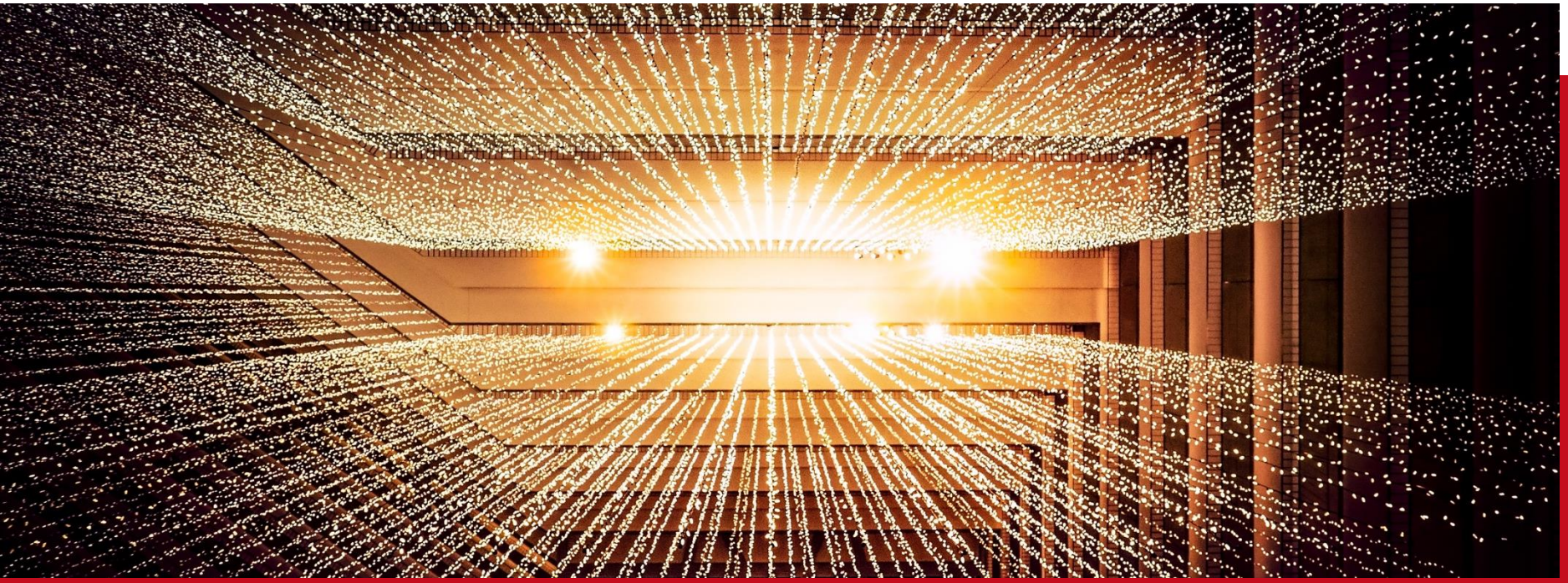
WOSCx4

TECHNISCHE UNIVERSITÄT BERLIN



Umbilical Choir:

Automated Live Testing for Edge-To-Cloud FaaS Applications

M. Malekabbasi, T. Pfandzelter, D. Bermbach | Scalable Software Systems

EINSTEIN CENTER Digital Future

3S

The Challenge: Risky Releases in Edge-to-Cloud FaaS

- Users demand high QoS[1]; frequent releases (thanks to CI/CD[2]) increase risk.
  - Amazon: Deploying every 11.6 seconds[3].
  - Siemens: Over 4 production deployments per month by GitLab and AWS

- **Edge-to-Cloud Gap:** Standard cloud live testing (A/B, canary) doesn't fit.
  - Lacks **geo-awareness** for distributed edge nodes.
  - FaaS platforms offer limited native live testing features.

- **Result:**

  Developers hard-code complex, error-prone solutions, or skip thorough testing.

