Orchestrating Applications in the Cloud-Edge Continuum

Giuseppe Bisicchia

Mauriana Pesaresi Seminar Series
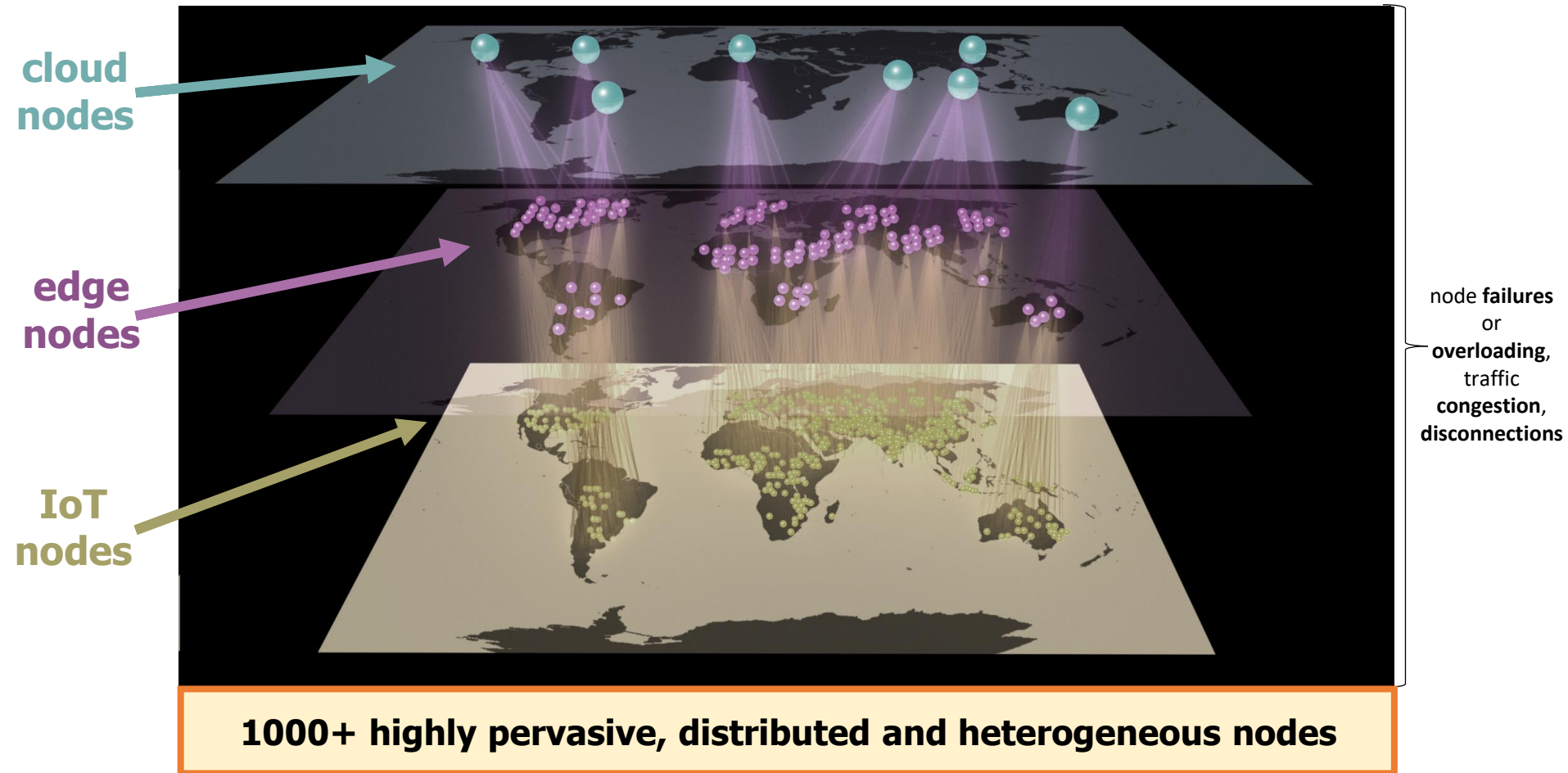
Context: Multi-Service Applications

