

Failover de UPF para Aplicações de Borda

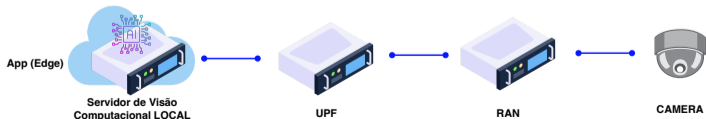
Apresentador: Flávio Geraldo C. Rocha

Centro de Excelência em Redes sem Fio Inteligentes e Serviços Avançados (CERISE)
Universidade Federal de Goiás (UFG)

26 de março de 2026

Contexto & Motivação

- **Redes 5G privadas:** além dos benefícios já conhecidos, redes 5G privadas podem proporcionar *edge computing* no domínio empresarial, hospedando aplicações.
- **Necessidade de um ponto de ancoramento:** A UPF fornece o PSA (PDU Session Anchor) para viabilizar a aplicação perto dos dispositivos, reduzindo latência.
- **Desafio 1** — recursos limitados na edge: CPU/memória/armazenamento restritos;
- **Desafio 2** — falhas imprevisíveis na edge: quedas de energia, congestionamento, crash de pods, outras falhas de hardware/software;
- **Limitação das abordagens de redundância:** 1:1 ou N:M com instâncias em espera implicam sobrecusto e subutilização dos recursos.



Background

- **Edge computing:** computação próxima à RAN reduz *hops* e melhora parâmetros de QoS, utilizando a UPF para *local breakout*. Em contraste com a AMF, a UPF é fluxo contínuo. Sua instanciação junto a App é decisiva em aplicações com baixo *budget* de latência.
- **Gestão da Session/PDU: CRIU** (Checkpoint/Restore in Userspace) permite suspender e restaurar processos/containers e uso da biblioteca go-gtp5gnl, objetivando reduzir *downtime*.
- **Cloud-native na edge:** UPF containerizado, uso do Kubernetes, e possivelmente Karmada para gestão dos clusters, proporcionando escalabilidade.

Estado da Arte

- **Redundância clássica:** ativa-passiva (1:1) ou N:M; usada para alta disponibilidade, mas **custa capacidade ociosa**.
- **Orquestração on-demand:** ONAP aciona UPF dinâmico no failover, porém há **atrasos** na preparação de rede do novo pod.
- **CRIU para restauração:** bom resultado, mas a solução usa **live VM migration**, aumentando complexidade.
- **Lacuna:** como obter **alta disponibilidade de UPF na borda** reduzindo o downtime?

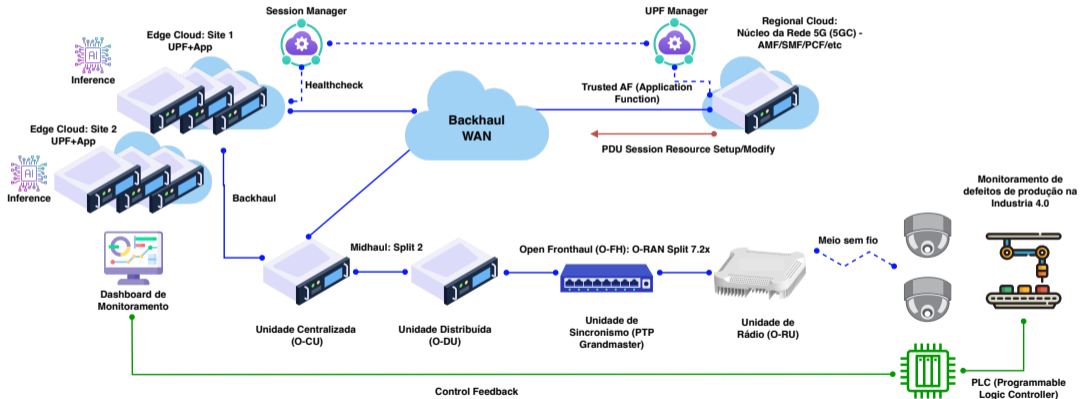
Proposta

- **Proposta:** detecção de falhas de UPF e mecanismos de *failover* para UPF containerizado:
 - Restauração da UPF no próprio site (em outro nó ou no nó original) ou restauração da UPF em outro site, conforme disponibilidade de recursos.
- **Avaliação:** avaliação comparativa com métodos do estado da arte.
 - Reimplantação da UPF vs. baselines.
 - Comparativo entre custo sob requisitos de alta confiabilidade vs. redundância tradicional.

