

2022.2.22 Open Mobile Network Infra Meetup #6

# コンピューティングが ネットワークに溶け込む世界

MEC (Multi-access Edge Computing) 関連の標準化動向

Yudai Ono

# Introduction

## 小野 佑大 (Yudai Ono)

所属：Red Hat

- ✓ 通信キャリアにて主にモバイルコアやNFVなどの技術戦略を担当
- ✓ 2021年12月よりRed HatにてOpenShiftのSolution Architect
- ✓ **エッジコンピューティング市場を盛り上げたい**



@pinkink\_yudai

Qiita @ydo

# 本日の内容

- MECの標準化動向の**今**を俯瞰する
- MEC関連の**次**の動向を俯瞰する
- MECの**未来**の姿を想像する

# 参考

本日は話す内容の一部は、Qiitaの以下の記事にてまとめています。

ぜひ LGTM をよろしくお願いします！

<https://qiita.com/ydo/items/87945c14359ca2100224>

 @ydo  
投稿日 2022年01月12日 更新日 2022年01月14日 9287 views

## コンピューティングがネットワークに溶け込む未来～Multi-access Edge Computing(MEC)の標準化動向概要(2022年1月時点)～

 ネットワーク, モバイル, 5G, EdgeComputing, エッジコンピューティング



# 本日の内容

- MECの標準化動向の**今**を俯瞰する
- MEC関連の**次**の動向を俯瞰する
- MECの**未来**の姿を想像する

# MEC関連の主な標準化団体

## ETSI ISG MEC

ISG=Industry Specification Group  
MEC=Multi-Access Edge Computing

エッジコンピューティングの  
プラットフォーム(実行基盤)を検討

参照アーキテクチャ・API

## GSMA

GSMA=GSM Associate  
GSM=General System  
for Mobile Communications

MNO間のプラットフォーム相互接続  
やビジネスモデルの検討

キャリア間の協調

## 3GPP

3GPP=3rd Generation Partnership Project

モバイル通信システムを介して  
MECへ接続するための方式を検討

モバイル通信規格

標準化された仕様を参考に実装・検証

オープンソース(OSS)



LF EDGE

LF NETWORKING



CLOUD NATIVE COMPUTING FOUNDATION

# ETSI MEC リファレンスアーキテクチャ (2014~)

端末アプリとMobile edge orchestratorのAPI連携の中継

## MECのプラットフォーム機能

- Data Plane制御やDNSのハンドリング
- API提供(無線情報や位置情報等)、など

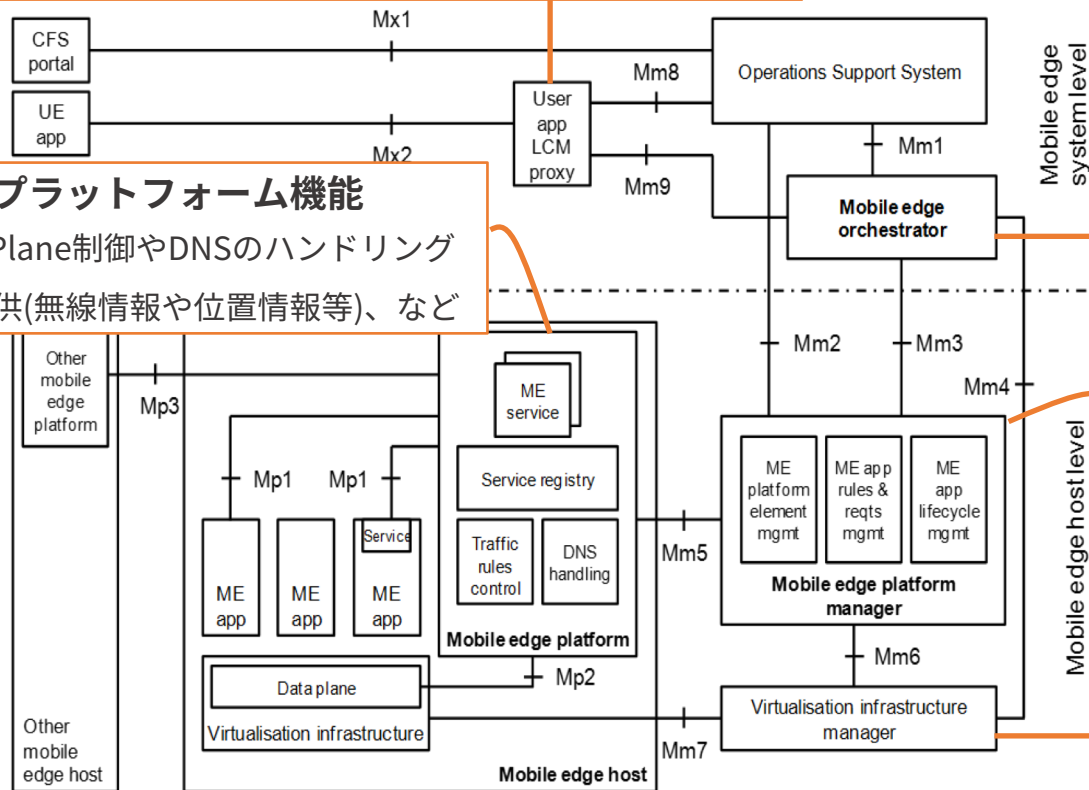
## 中央集権型のコントローラ

- 端末要求に応じたME App配置制御
- MEC全体のリソース管理
- 遅延等の何かしらの条件に基づくME hostの選択、など

## ME PlatformやME appの管理

- ME appのライフサイクル管理
- ME appのルールの管理(認可、トラフィックルール、DNS設定…)
- 構成管理、など

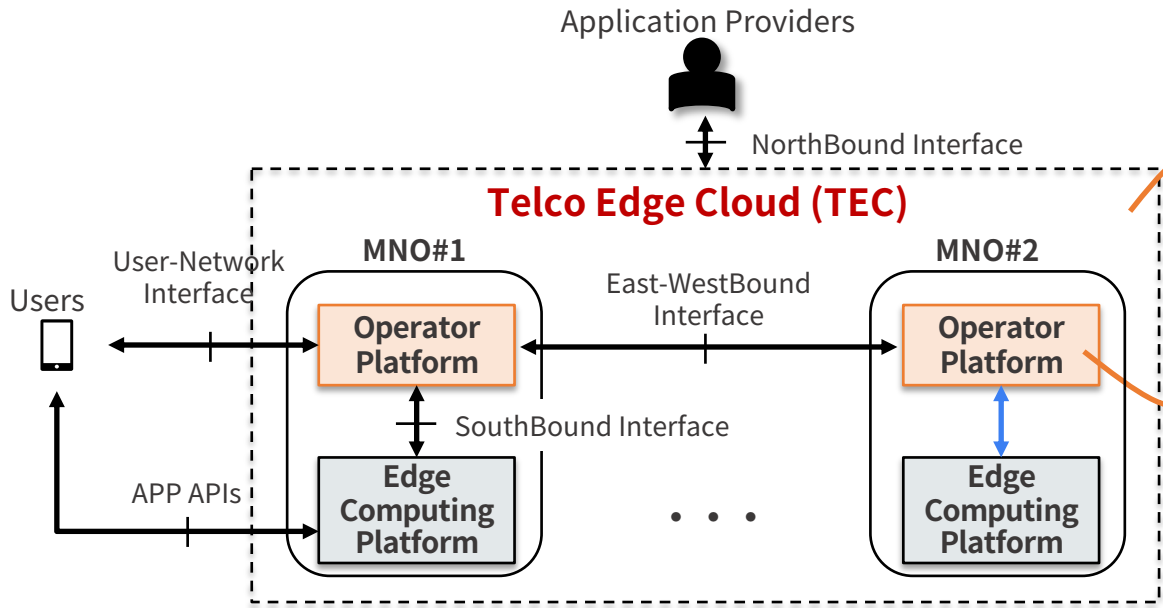
## 仮想化基盤の管理(VIM)



# GSMA Operator Platform (2020~)

(ETSI MECがうまく機能せずMNO間でMECの実現構成がバラバラに)

**相互運用性を高めてアプリ開発者が使いやすいプラットフォームを目指す！**



## Operator Platformのビジネス検討

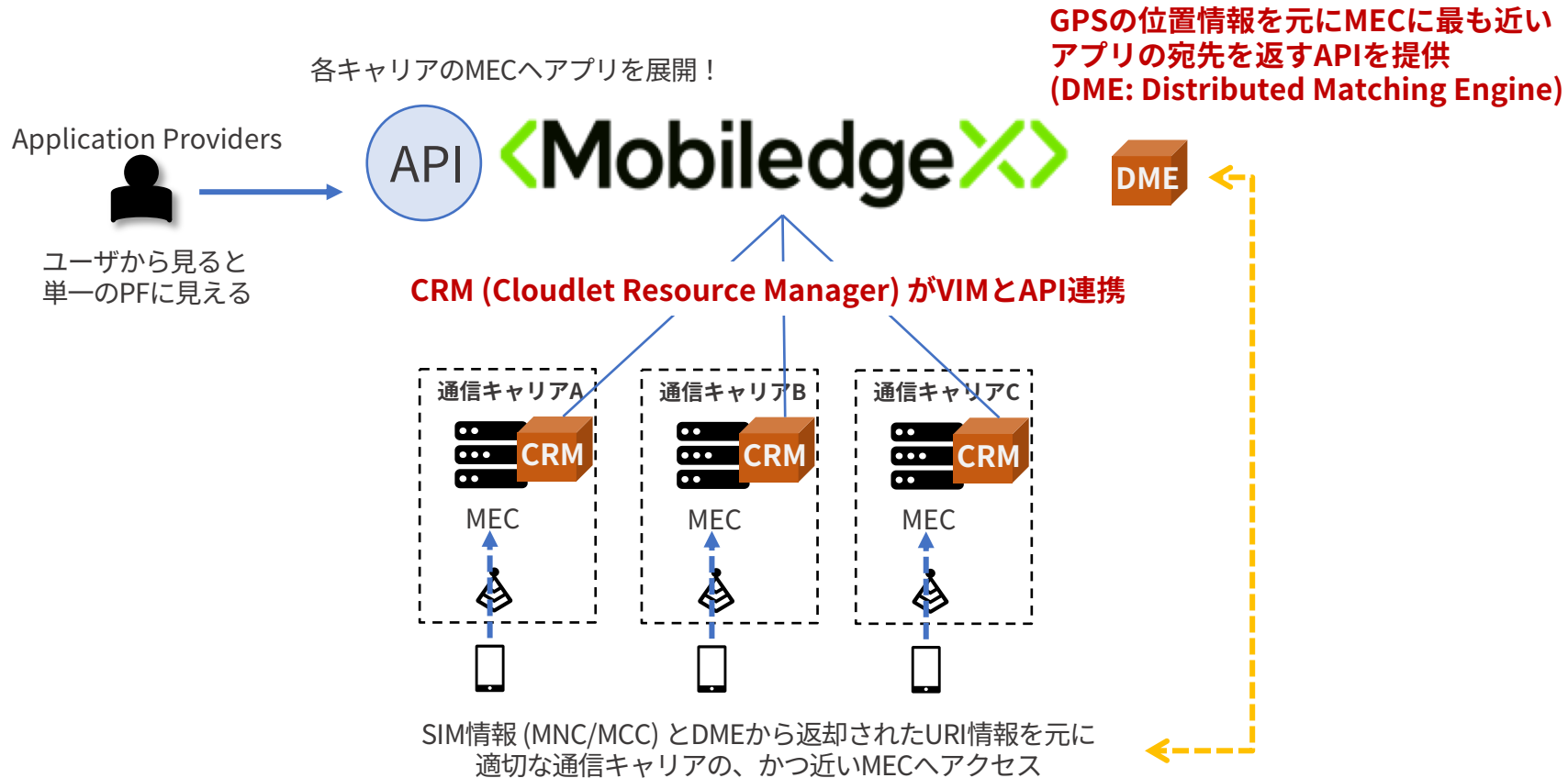
各オペレータが提供するエッジクラウド (NaaS, PaaS, IaaSなど)をグローバルへ単一のプラットフォームとして提供

