

オープンソースから見る エッジコンピューティングの現在と未来

レッドハット株式会社

クラウドスペシャリストソリューションアーキテクト

Edge Strategy Tech Lead

小野 佑大

自己紹介



Red Hat 小野 佑大

 @pinkink_yudai

現在

- OpenShiftのソリューションアーキテクト
- エッジコンピューティングを推進

経歴

- 通信キャリアにて5Gやクラウドの技術戦略
- 2018年頃からエッジコンピューティング

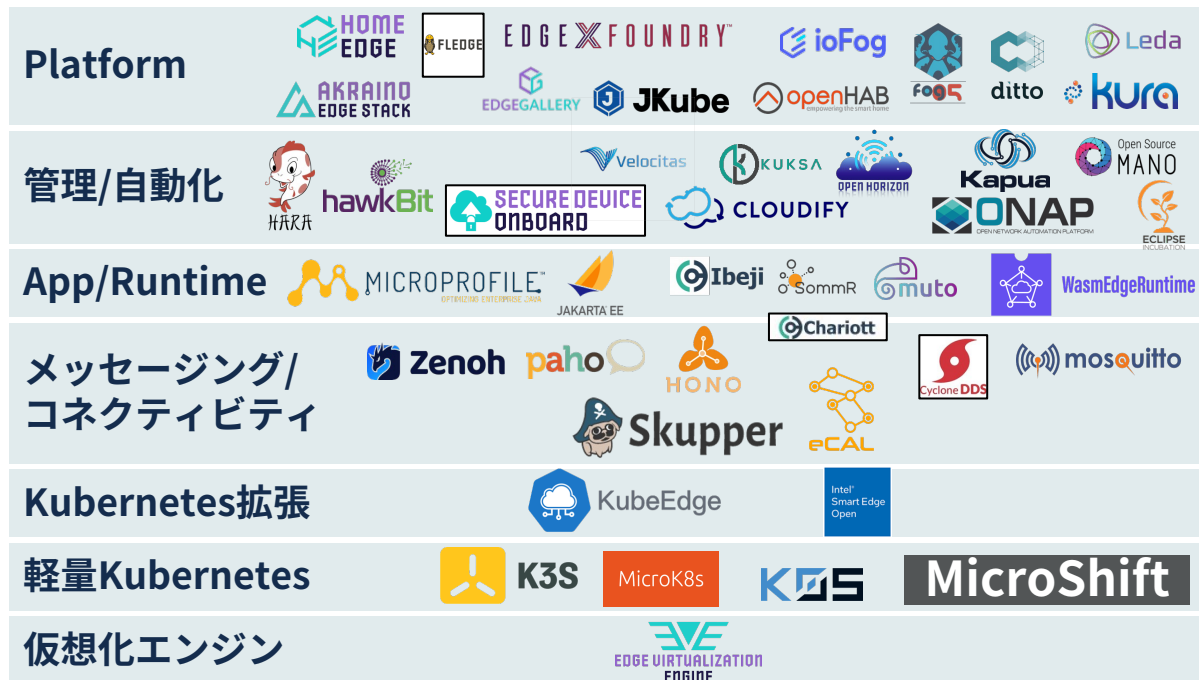
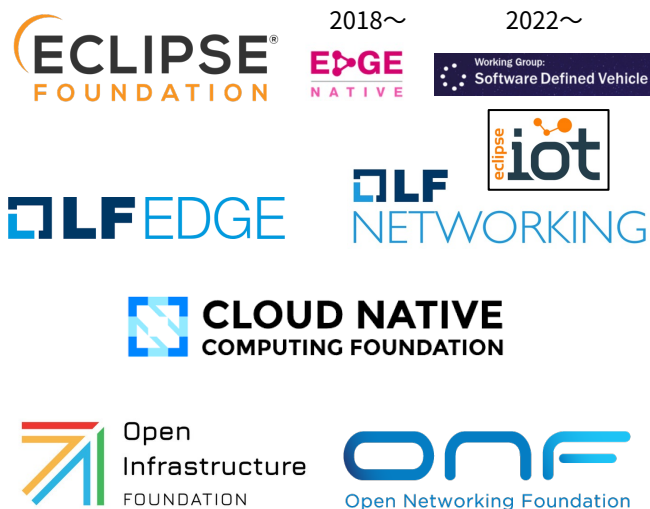
Edge Computingとは