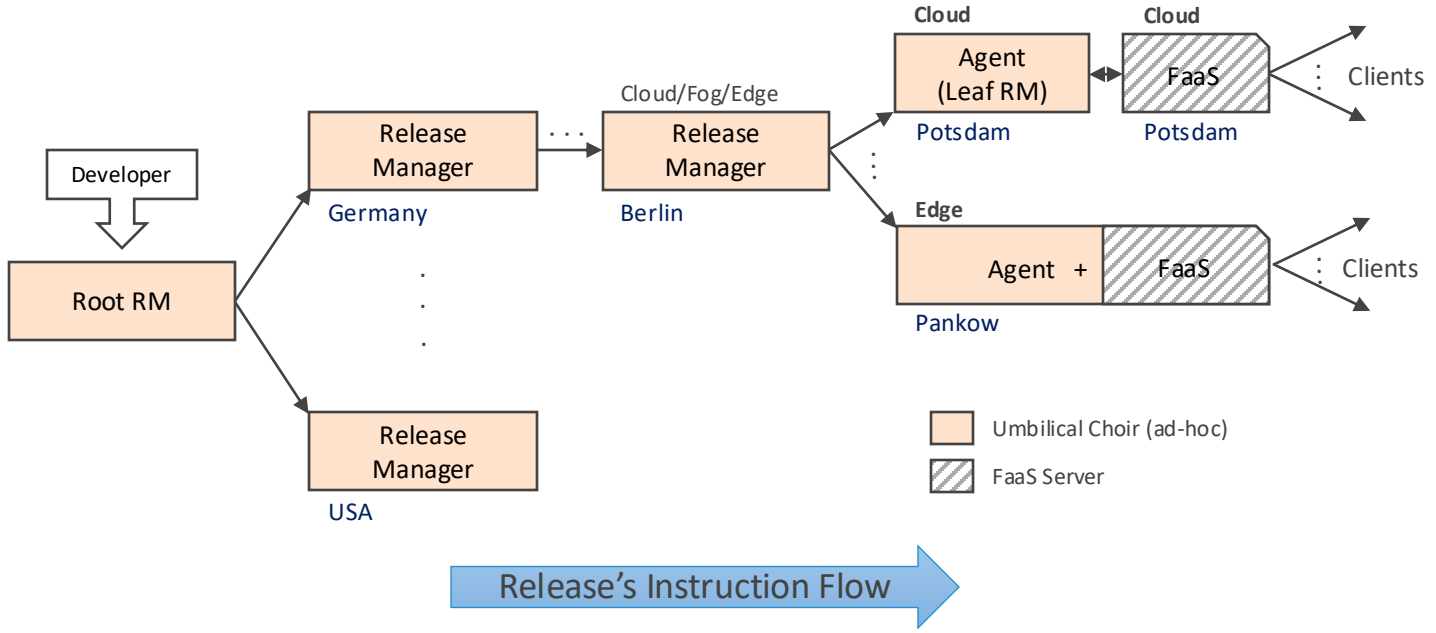
Elfic, a comedy juggler

1. Quality of Service
2. Continuous Integration / Continuous Delivery
3. DevOps Research and Assessment (DORA) State of DevOps Report

Umbilical Choir (UC): Automated Live Testing for the Edge

- A novel framework for **serverless edge-to-cloud live testing**.

- **Core Idea: Tree-like Hierarchy**
  - **Release Managers (RMs):**
    Plan & delegate strategies across regions.
  - **Agents:**
    Execute tests on FaaS platforms
    (via a Proxy Function for traffic control).



- **Key Design:** Scalable, geo-aware by structure,
  FaaS platform-agnostic (deploys a Proxy function),
  works with edge network limitation (no push/ssh access).

Umbilical Choir: Declarative & Flexible Live Testing

- **Declarative Strategies (YAML):** Define multi-stage releases.
  - Supports: A/B, Canary, Dark Launch, Gradual Rollout.
  - Metric-based conditions for progression/rollback.

- **Geo-Aware Rollouts:**
  - **Global Incremental:** (Good for critical patches).

  - **Local Sequential:** (Minimizes risk).

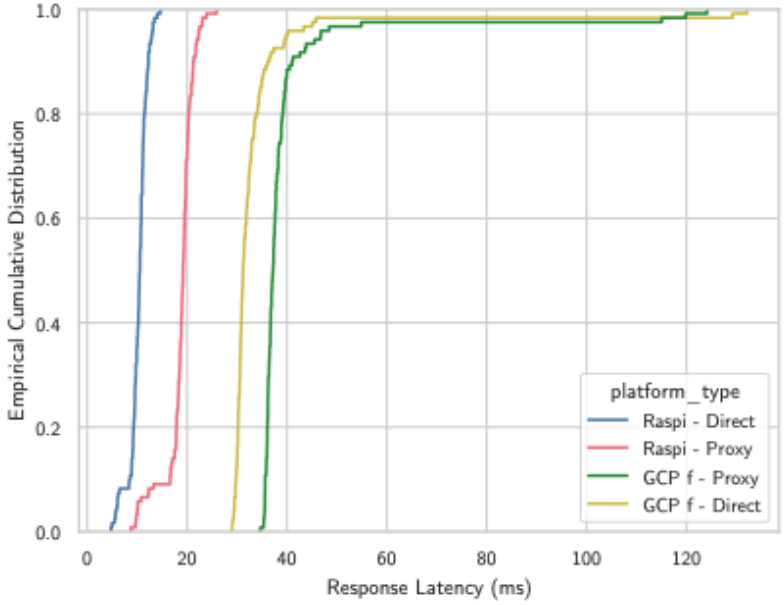
  - **Regional Incremental:** Global Incremental applied to a specific region first.
  - **Regional Sequential:** Local Sequential applied within a specific region.
- These can be combined for complex scenarios.