## 各MNOのMECのFederation

- ネットワーク相互接続による実現、オーケストレーションレイヤのAPIのaggregateによる実現…
- ETSI MECでもこの動向を踏まえ、V2X文脈でMNO間のMECの相互接続が標準化(Inter-MEC)

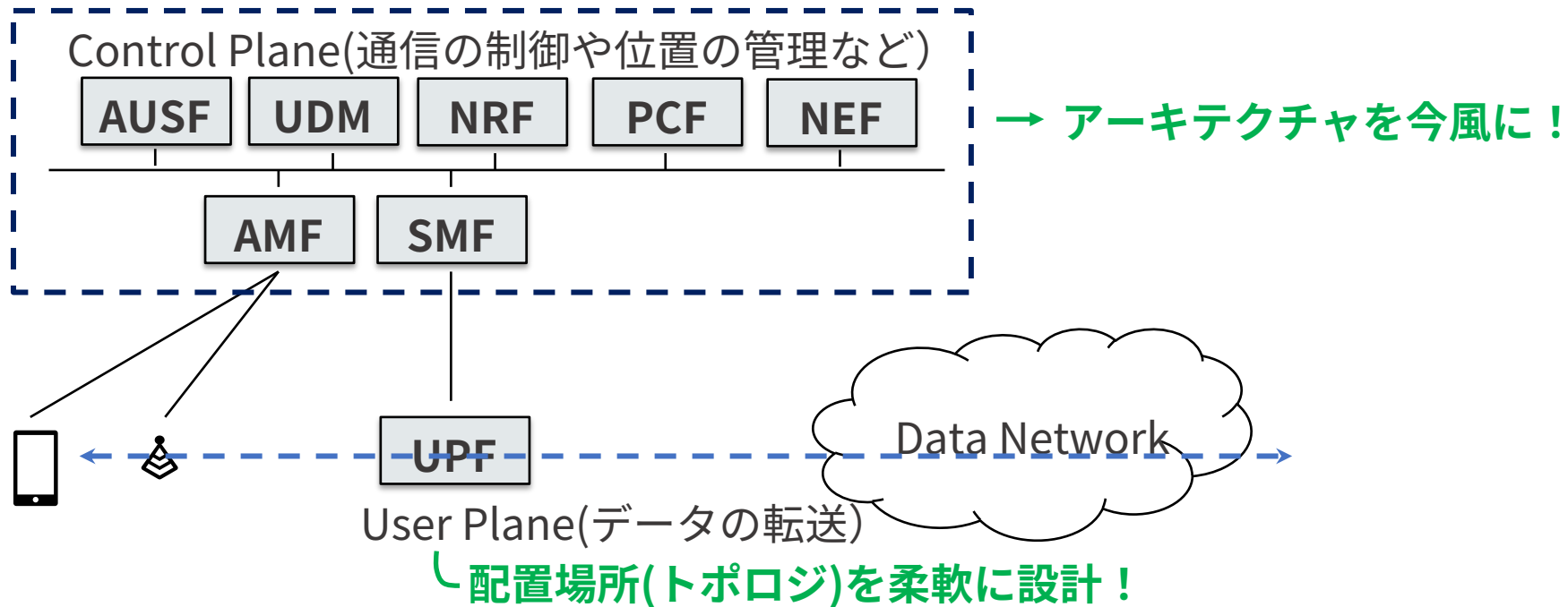


# Operator Platform界隈でよく出てくるMobileedgeX



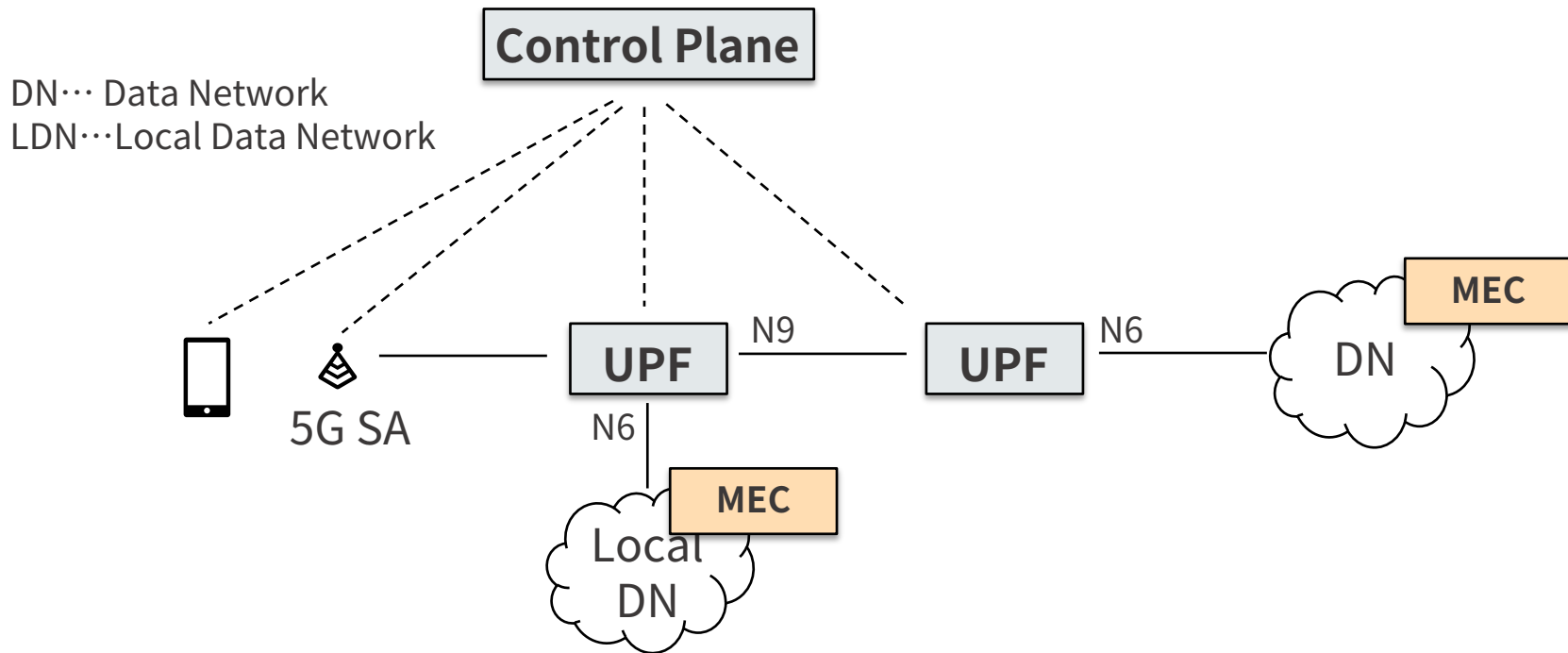
# 3GPP 5GC Architecture

Control PlaneとUser planeの分離が最大の特徴！

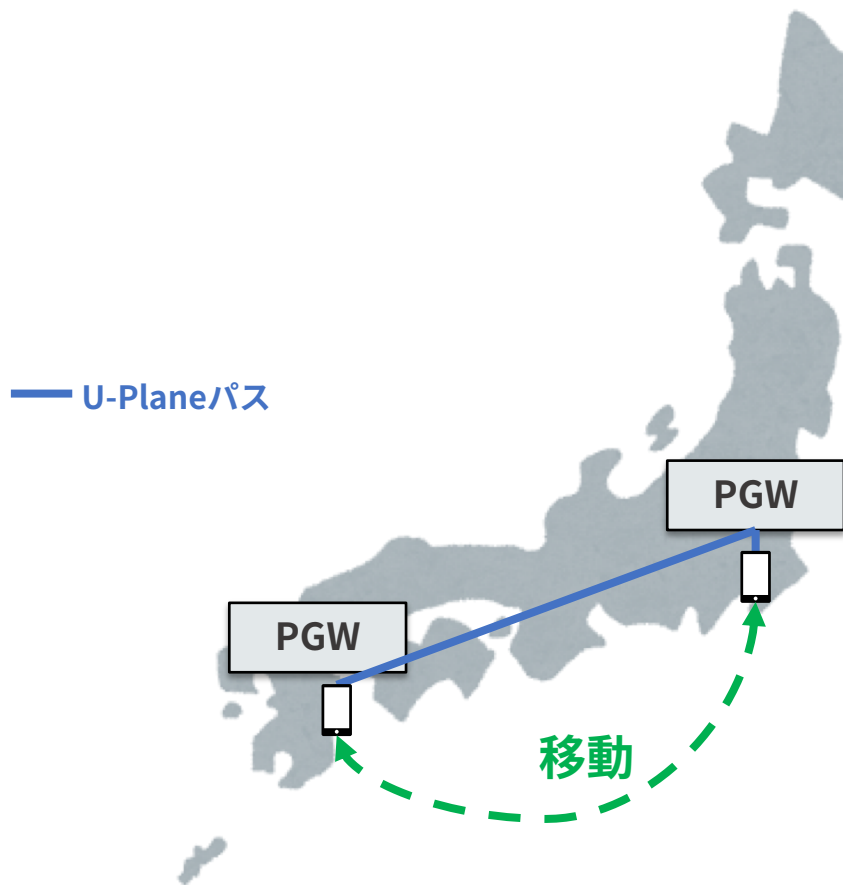


# 3GPP MECを配置する場所

## UPFの配置場所と合わせて設計



# 3GPP 端末が移動した時の課題



**端末移動時もPGWとのセッション維持**

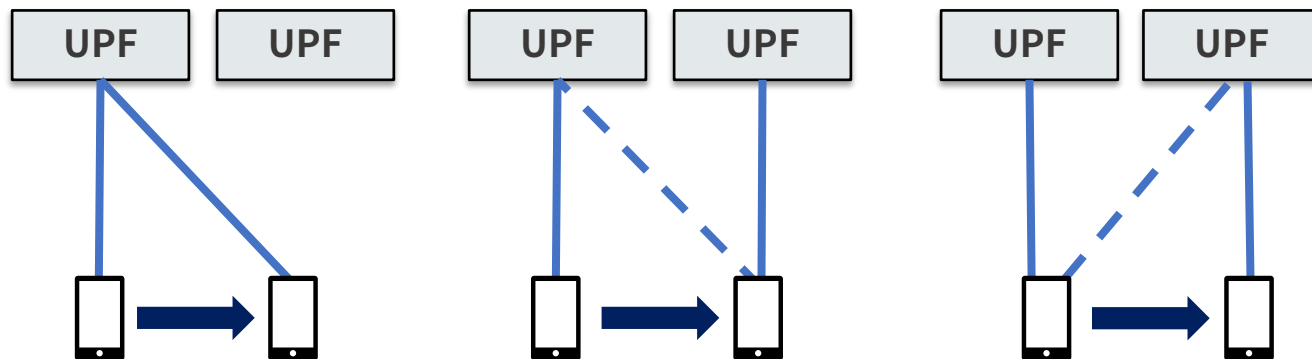