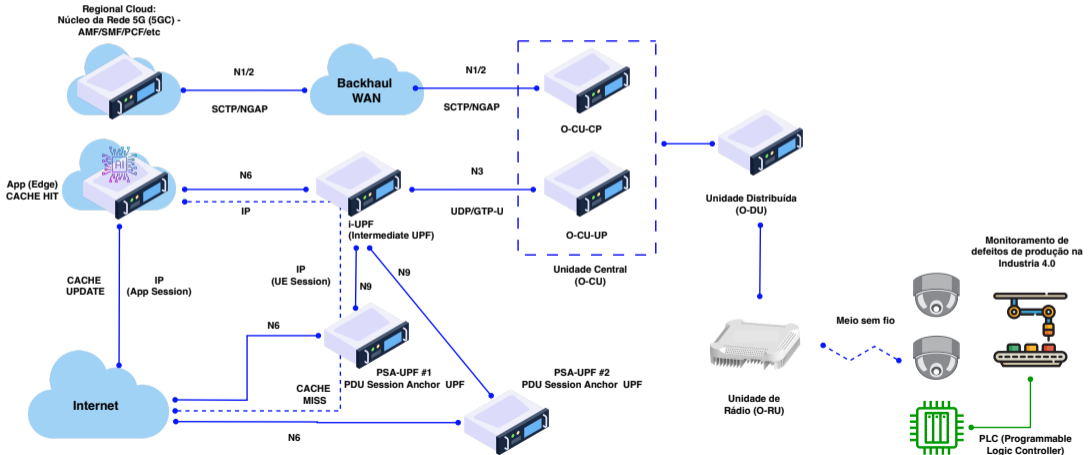
Arquitetura

- **Escopo:** Exemplo de Indústria 4.0 como monitoramento da produção com câmeras e PLC. Failover de **UPF containerizado** na borda para rapidez do control-loop desta aplicação.
- **UPF Manager:** monitora UPFs (status + heartbeat), coordena **checkpoint/restore**, interage com 5GC (como *trusted AF* ou via componentes externos).
- **Session Manager** (embutido ou co-local): sincroniza *em tempo real* sessões/PDR/FAR/QER/URR com o UPF Manager; após eventos PFCP, dispara **checkpoint**.

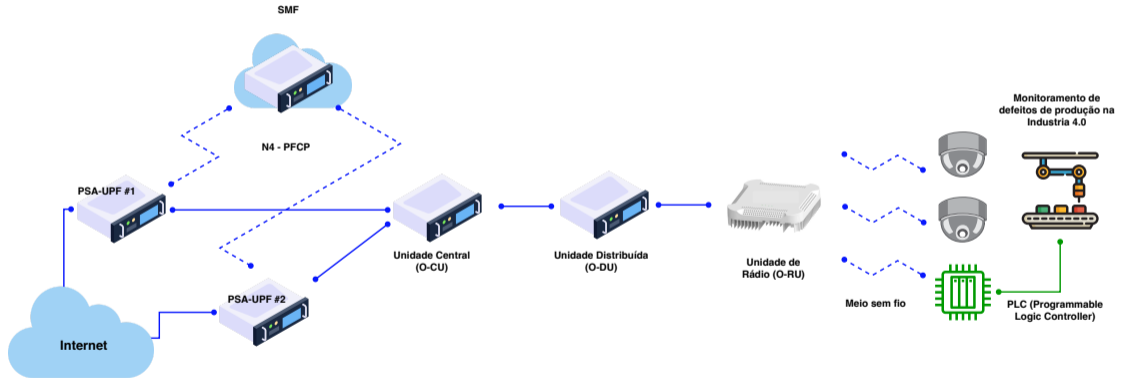
Cenário Proposto



Cenário em desenvolvimento - por Aplicação com Free5GC



Cenário em desenvolvimento - Redundância com reancoragem com Open5GS



Obrigado!