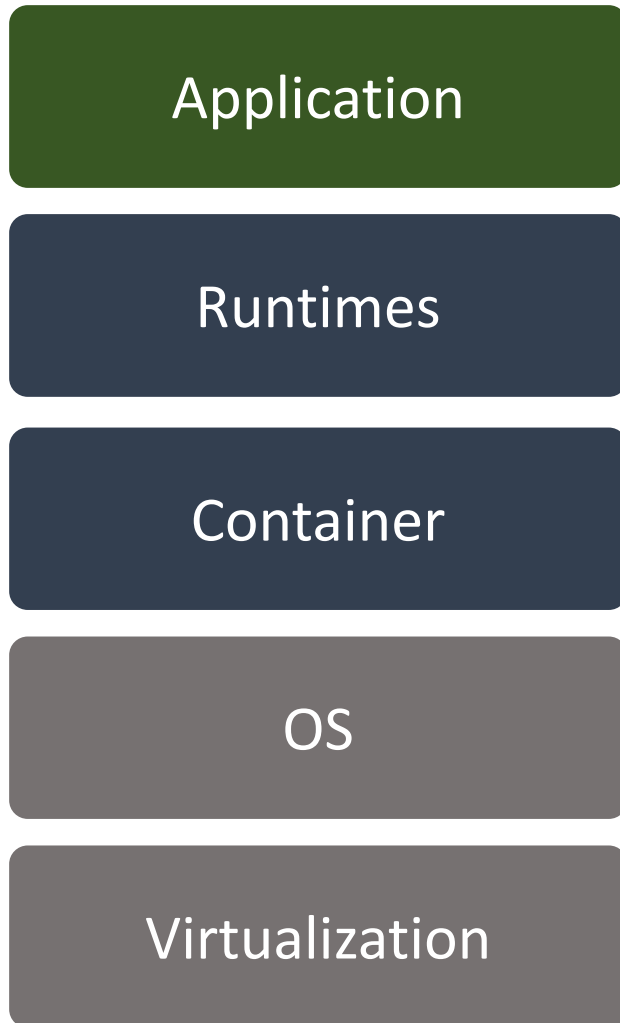
開発委託先

Pivotal CloudFoundryの「仕様書」を  
提示してください



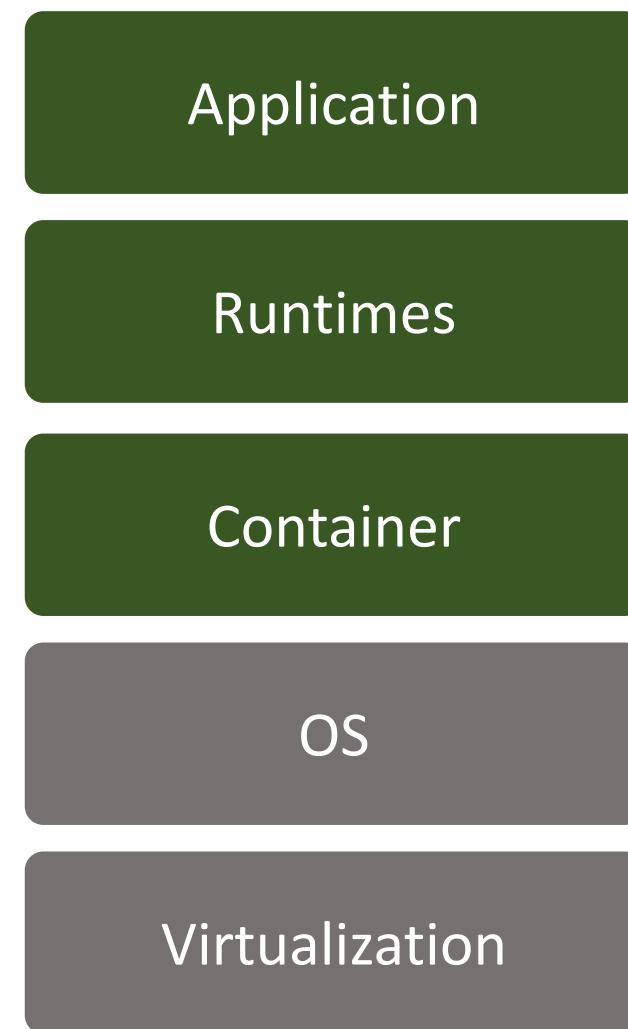
すこし、  
一足飛びに行き過ぎたかも…

## PAS/CloudFoundry

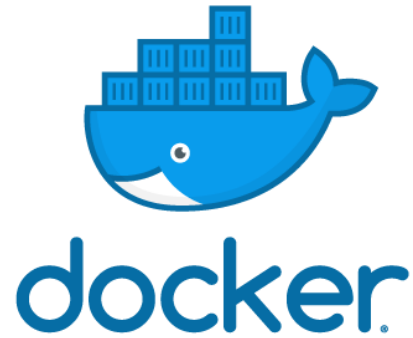