仮に東京在圏時に福岡のMECへ接続  
要求しても東京経由で接続される

# 3GPP Session and Service Continuity (SSC)

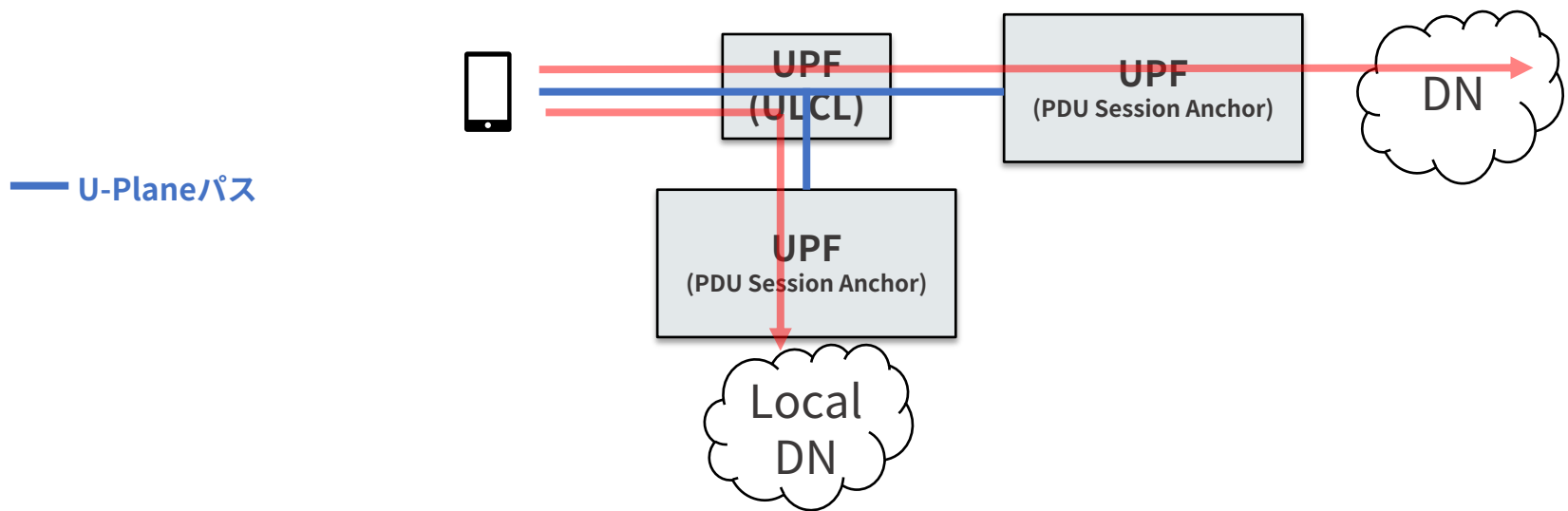
	Mode 1 → 3G/4G同様	Mode 2 → 現実解?	Mode 3 → 端末依存…
切り替え動作	切り替えなし	切断してから接続	接続してから切断
端末側の対応	不要	不要	必要
ユーザ体感	断なしだが遅延増	断ありだが遅延減	断なしで遅延減

## U-Planeパス



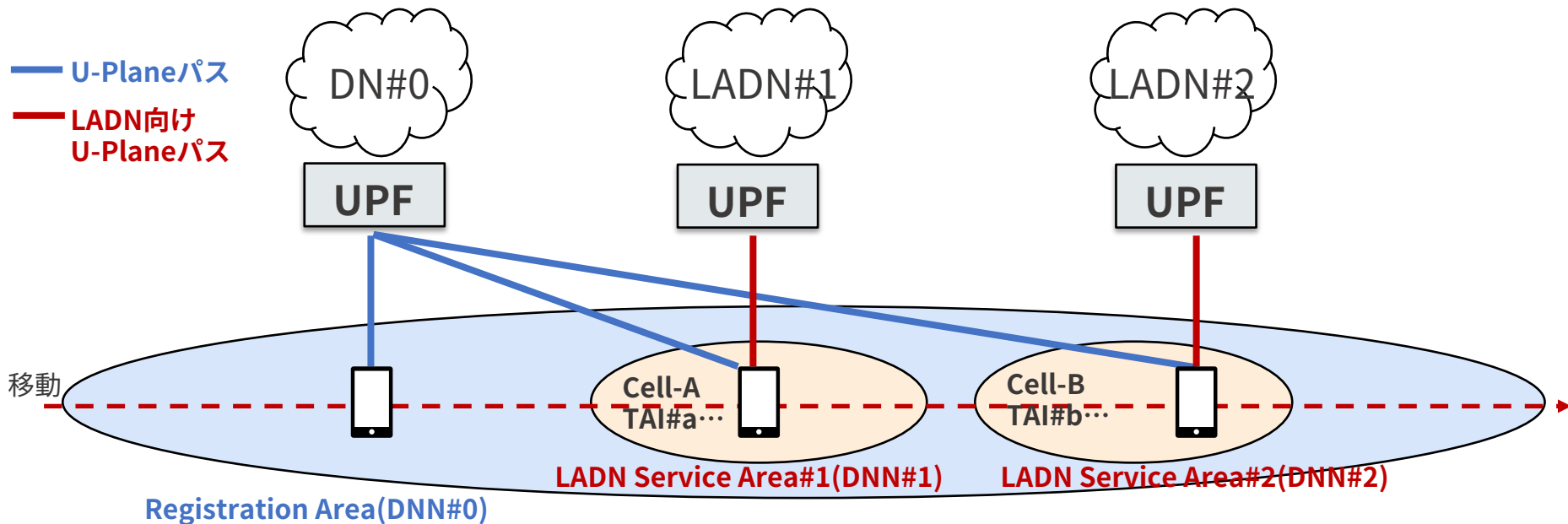
# 3GPP Uplink Classifier (ULCL)

U-Planeパスを5-tuple情報を元に特定の通信のみ分岐する技術



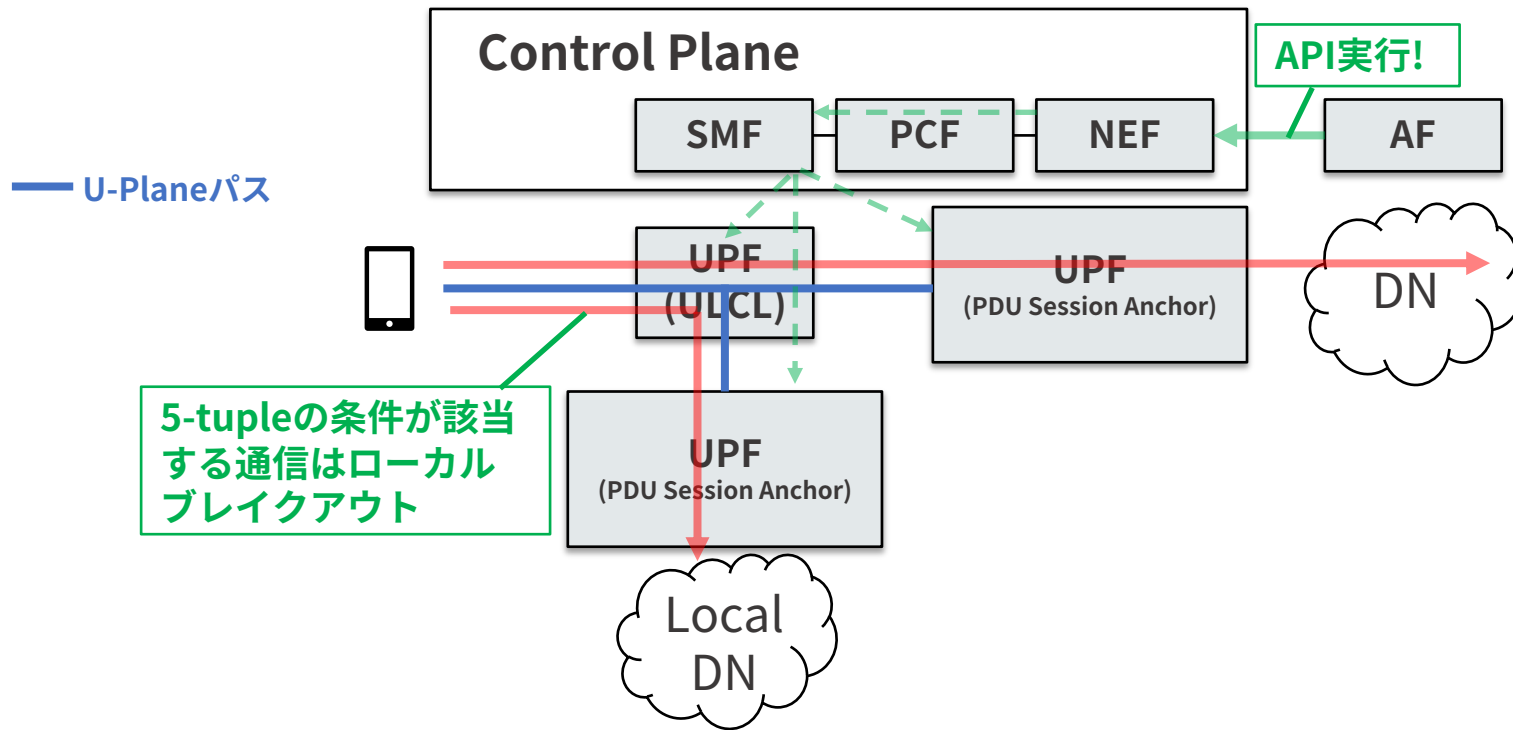
# 3GPP LADN (Local Area Network)