アプリケーションをセンサー等のデータ生成元に近い場所で実行



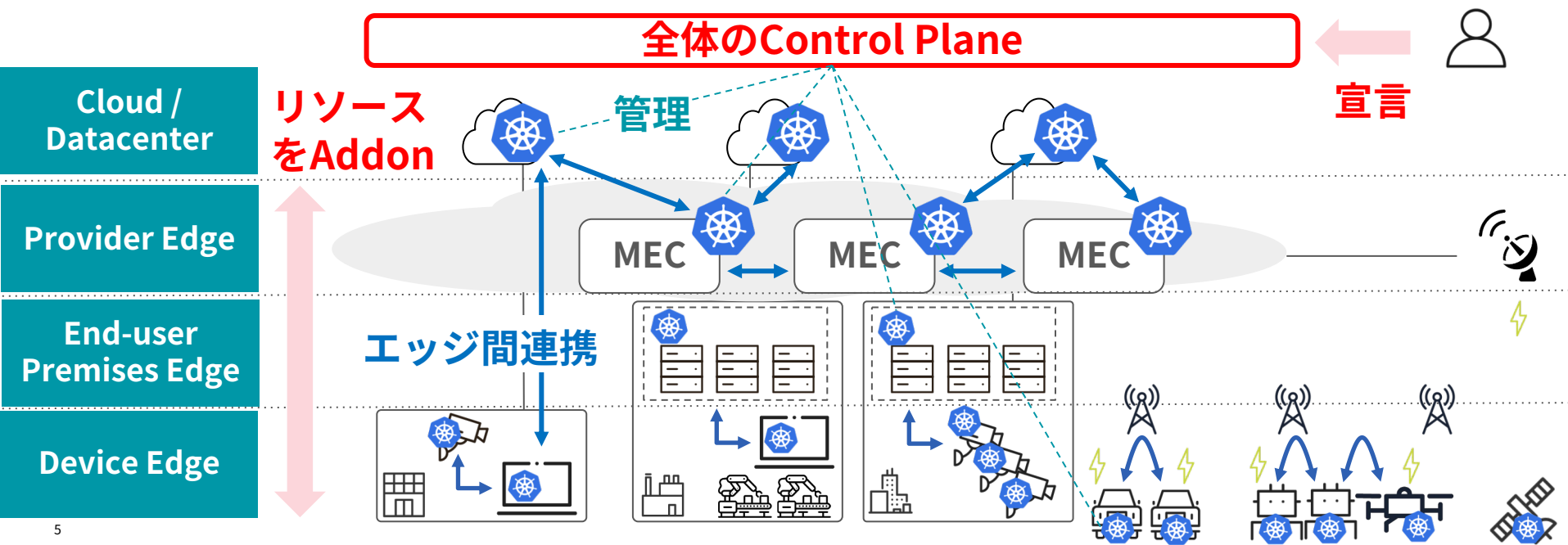
エッジコンピューティング関連のオープンソース

軽量版クラウドネイティブ × エッジの管理の効率化



エッジコンピューティングの方向性

エッジとクラウドをまとめてセルフサービスで



EdgeOps

DevOpsの原則をエッジに適用しつつエッジ特有の問題解決

EdgeOps

Adapting DevOps for the Edge

DevOps Principles

Short Lifecycle,
Collaboration, Continuous
Integration and Delivery
(CI/CD), Microservices,
Infrastructure as Code