↑  
固定  
↓

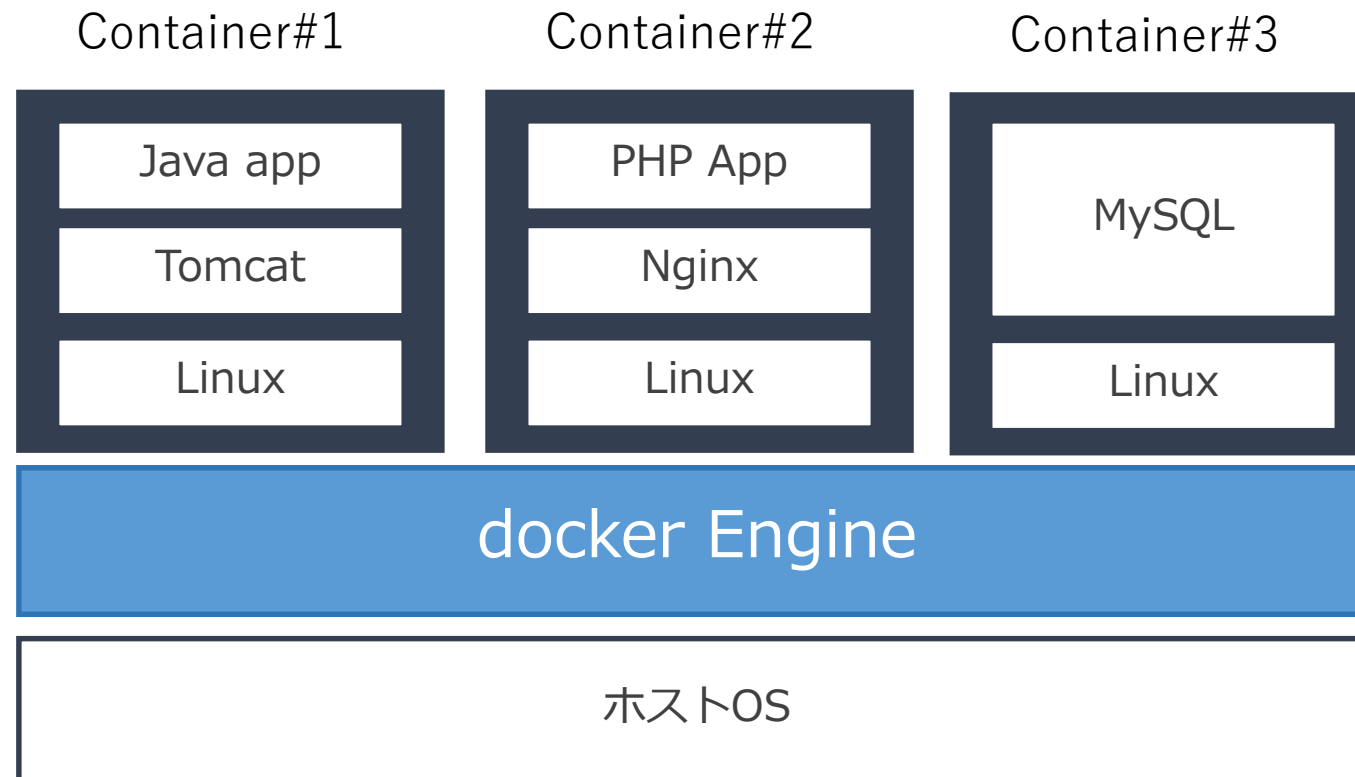
## docker/Kubernetes/PKS



↑  
自由  
↓



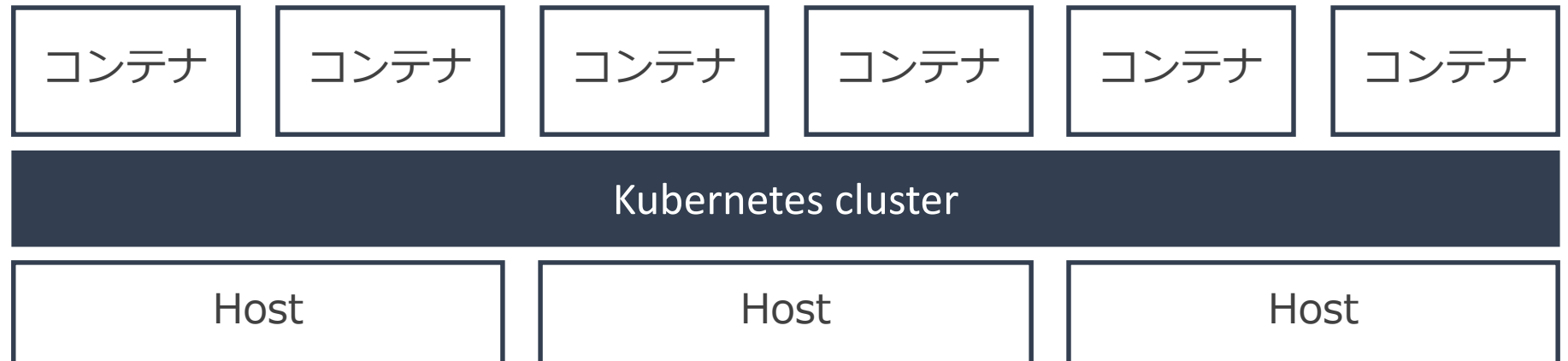
アプリケーションの実行環境を「コンテナ」に閉じ込め、  
再利用性と、可搬性を高める仕組み  
ただし、Linuxに限る





