Towards characterization of edge-cloud continuum

http://d3s.mff.cuni.cz

Department of Distributed and Dependable Systems

CHARLES UNIVERSITY IN PRAGUE

faculty of mathematics and physics

Danylo Khalyeyev

Tomaš Bureš

Petr Hnětynka

2000s: Cloud computing

- Benefits:
  - High resource availability and scalability
  - Decoupling application development from infrastructure maintenance
  - Universal accessibility
  - etc.
- Drawbacks:
  - High latency
  - Harder to ensure Quality of Service
  - Hard to support cyber-physical applications
  - etc.

2010s: Edge computing

- Moving computation closer to the end users
- Providing higher security and robustness
- Using smaller servers and data centers
  - Just one network hop away from the end users
- Can be used even without cloud

2010s: Fog computing

- Moving storage and communication closer to the end users
  - I.e. not only the services
- Improving application efficiency
  - By reducing network consumption
  - By masking network outages
  - By reducing latency and response time

2010s: Dew computing

- Using resources of the end devices themselves
  - Within the cloud computing applications
  - When there is a need or when it makes sense
  - e.g. caching cloud data locally
- Cannot really be used without cloud or edge

2010s: Mist computing

- Lightweight computing between networking nodes, sensors and microcontrollers
- „Everything computing everywhere“
- Theoretically can be used as standalone
  - In practice, is almost always complementary

Putting it all together

- Edge computing: smaller servers and data centers, close to the end users
- Fog computing: communication and storage at the edge
- Dew computing: utilizing the capabilities of end devices
- Mist computing: using extremely resource-constrained devices

- In practice, rarely any of these approaches is used on its own
- And the boundaries are not clearly defined

2020s: Edge-cloud continuum

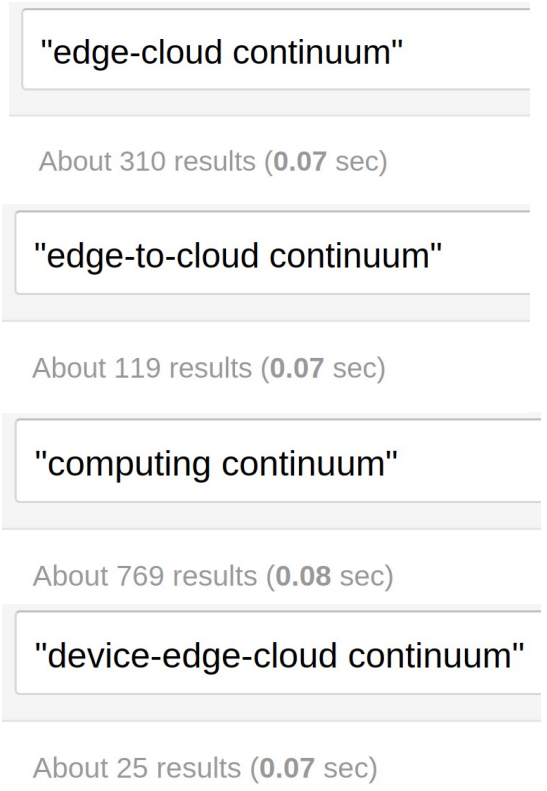
- Lots of variations:
  - Cloud continuum
  - Computational continuum
  - Device-Edge-Cloud continuum
  - Device-to-Cloud continuum
  - etc.
- Usage is increasingly common, e.g., recent Horizon Europe callls:
  - „Cognitive Cloud: AI-enabled computing continuum from Cloud to Edge“ (2021)
  - „A clear technological strategy for edge integration into a cloud continuum offering opportunities for European cloud/edge technology suppliers“ (2022)

Edge-cloud continuum

- What is it?
- So far is rather a vision than a coherent paradigm
  - And even as such, the vision is not clearly outlined
- Our goal: to clarify the ECC vision
  - What are the elements of ECC ecosystem?
  - What properties are expected from ECC?
  - Is this vision feasible?
  - What does it bring to the table?

Methods

- The term can be traced back to 2017
- Hundreds of papers mention „edge-cloud continuum“ or very close terms
- Much fewer engage with the term more extensively
  - i.e., beyond mere mention or reference
- We look for commonalities between different views on ECC



Defining characteristics of ECC

1. Hierarchical structure

- ECC consists of a wide array of devices
  - With drastically different computational capabilities
  - From giant data centers to smallest end devices
  - Being interconnected and interdependent, these devices form a hierarchical structure

2. Cross-level situation-aware cooperation between components

- Components cooperate dynamically
  - Based on current conditions and objectives
  - Forming ad-hoc task-oriented groups
  - With a significant degree of autonomy

Defining characteristics of ECC

3. Fully liquid software

- Software components and functions can „flow“ from one device to another
  - Between devices of the same level – handover
  - Between devices on different levels
  - Based on current environment conditions, application needs, etc.
  - Not easy to achieve in practice

4. Edge intelligence

- Deployment of ML models in ECC
  - Training and inferencing distributed across ECC
  - Can potentially help to achieve the „smart environment“ properties of ECC
    - e.g., Dynamic self-adaptation and situational cooperation

Open questions

- Will hardware capabilities be sufficient in the near future?
  - Can we assume continuous IoT hardware improvement?
- To what degree it is possible to achive ECC properties with current software approaches?
  - E.g., dynamic self-adaptive management, software liquidity
- Are there sufficient incentives to adopt the ECC paradigm in industry and how to create them?
  - Will the industry be willing to move towards ECC?
- What are the suitable representative examples?
  - What are the new use cases that are enabled by ECC?