Challenges

- Latency
- Bandwidth
- Resiliency
- Data sovereignty

NW接続に伴う可用性・性能品質の劣化を低減し、かつデータ主権を保護できる

Characteristics

- Long lifespan
- Heterogeneous
- Constraints
- Connectivity

長期のライフスパンの多種多様なハードウェアに対応できる

Deployment

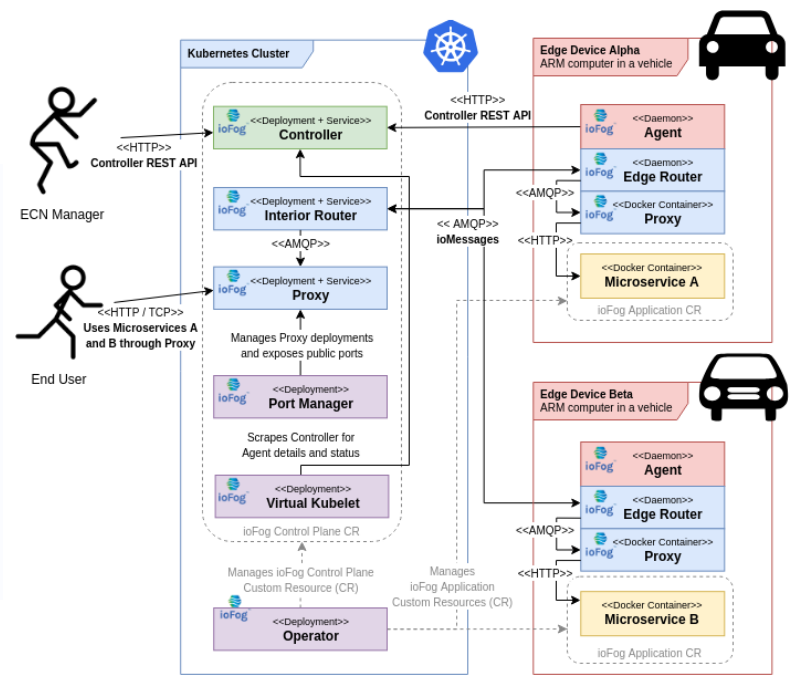
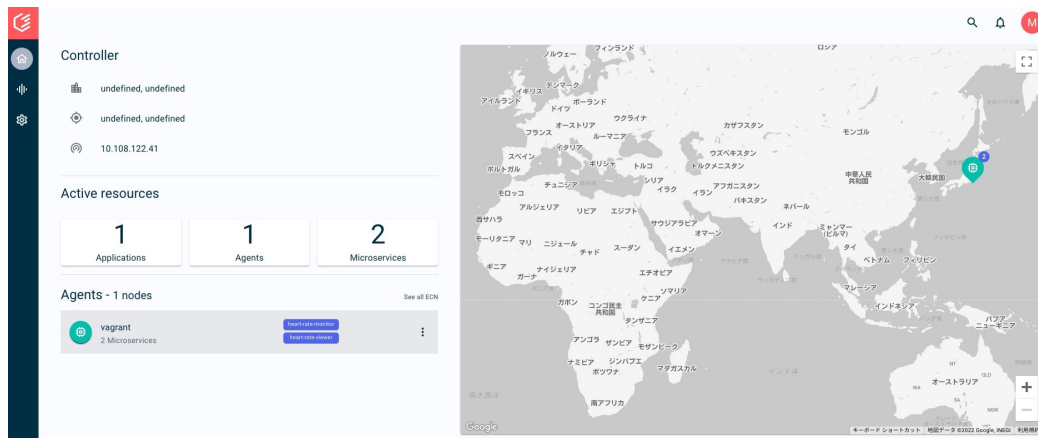
- Workloads
- Artifacts
- Strategies