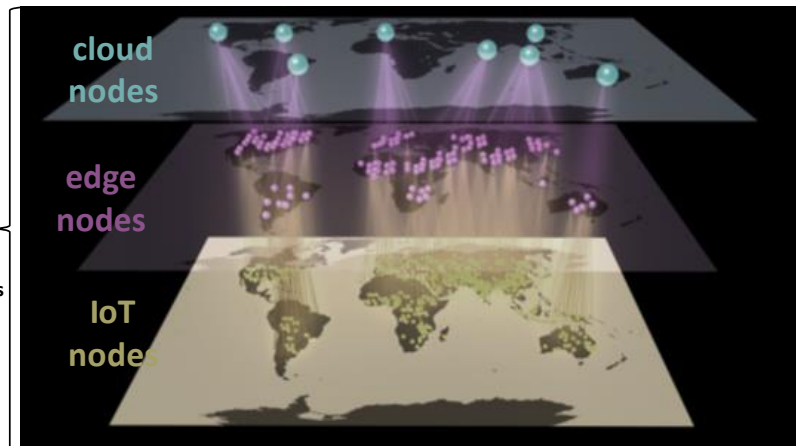strictly hardware and **software requirements, QoS ...**

100+ interacting (micro)services

Context: Cloud-Edge Infrastructures

cloud nodes

edge nodes

IoT nodes

node **failures** or **overloading**, traffic **congestion**, **disconnections**

1000+ highly pervasive, distributed and heterogeneous nodes

Research Context

Cloud-IoT Infrastructures

node **failures** or **overloading**, traffic **congestion**, **disconnections**

1000+ highly pervasive, distributed and heterogeneous nodes

100+ interacting (micro)services

strictly hardware and **software requirements, QoS ...**

Multi-Service Applications

UNIVERSITÀ DI PISA

Research Context

Cloud-IoT Infrastructures

100+ interacting (micro)services

cloud nodes

edge nodes

IoT node

node **failures** or **overloading**, traffic **congestion**, **disconnections**

strictly hardware and **software requirements**, **QoS** ...

None of the **existing tools** supports a **continuous**, **QoS-** and **context-aware** management of **microservices** on **Cloud-Edge infrastructures** in **continuity** with the **CI/CD** pipeline

1000+ highly pervasive, distributed and heterogeneous nodes

Multi-Service Applications

UNIVERSITÀ DI PISA

Open Research Problem

*How can we **ensure** the **continuous satisfaction** of application **requirements** in a **scalable** manner over the **Cloud-Edge continuum**?*

Targets

- Reducing **decision-making times** and **unnecessary management operations**

- **Improve** the orchestration **quality** of next-gen **applications** by:

  - **continuously ensuring** their **requirements** and

  - **balancing** - possibly **contrasting** - **QoS objectives** (e.g., energy/carbon footprint vs operational costs).

Key Factors

Infrastructural changes

Changes in services requirements

Addition/removal of services

Technical Issues

Required **new orchestration processes**, more **lightweight** and better suited for **Cloud-Edge** settings

Should consider **various** application **paradigms** (e.g. serverless, osmotic)

Support **data migrations** and **scaling** of services

Possible Research Lines

Learning functionalities

Orchestration methodologies

Cost models & heuristics
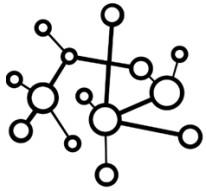
Formal Models

How can we **ensure** the **continuous satisfaction** of **application requirements** in a **scalable** manner over the **Cloud-Edge continuum**?
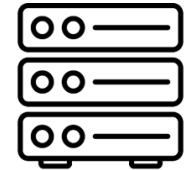
?

Application Requirements Data

Infrastructure Monitored Data

Software Reqs — Software Caps

Hardware Reqs — Hardware Caps

IoT Reqs — IoT Caps

Service-to-Service Bandwidth/Latency — Featured Node-to-Node Bandwidth/Latency

Reducing **decision-making times** and **unnecessary management operations** while **improving** the **orchestration quality** of next-gen applications

Thank You