端末の在圏エリアに応じてセッション切り替えを制御



# 3GPP AF Influence on traffic routing

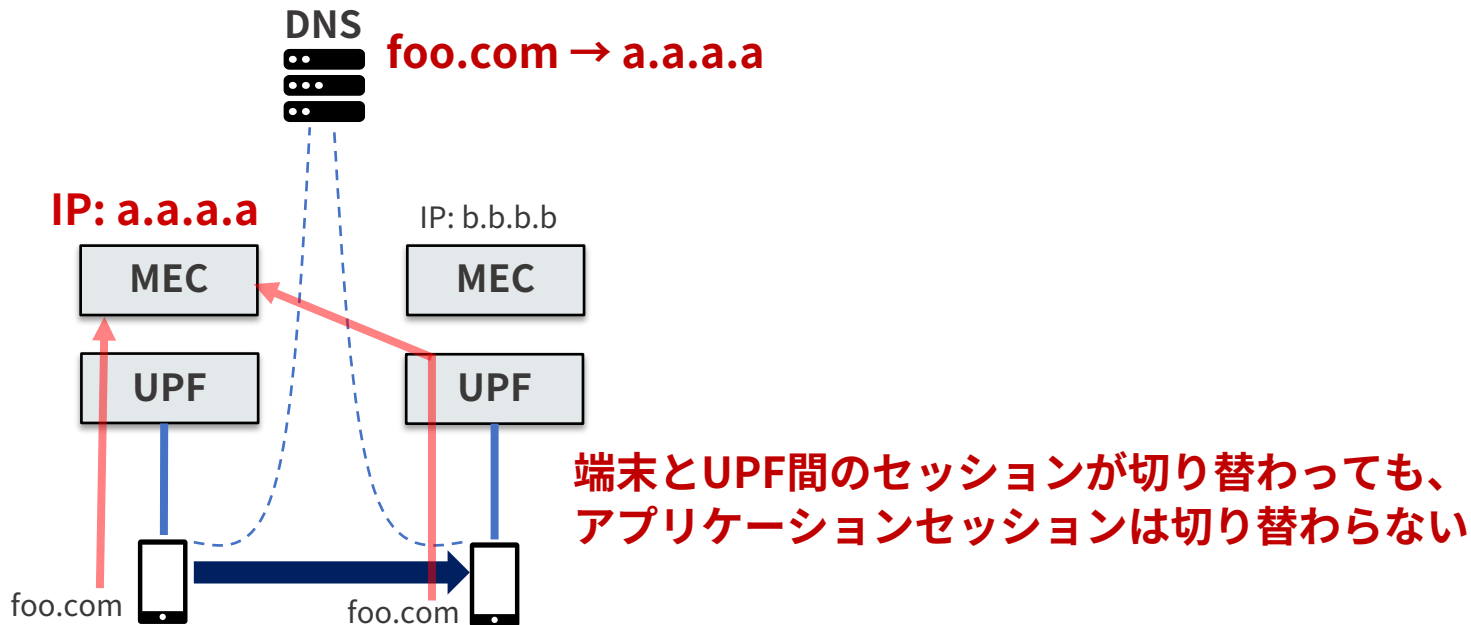
AF (Application Function) が MEC 宛のルーティングを指示し動的制御





# MECが分散配置されたときの課題

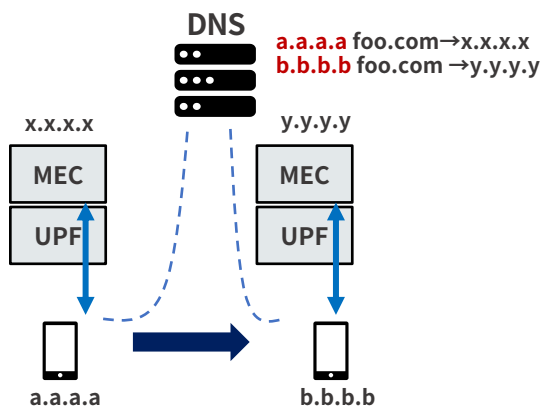
端末が移動した時の近傍のMECのIPアドレスをどのように解決するか？



# ソリューションStudy (一例)

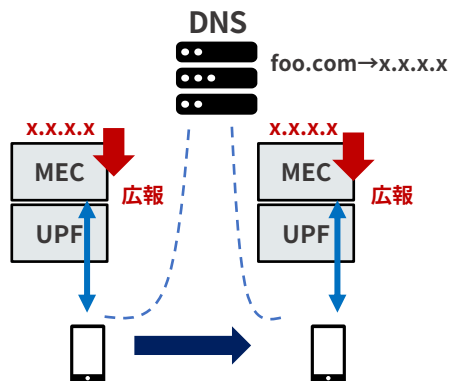
名前解決は頑張れば何とかいけそうだけど、アプリセッションどうしよう...

## DNSベース



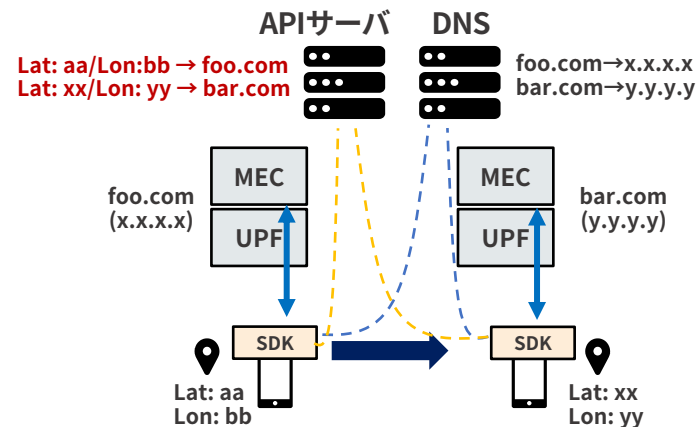
- 😊 マチュアな仕組みで実現
- 😞 IPと地理情報の管理煩雑
- 😞 DNSの性能影響