複数のdockerコンテナを商用環境で使うために  
必要な機能を備えたdocker実行基盤

- 複数ホストOSでのクラスタリング
- 可用性・耐故障性
- 負荷分散
- サービスディスカバリ etc





**kubernetes**

素のままのKubernetesでは難しすぎる  
インフラ稼働と専門スキル減らそうと  
思ったのに、逆に増えそう！？

**kubelet**

**scheduler**

**etcd**

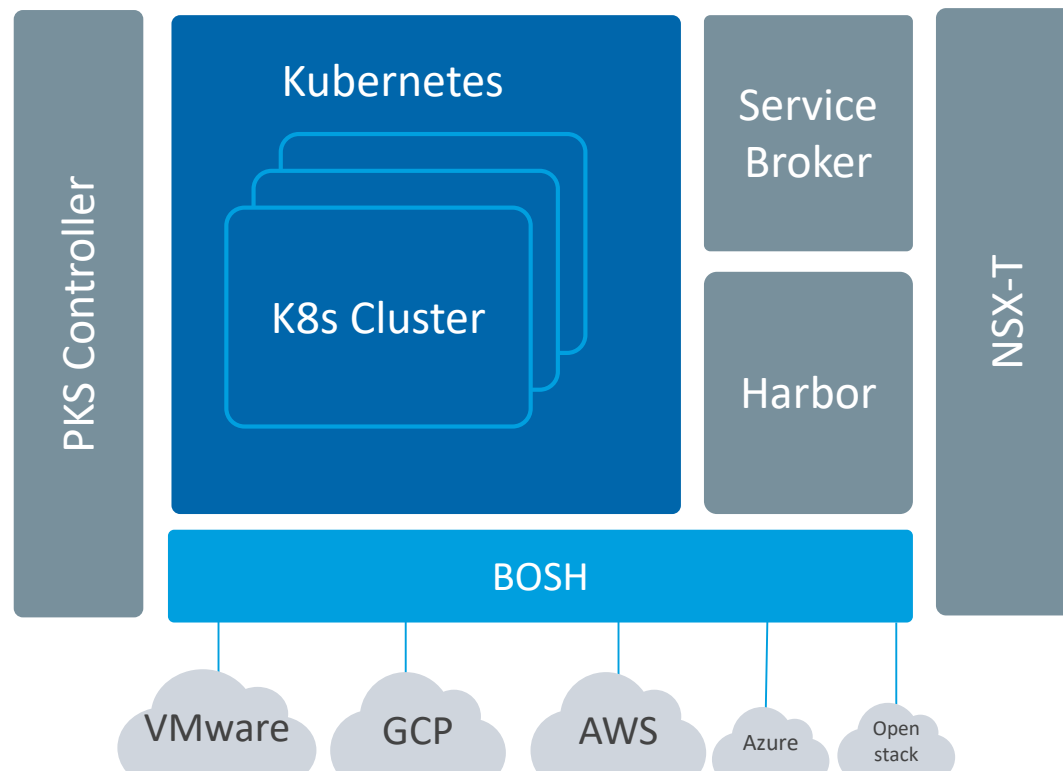
**apiserver**

**cloud-controller**

**kube-proxy**



# Pivotal Container Service™



※AzureとOpenStackは今後対応予定らしい

- Kubernetes Cluster構築が簡単
- リポジトリを内蔵
- ネットワーク&セキュリティ管理
- IaaS連携

# Harbor

English

admin

Projects &lt; Repositories

library/mariadb

Info Images

SCAN

COPY DIGEST

+ ADD LABELS

× DELETE

☐

Tag

Size

Pull Command

Vulnerability

Creation Time

Labels

☒

10.3-signed

109.21MB

8/17/2018, 4:27 AM

☐

10.3

109.21MB

8/17/2018, 4:27 AM

1 - 2 of 2 items

Vulnerability Severity: **Medium**

11 of 111 packages have known vulnerabilities.

