

# Edge Deployment Planning API

A CAMARA proposal for pre-deployment rollout planning

Dimitris Giannopoulos

University of Patras



# 1. The Gap

CAMARA already covers key edge-cloud steps:

- **OED**: runtime edge target selection
- **EAM**: deployment and lifecycle management
- **AED**: runtime endpoint discovery

**Missing step**: pre-deployment planning.

**Open question:**

Before deployment starts, which edge options are feasible in a target area?

## 2. Example Use Case - Multi-region cloud gaming rollout

A provider wants to assess where a **public** latency-sensitive gaming service can be launched.

**The Problem:** without a common planning API, the developer must discover and compare edge options operator by operator.

### Expected API output

- feasible edge zones in the target area
- which operators / providers can serve them
- estimated latency envelope

## 2. Example Use Case - Multi-region cloud gaming rollout

A provider wants to evaluate launch feasibility for a **public** gaming app across **multiple countries and mobile operators**.

**The Problem:** Without this API, the developer would need to separately integrate and query every single MNO in every target area to find out what edge options exist to satisfy their latency constraints.

### Expected API output

Through a single query (e.g. via an aggregator), the developer receives:

- which candidate edge zones are feasible in a target area
- through which operators / cloud providers
- with what estimated latency envelope

### 3. Why This Is Not Covered By Optimal Edge Discovery (OED)

#### OED

- runtime / near-runtime decision
- active device or session context
- returns the **best** target edge

#### Edge Deployment Planning API

- pre-deployment decision
- no active device required
- returns the **feasible set**

**In one line:** OED is optimization; Edge Deployment Planning is viability.

## 4. What the API Should Do

Given:

- a **target area**
- an **application profile**
- a set of **constraints**

Return:

- feasible **edge zones**
- relevant **edge provider/ mobile network operator information**
- estimated **performance metrics**

## 5. Position in the Edge Cloud Family

API	Purpose
OED	best target selection
EAM	deployment / lifecycle
AED	runtime endpoints
Edge Deployment Planning API	pre-deployment feasibility

This fills a new lifecycle step rather than duplicating an existing one.

## 6. Design Principle

### Developer-facing abstraction

The API should expose:

- candidate edge options
- viability results
- estimated planning metrics

The API should **not** expose:

- DNN
- S-NSSAI
- internal routing policies
- operator topology

## 7. Why It Matters

- helps developers decide **whether to launch**
- supports **rollout planning** across areas
- enables **multi-operator aggregation**

**Goal:** allow developers and platforms to make informed go / no-go rollout decisions before committing engineering work.

# Thank you

## Discussion points

- Is pre-deployment feasibility a useful missing capability?
- Is the right path a new sibling API, rather than an OED extension?