## IP anycast



- 😊 マチュアな仕組みで実現
- 😞 通信の状況変化で接続先のMECが変わる可能性有

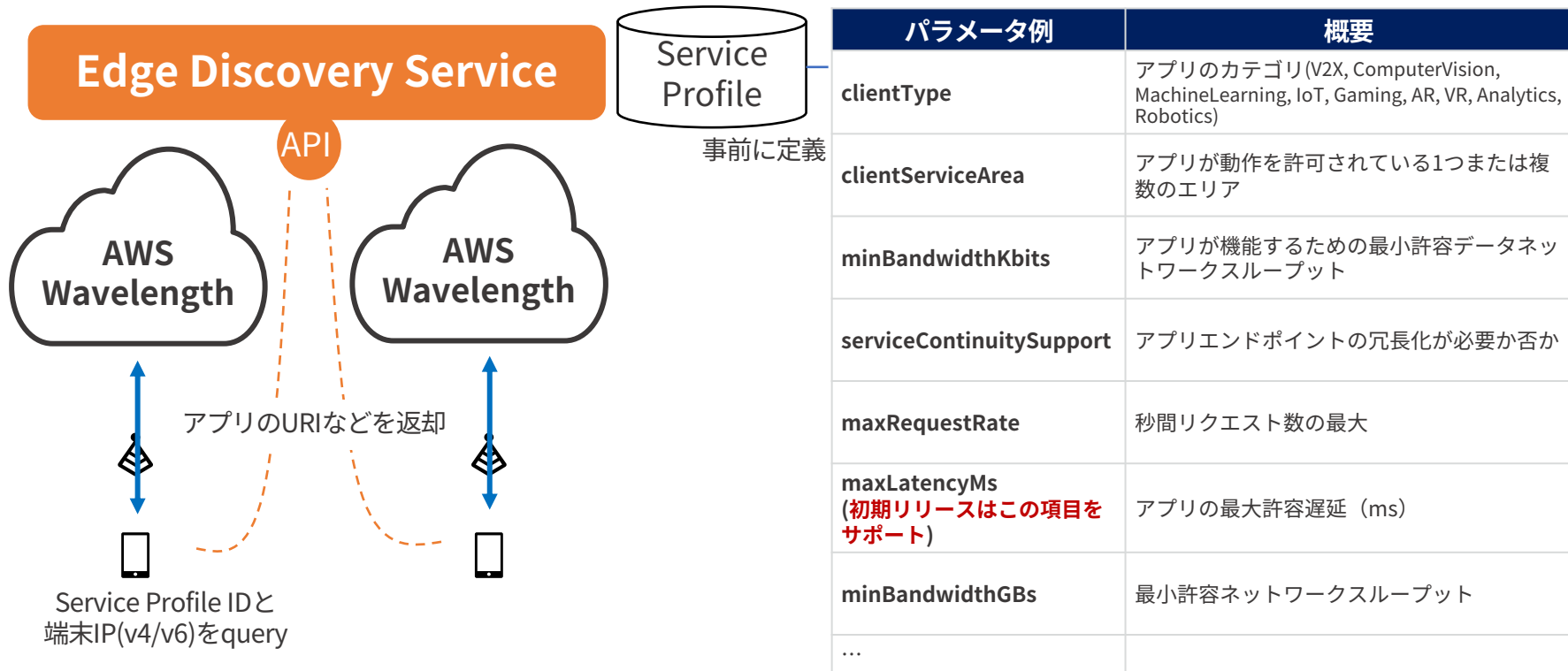
## 緯度・経度ベース



- 😊 アプリケーションレイヤで実現
- 😞 GPS依存
- 😞 電池消費

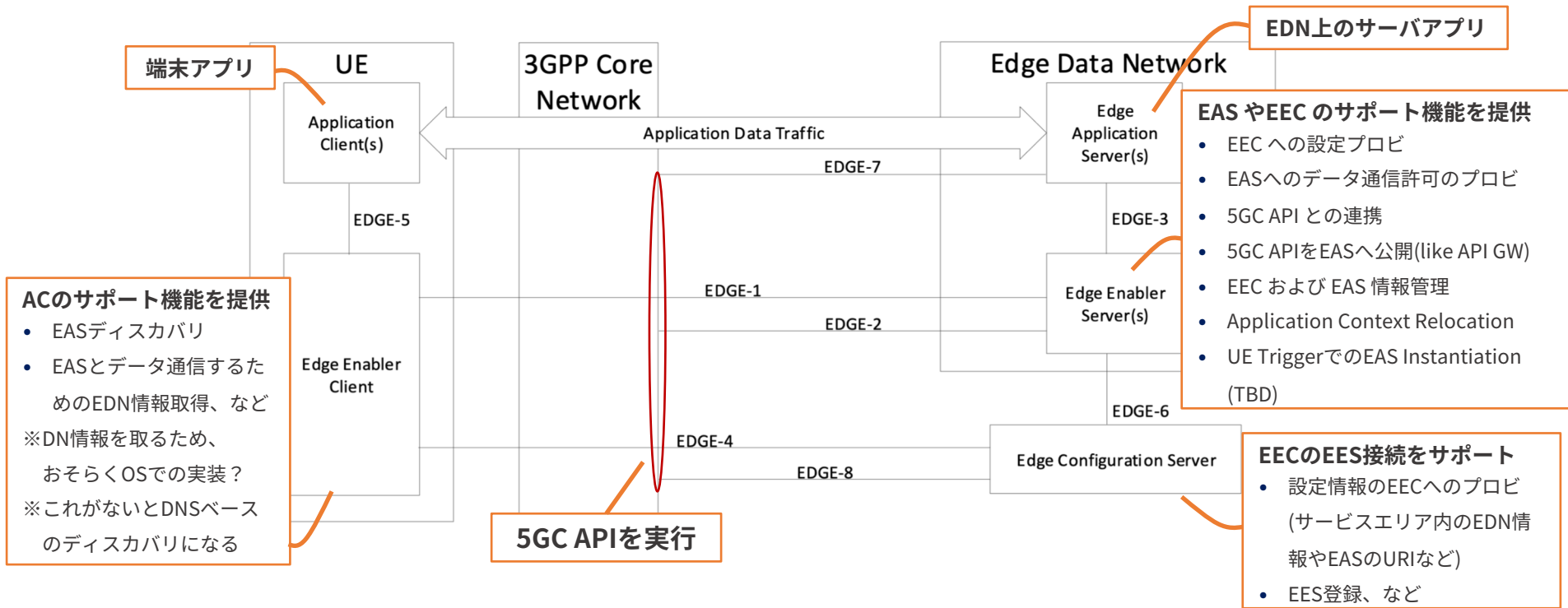
# Verizon 5G Edgeの例

## Edge Discovery ServiceというAPIを提供

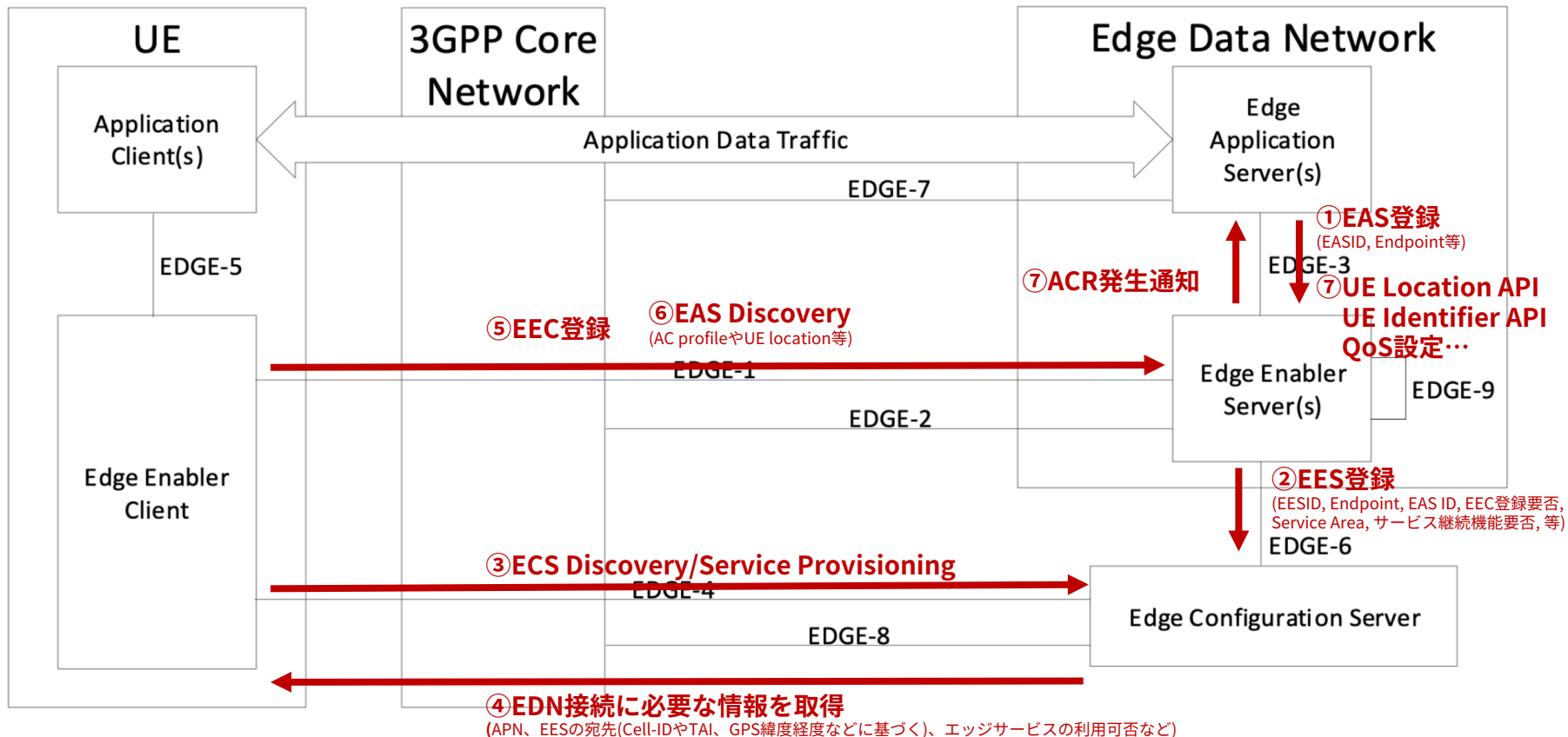


# 3GPP Edge Enabler Layer – EDGEAPP (2021～)

## 5GC APIと連携しEdge Applicationのモビリティを管理

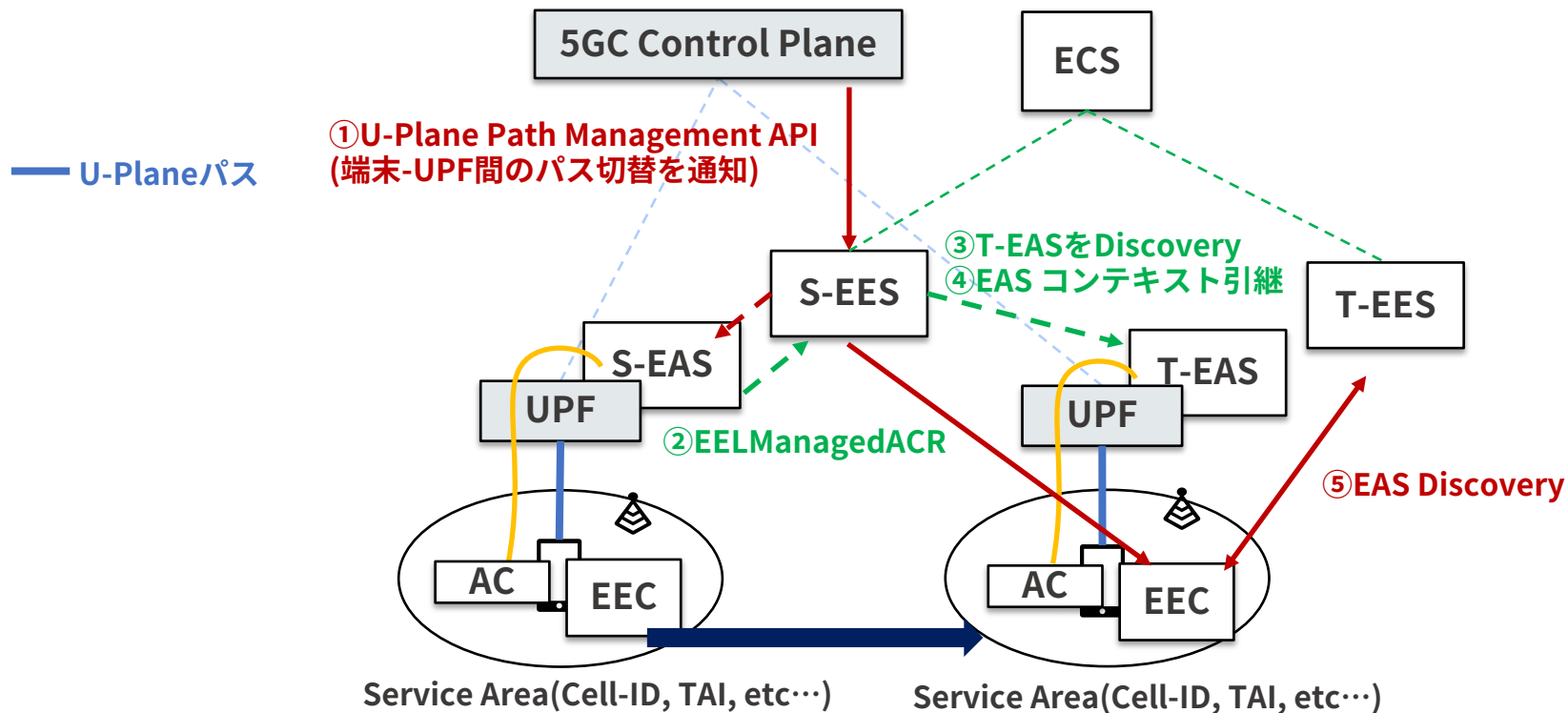


# 3GPP EDGEAPPの各コンポーネントの連携概要



# 3GPP 端末移動時の処理イメージ

## U-Planeパス切替りを検知し、切替先EASの特定・セッション転送



# 本日の内容

- MECの標準化動向の **今** を俯瞰する
- MEC関連の **次** の動向を俯瞰する
- MECの **未来** の姿を想像する

# 3GPP Release 18に向けて

## EDGEAPPの改善が引き続き議論される

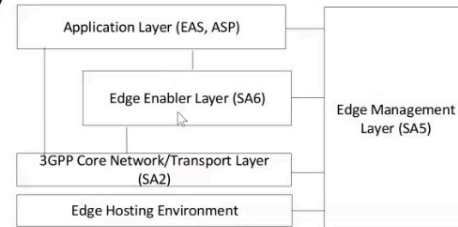
### 3GPP Edge Computing



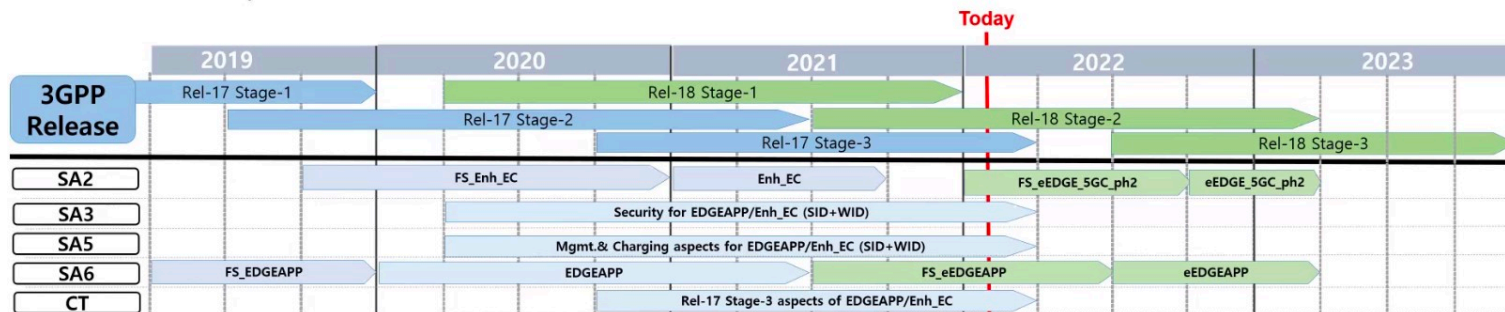
Legend: SA2 SA3 SA5 SA6

Edge Computing has been a major focus area in 3GPP Rel-17

- SA6: Edge Enabler Layer architecture, and deployment scenarios
- SA2: System Architecture enhancement for supporting Edge Computing