物理的・NW接続上の制約を考慮して、現場への技術者の派遣を極力回避できる



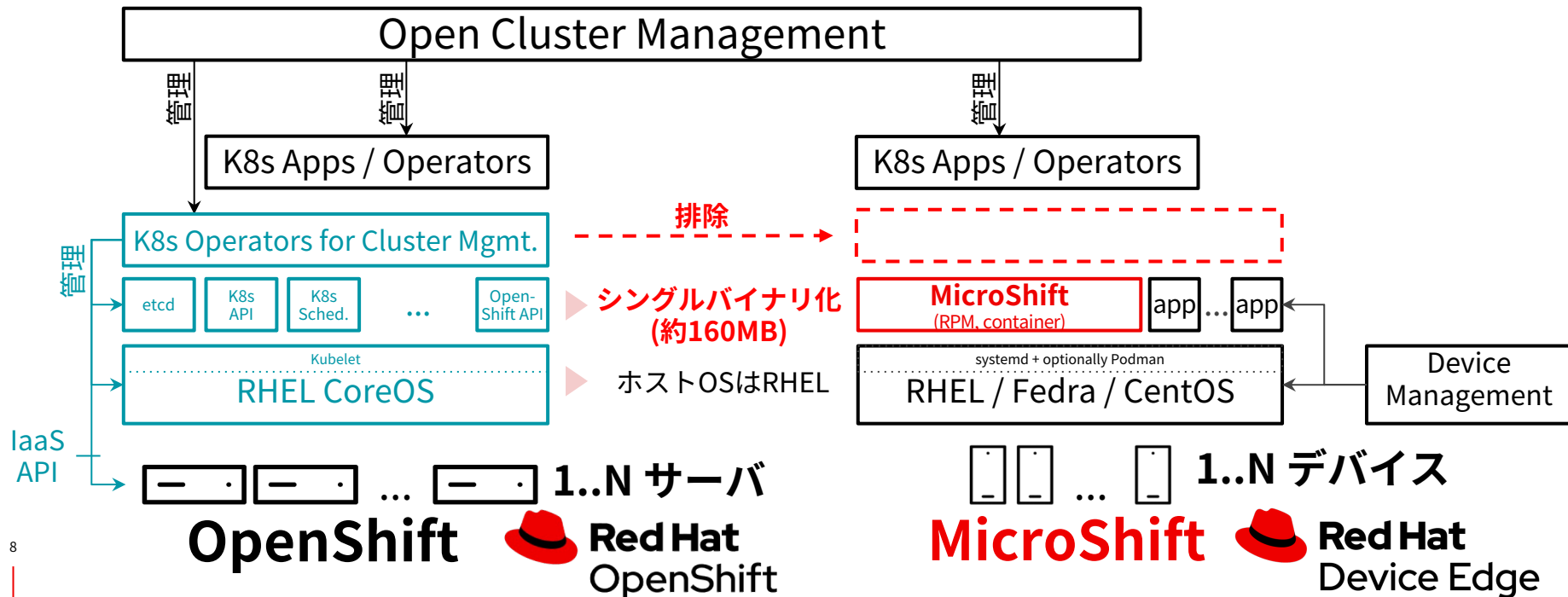
エッジへのマイクロサービスの展開を効率化しよう

- Edgeworksが開発を推進（商用でも利用）
- エッジのソフトウェア開発をクラウド同等に



MicroShift

エッジデバイスでもOpenShiftを活用



Open Cluster Management

アプリケーションのFleet Management



マルチクラスタ運用を効率化

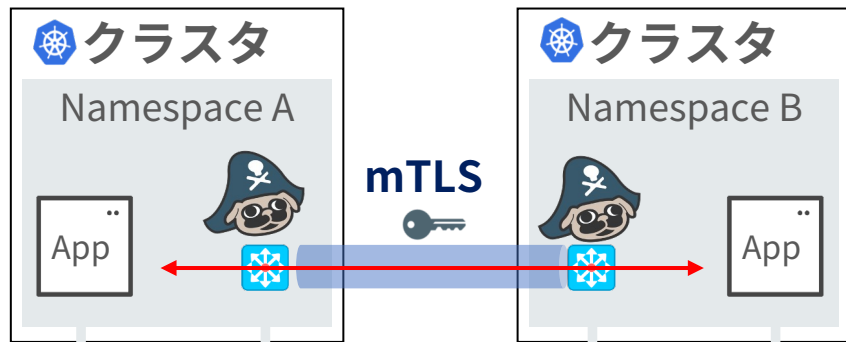
- ポリシーをアドオンとして追加（自作も可能）
- アプリケーションのライフサイクル管理
- ラベルベースのクラスタスケジューリング

Hub/Agent間で状態を同期

- Agent Podが起動
- Hubで宣言された状態がAgent経由で同期



AMQPメッセージベースのマルチクラスタ・サービスメッシュ



Service app:8080へReq
(TCP/gRPC/HTTP)

app:8080のEndpoint
はSkupper Router

AMQP Message Routing

“app:8080” をexposeするRouterへメッセージ交換

- ・宛先のディスカバリはOSPF相当のアルゴリズム
- ・同じ宛先がある場合は負荷分散

Qpid Dispatch Router

片方向からのセッション
接続で双方向通信が可能

Namespace単位で
一般権限でセットアップ

様々なトポロジーをサポート
(P2P, Hub&Spoke)

まとめ

▶ エッジコンピューティングの現在
クラウドネイティブ技術をエッジでも使おう！

⇒ エッジの管理の効率化とクラウドネイティブ技術の軽量化

▶ エッジコンピューティングの未来
クラウドネイティブやDevOpsの原則がエッジへも

⇒ クラウドと同様にエッジ環境のセルフサービス体験を広げる



Red Hat