4 Medium

7 Low

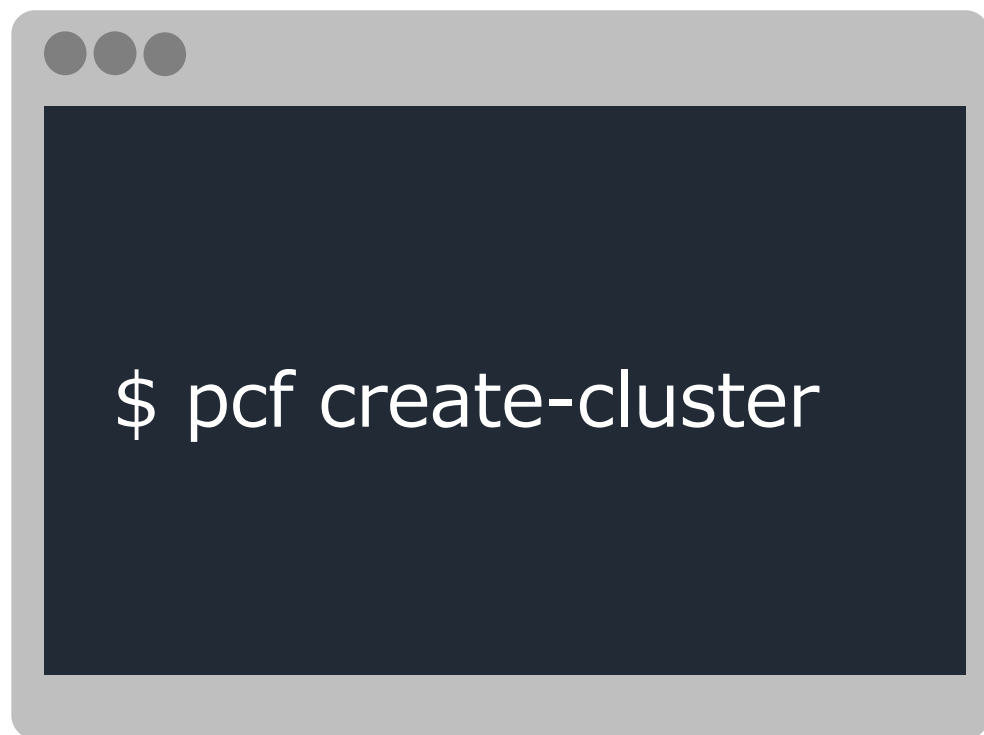
100 None

Scan completed time: 08/20/2018 16:43:06

EVENT

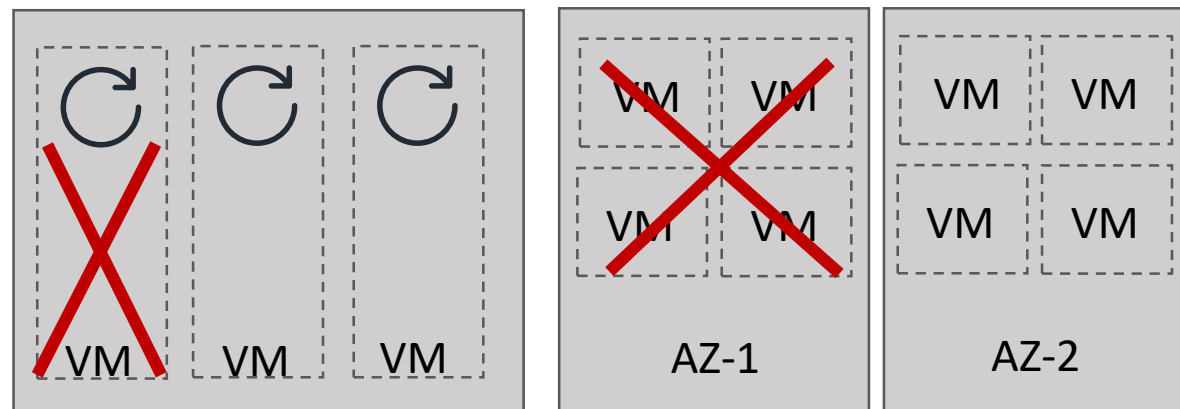
# BOSH

コマンド 1 行で、  
k8s クラスタを作成できる



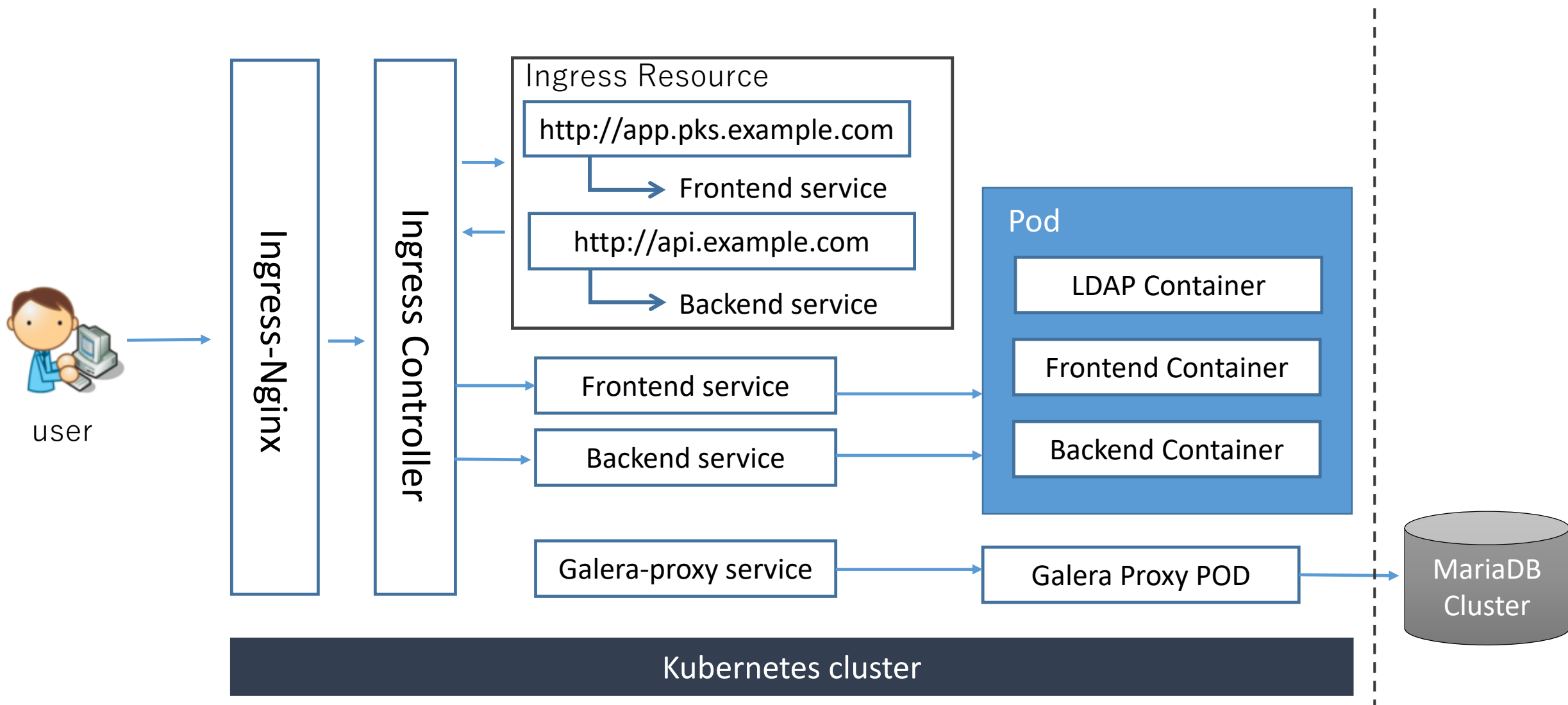
IaaS層を隠蔽し、  
面倒を見てくれる

## Kubernetes Cluster

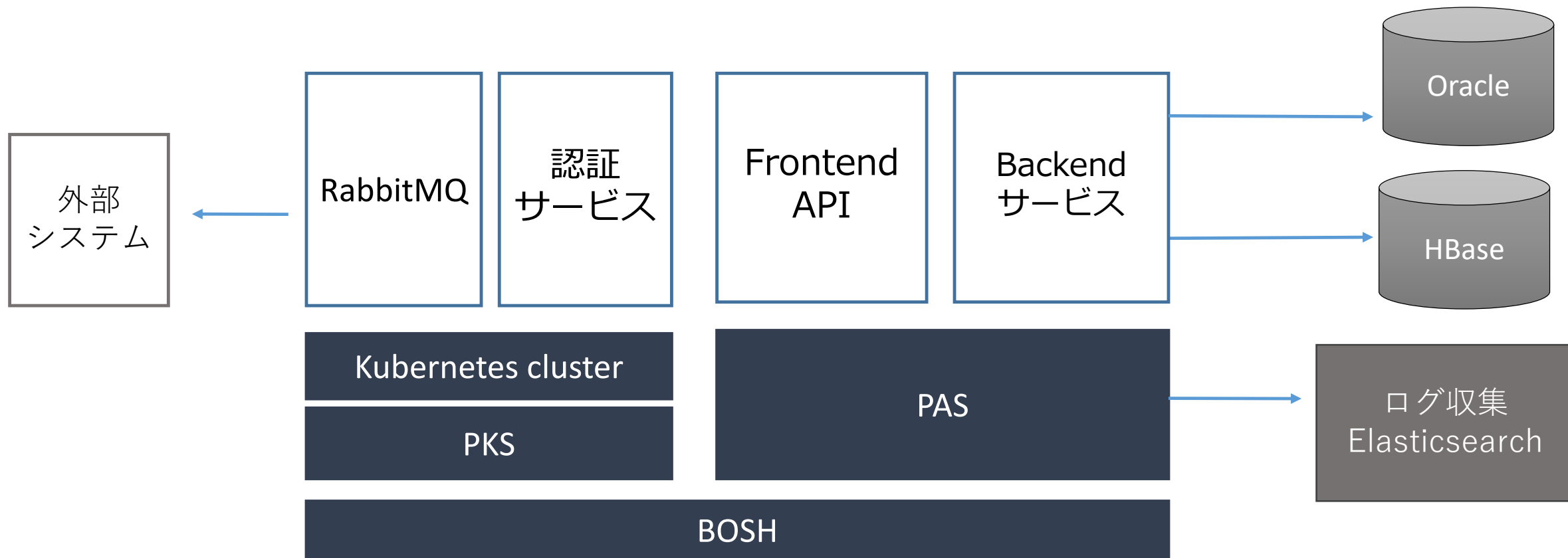