- SA3: Security aspects for supporting SA2 and SA6 architectures
- SA5: Management & Charging aspects on Edge Computing



3GPP Rel-17/18 Timeline



参考 : [https://www.3gpp.org/news-events/2248-mec\\_gsm](https://www.3gpp.org/news-events/2248-mec_gsm)

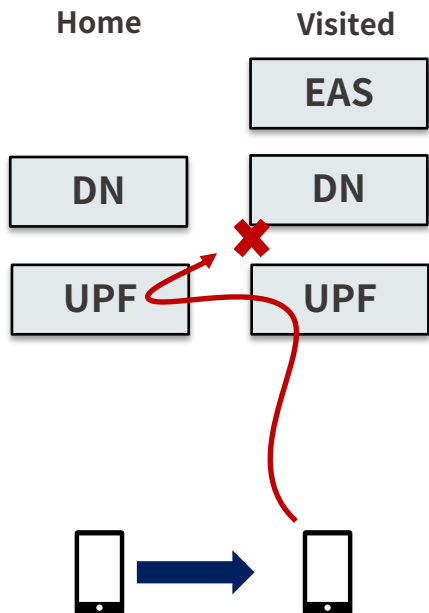
参考 : <https://twitter.com/3GPPLive/status/1488451443682160643>



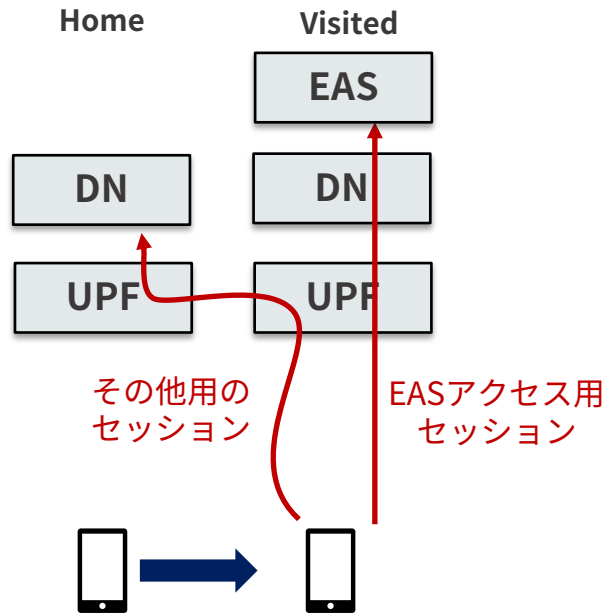
# 3GPP Release 18でのeEDGEAPPの検討内容 (一部)

## 目立つのはローミングシナリオでのEASアクセスの改善の検討

Home-Routed(HR) PDU session経由  
でV-PLMNのEASへアクセスできない



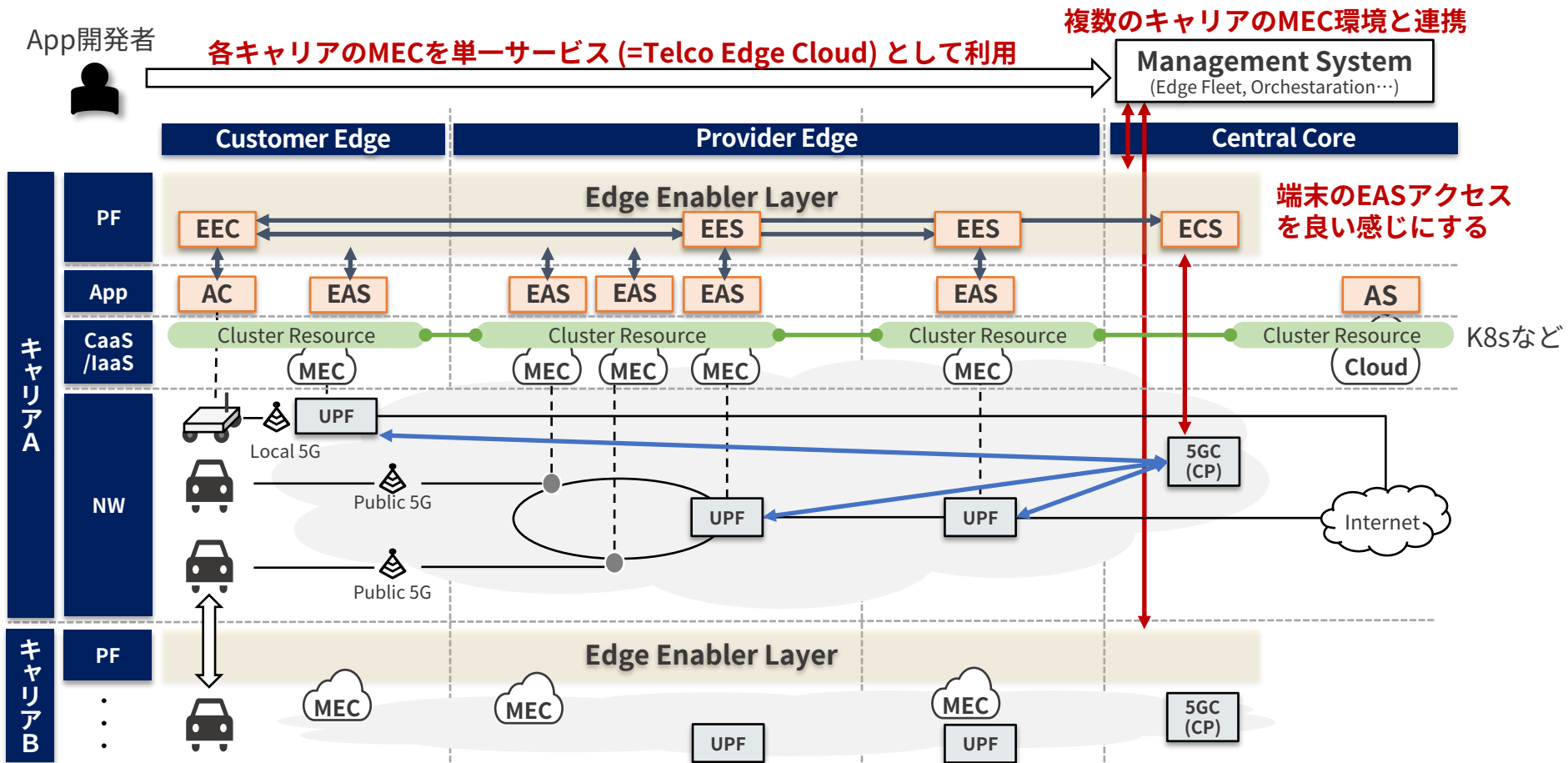
VPLMNでEASアクセス用にLBO、  
その他通信用にHR PDU session



# 本日の内容

- MECの標準化動向の**今**を俯瞰する
- MEC関連の**次**の動向を俯瞰する
- MECの**未来**の姿を想像する

# MECの未来の姿をまとめるとこんな感じ!?



**Thank You!**

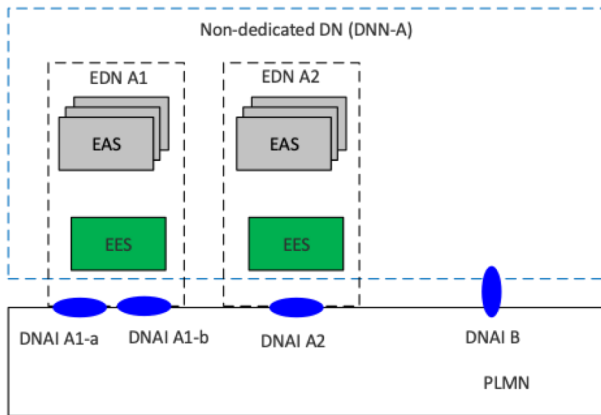
# 【参考】 3GPP EDN (Edge Data Network) とは

## DNN (APN)と1つ以上のDNAIで識別される接続先のデータネットワーク

(\*) DNAIは、アプリケーションから指示されたLDNを指す

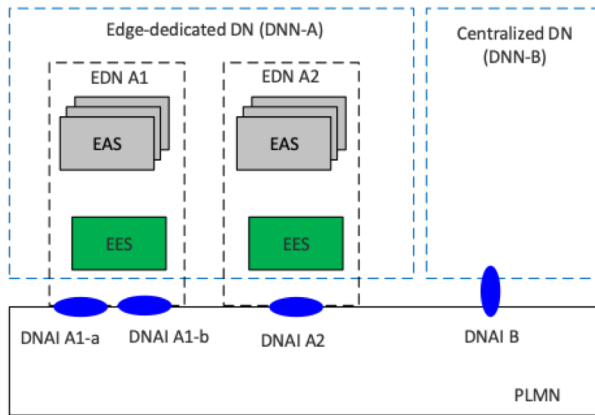
### Non-dedicated DN

他サービスと共通のDNNを使う  
(コンシューマユーザの使用  
するスマホ向けと想定)