A simple "Release Strategy"

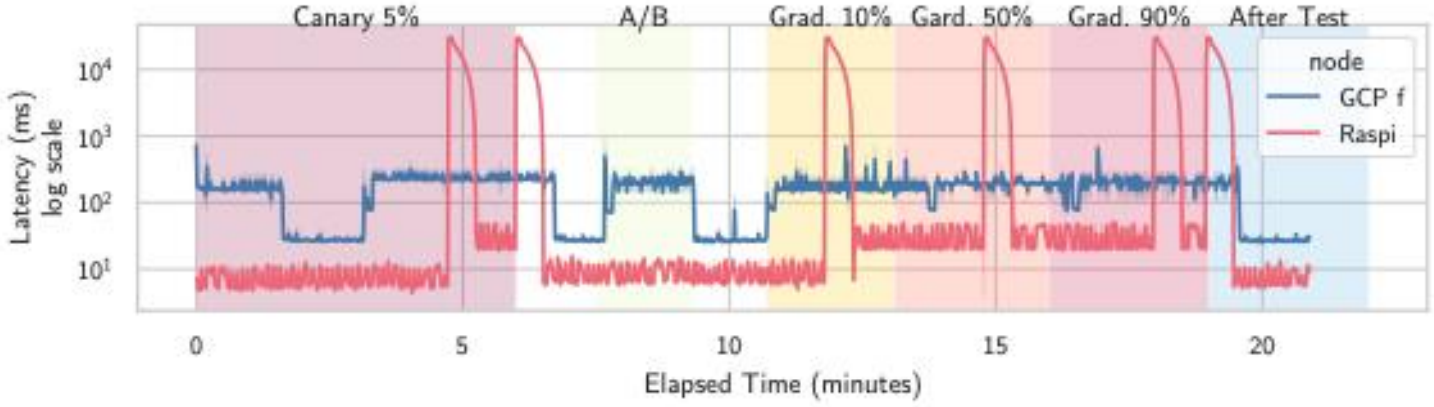
```yaml
id: 12
name: A/BTestF1Function
type: patch/major/minor
functions:
  - name: f1
    base_version:
      path: fns/f1_v1
      env: nodejs
    new_version:
      path: fns/f1_v2
      env: nodejs
stages:
  - name: A/B Test f1
    type: Sequential
    func_name: f1
    variants:
      - name: base_version
        trafficPercentage: 50
      - name: new_version
        trafficPercentage: 50
    metrics_conditions: # AND condition
      - name: errorRate
        threshold: "<0.01"
      - name: responseTime
        threshold: "<=200"
        compareWith: "Median"
    end_conditions:
      - name: minDuration
        threshold: 10s
      - name: minCalls
        threshold: "50"
    end_action:
      onSuccess: rollout # keyword
      onFailure: rollback #Or "Canary f1 10"
#  - name: Canary f1 10
# ...
rollback:
  action:
    function: base_version
```

Evaluation: UC Works in Practice

- **Open-Source Prototype (Go):** Supports major FaaS providers (e.g., GCP Cloud Run) and edge FaaS (e.g., tinyFaaS).
- **Proxy Overhead:** Measurable, but enables crucial features.



- **Complex Scenario Success:** Successfully orchestrated a multi-stage, geo-distributed release (canary -> A/B -> gradual rollout) across edge (RPi) and cloud (GCP).



Conclusion: Enabling Safer Edge-to-Cloud Releases

- Edge-to-cloud FaaS needs robust, geo-aware live testing.

- **Umbilical Choir provides a solution:**
  - Automates complex release strategies.
  - Reduces risk of QoS regressions.

- **Impact:**
  Empowers developers to innovate faster and more safely in the evolving edge-to-cloud landscape.

- **Thank You & Questions!**

Contribution and Technical Meat

Preprint
To be published at ICFEC 2025