# PKSでアプリケーション実装



# PKSとPASでアプリケーション実装



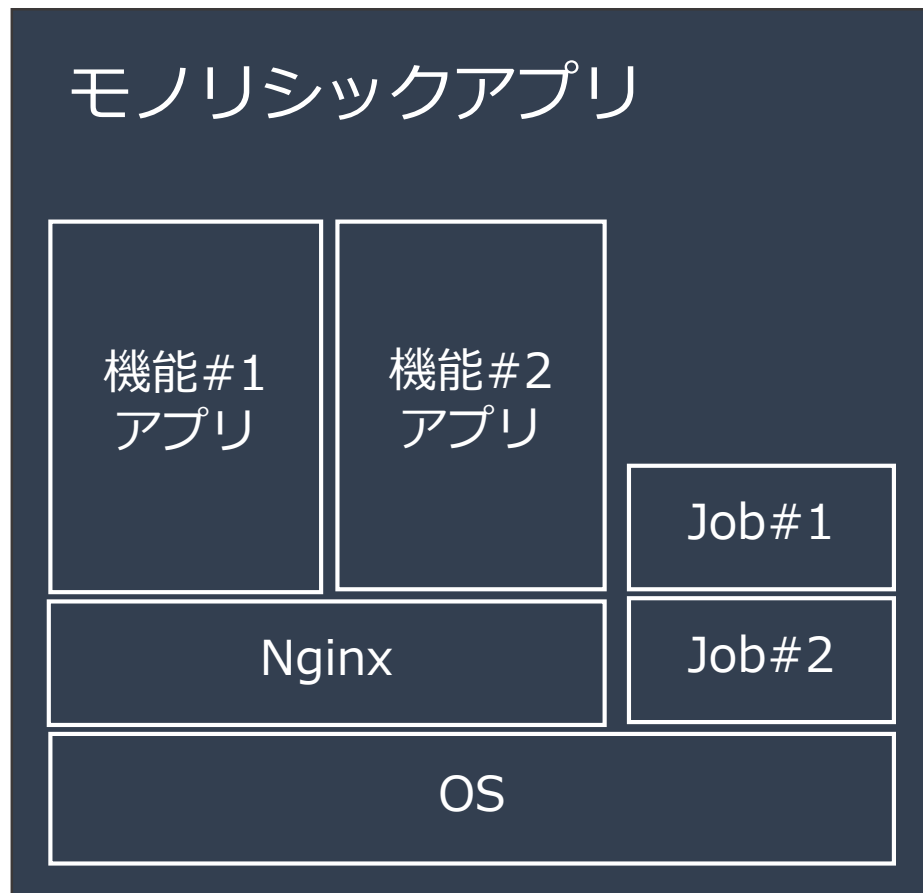
# PKSを使ってみて

- コマンド 1 つでKubernetesクラスタができるのは最高
- Kubernetesクラスタは同じか? → たぶん同じ
- NSX-Tは少し設定に癖がある → ドキュメントを良く読む
- BOSHによるk8sバージョンアップは?  
→ いろいろあるが、生のKubernetesの問題?
- 素のk8sやAnsibleゴリゴリやるのは無理なので、  
維持は圧倒的に楽
- パブリックIaaS環境へのマイグレも簡単になるかもという期待
- YAML地獄は変わらない

# インフラは効率化したか？

新しくアプリケーションを作る場面では、  
確実に効率化する（可能性を感じた）

# 既存アプリケーション



# Lift and Shiftは現実解か



## 必要悪の遠回り

- Excelの構築手順書→ Infrastructure as Code
- 環境依存の邪魔ものが何かがわかる

# 実際のアプリはこんな感じ

- COBOLで書かれている
- Javaコンテナサーバーに仮想環境でのライセンス問題がある
- Windows ServerとSolarisとHP-UXで動いている
- ジョブスケジューラでFTPしまくっている

# はじめることが大切

## IT技術の進歩に静止点はない

手を動かして始めてみれば、  
漠然とした不安が、現実の課題に変わる



# まとめ

- Kubernetesを使うなら、(今は) OSS版はつらい  
PKSはひとつの選択肢である
- Pivotal Container Service、  
Pivotal Application Serviceを活用することにより、  
インフラ維持を効率化し、  
アプリケーション開発に集中できる可能性を感じる

いつか、あたりまえになることを。