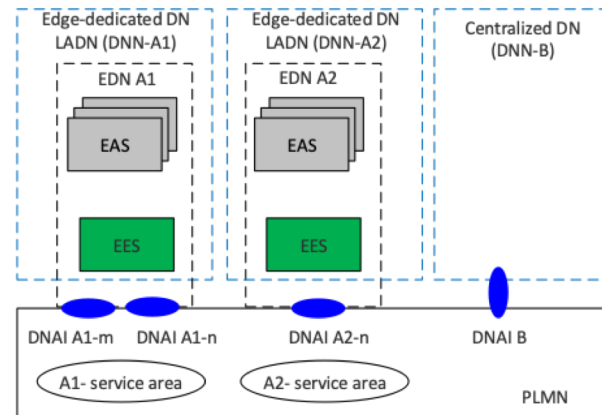
### Edge-dedicated DN

エッジ専用のDNNを提供  
(車など広い範囲で連続的に通信が  
発生するIoTユースケースと想定)



### LADN

EDNとしてLADNを使う  
(工場やスマートシティ等の狭い  
範囲でのIoTユースケースと想定)

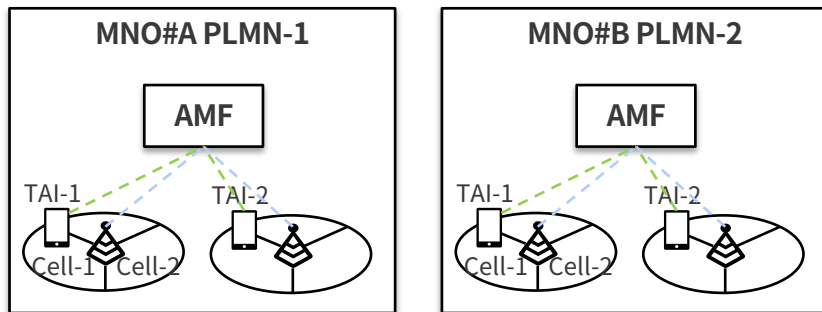


# 【参考】 3GPP Service Area

## 端末のEASアクセスを許可するエリアを指定できる (Optional)

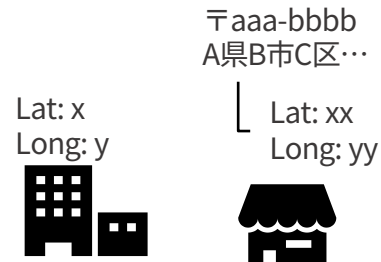
### Topological Service Area

CellID、TAI、PLMN ID など



### Geographical Service Area

有名な建物、住所、〒など

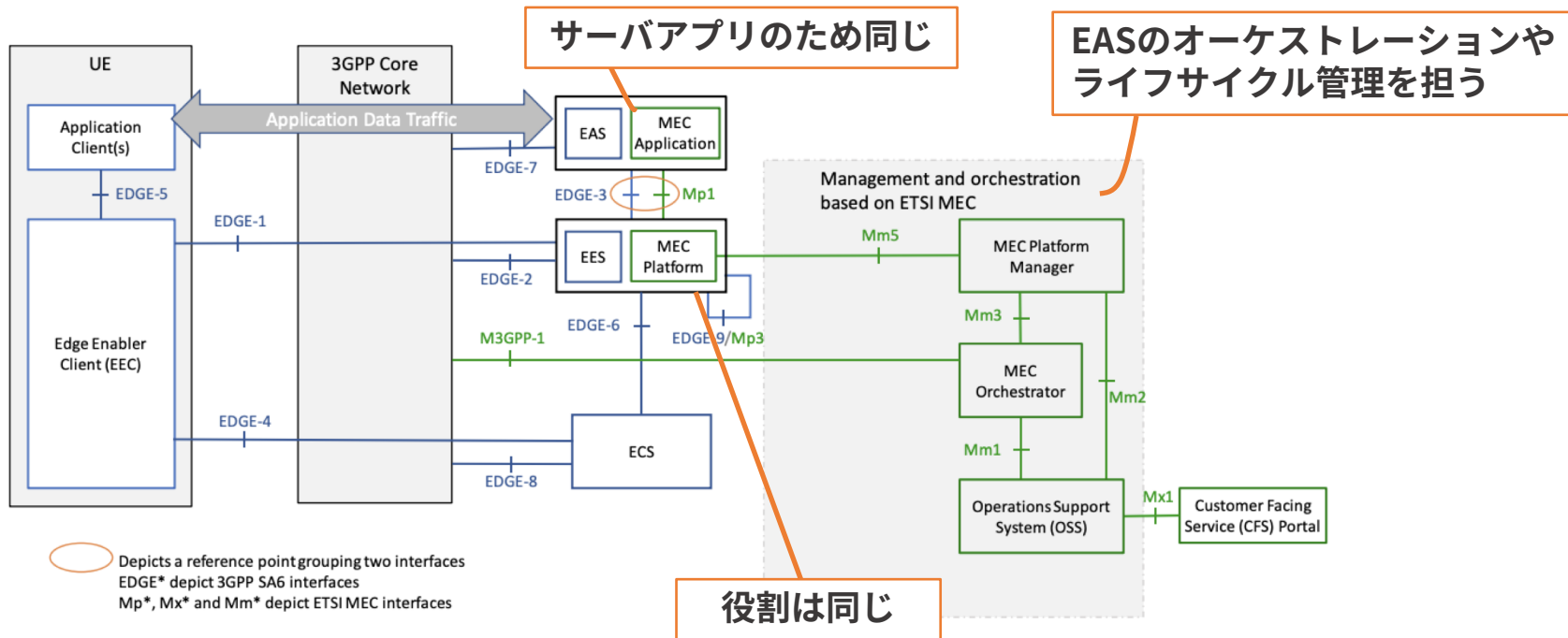


(\*)TS.23032に準拠

※DNごとにアクセス許可を行えたりもする

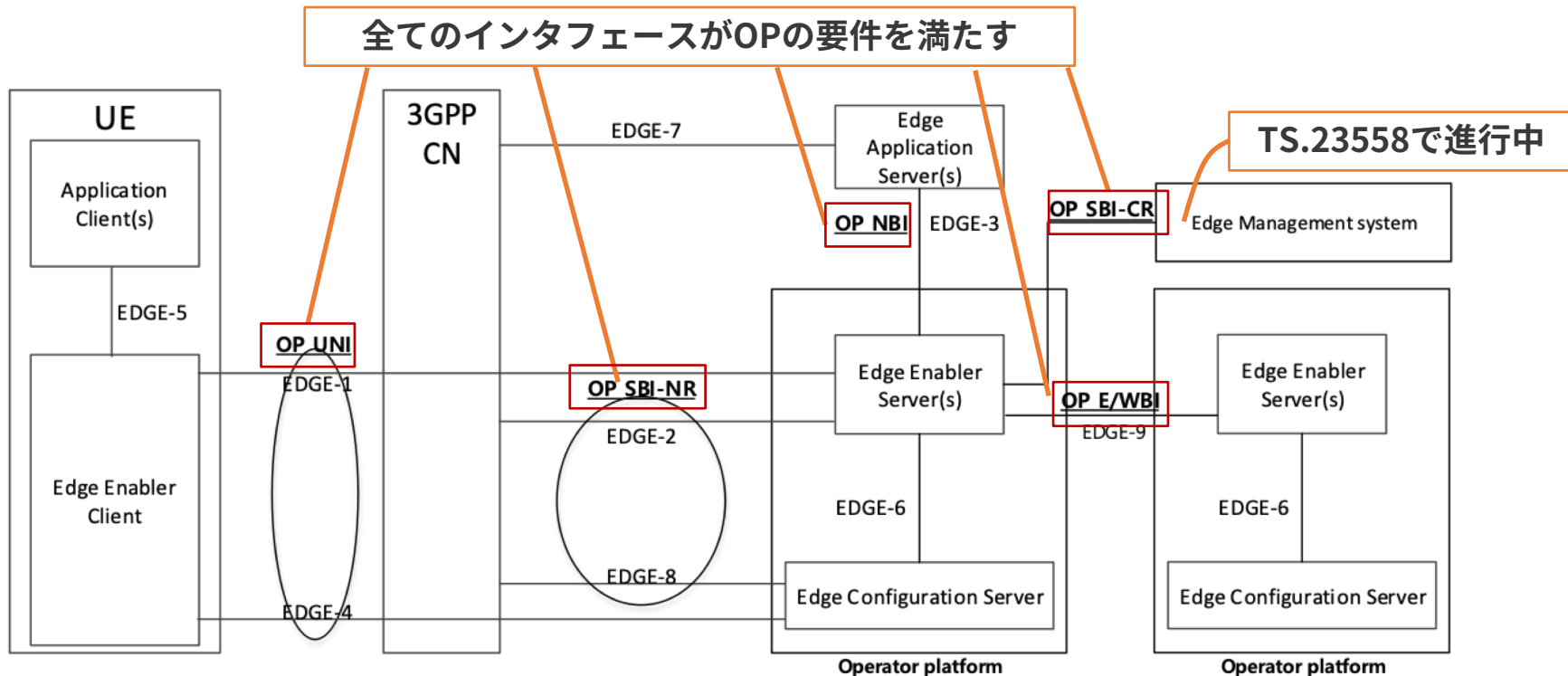
# 【参考】 3GPP EDGEAPPとETSI MECのシナジー

## EDGEAPP + MEC Orchestrationでマネジメントも含めたアーキテクチャ



# 【参考】 3GPP EDGEAPPとGSMA OPとの関係性

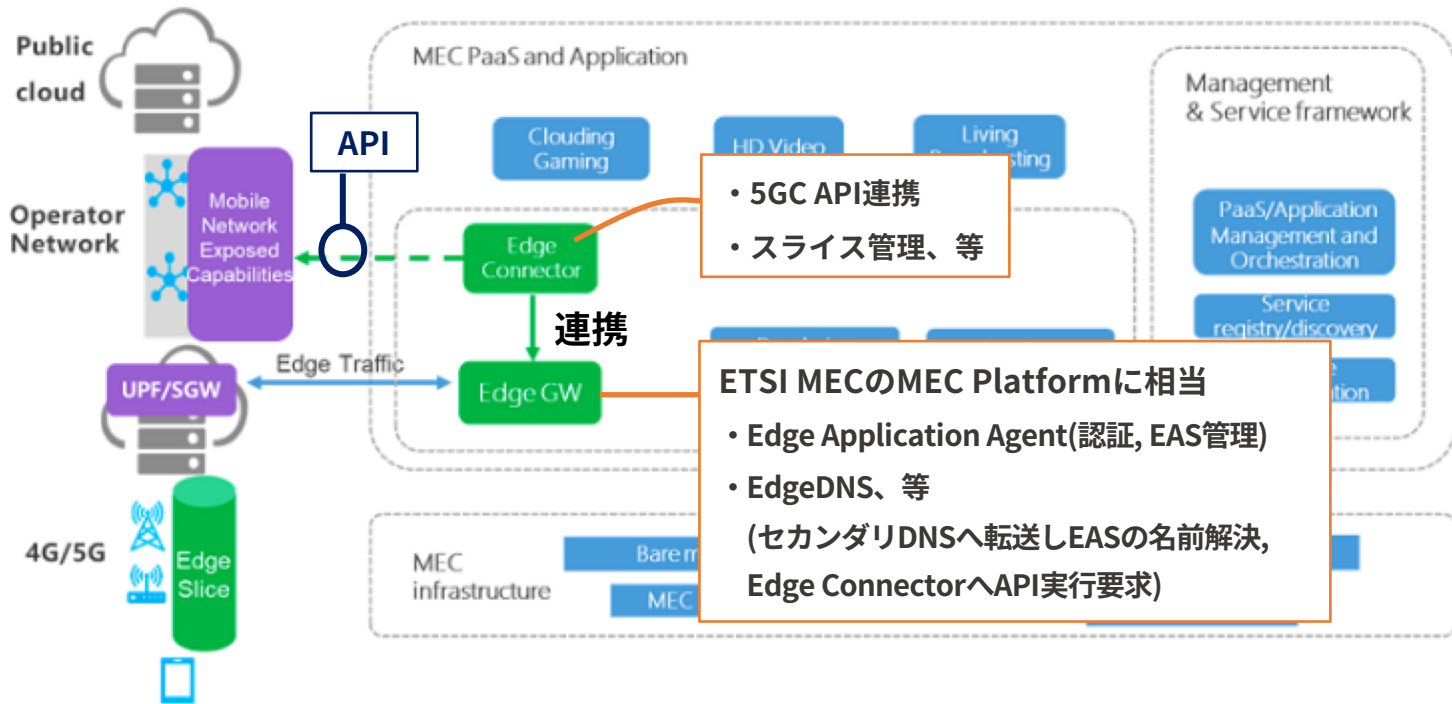
## 3GPP EDGEAPPはGSMA OPの要件をカバーする





# 【参考】 OSS Akraino Blueprints — 5G MEC System

## Intel Smart Edge OpenベースのMEC Platform



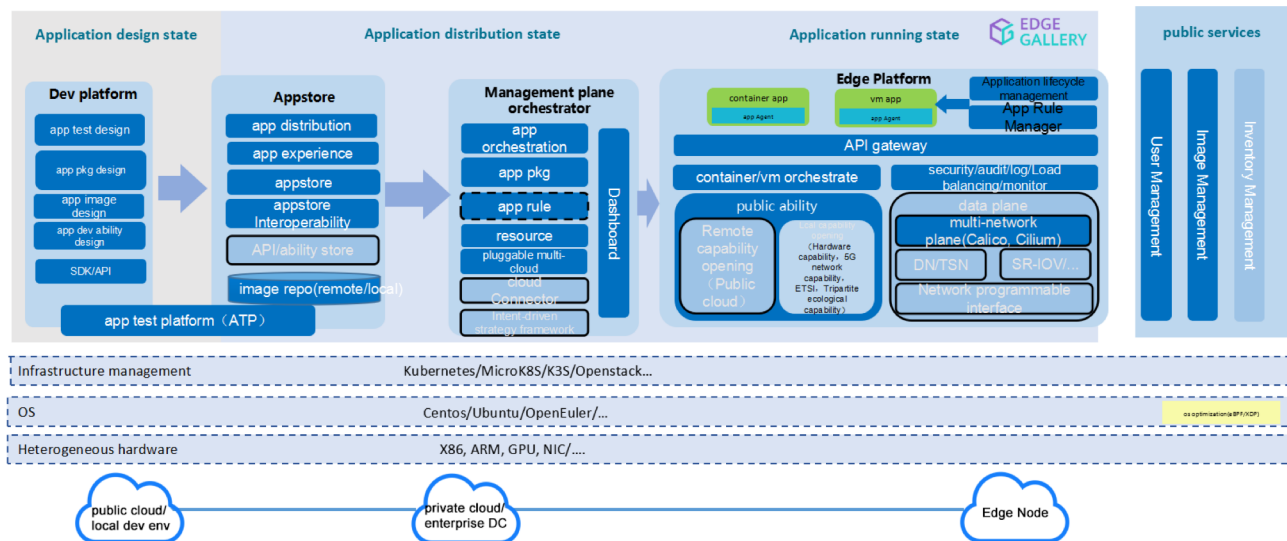
# 【参考】 OSS EDGE GALLERY

## 中国企業のみが参加するETSI MEC準拠のオープンソースMEC Platform



Release v1.1

legend: existing ability planning



CAICT 中国信息通信研究院  
China Academy of Information and Communications Technology

中国移动  
China Mobile

Tencent 腾讯

紫金山实验室  
Purple Mountain Laboratories

China  
unicom 中国联通  
联通 · 改变世界

HUAWEI

九州云  
Cloud

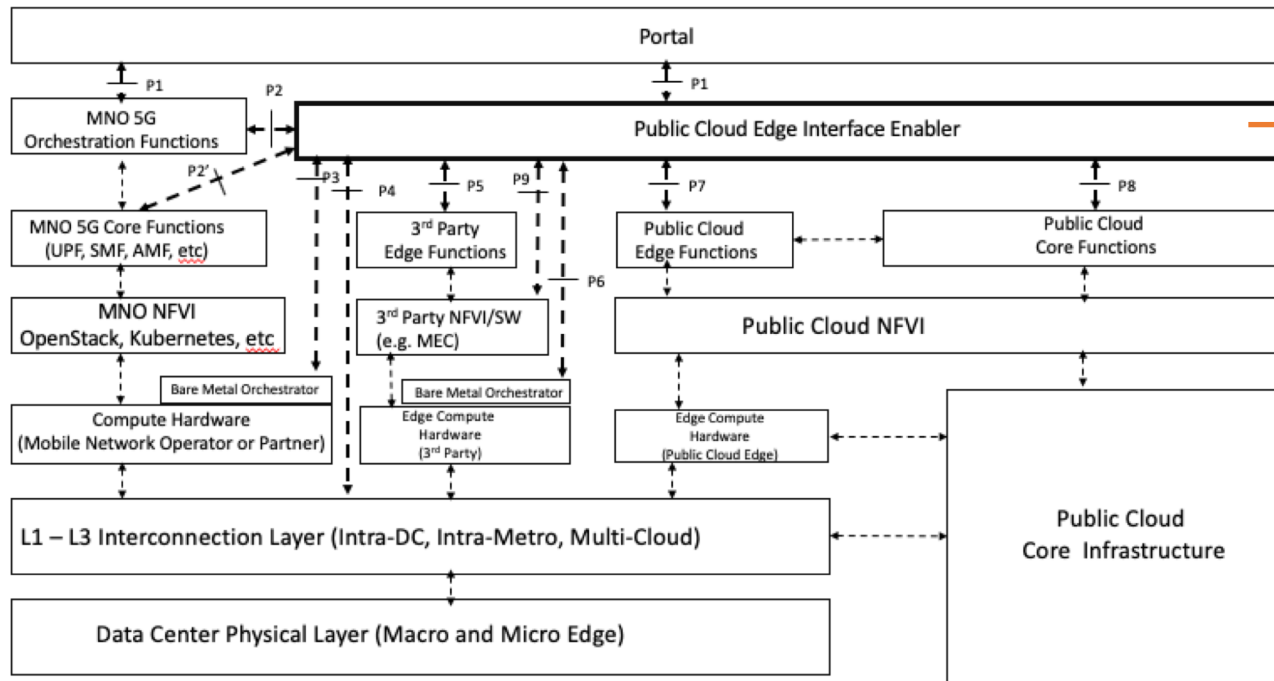
安恒信息  
DAS-SECURITY 安恒信息

参考：<http://docs.edgegallery.org/en/latest/Architecture/Architecture.html>

参考：<https://www.edgegallery.org/en/>

# 【参考】 OSS LF Edge Akraino Blueprints — PCEI

## MNOの5GC APIやMEC、パブリッククラウド等E2Eで抽象化・共通化



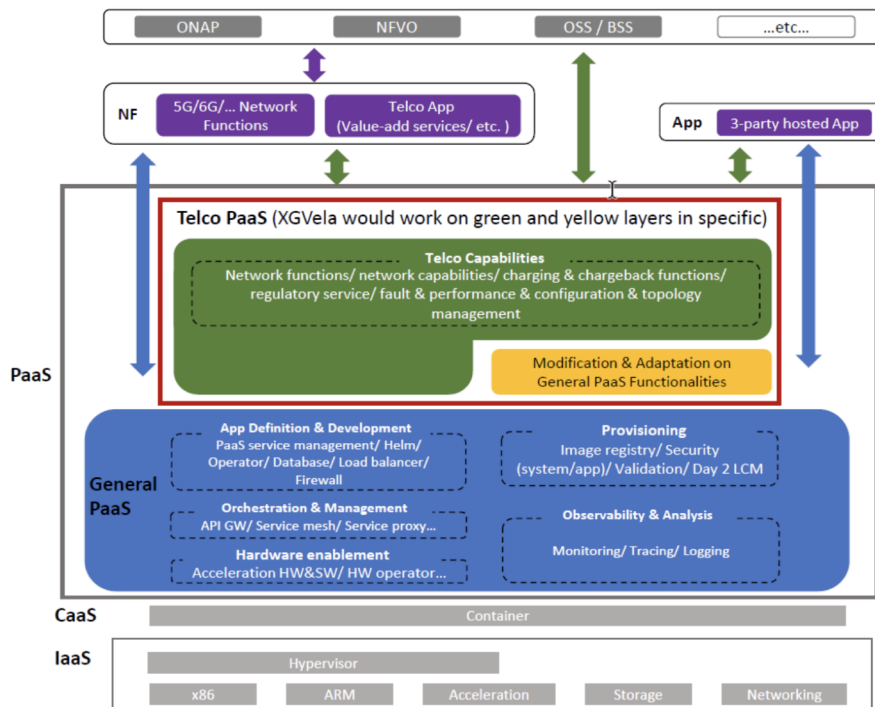
### PCEIのコア機能

#### 実態は

- API GW/管理
- 認証
- ポリシー管理
- 構成管理
- モニタリング
- ディスカバリ
- SDK提供

# 【参考】 OSS XGVela

## Telco PaaS (NaaS) のオープンソース実装



- NaaSのオープンソース実装
- 一般的なPaaSとしてOpenShift Container Platformを採用



参考：<https://xgvela.org/>

参考：<https://github.com/XGVela/XGVela>