

Proposta de provisionamento de recursos para plataformas de cloud continuum baseado no framework Globus Compute

Mateus Tenorio dos Santos
PPGComp - UNIOESTE
Cascavel, PR, Brazil
mateus.santos28@unioeste.br

Guilherme Galante
PPGComp - UNIOESTE
Cascavel, PR, Brazil
guilherme.galante@unioeste.br

Abstract—This work aims to investigate and implement a hybrid computing architecture using Kubernetes and the Globus Compute platform, aiming to optimize the execution of distributed functions in cloud and edge environments. The research addresses the challenge of integrating cloud and edge computing systems, considering the limitations of latency, scalability, and energy efficiency. The research aims to fill the gap that exists in enabling the use of FaaS platforms, such as Globus Compute, in hierarchical architectures for the Cloud Continuum.

Keywords—Resource provisioning; Cloud Continuum; Globus compute.

Resumo—Este trabalho tem como objetivo investigar e implementar uma arquitetura híbrida de computação utilizando Kubernetes e a plataforma Globus Compute, visando otimizar a execução de funções distribuídas em ambientes de nuvem e borda (edge). A pesquisa aborda o desafio da integração entre sistemas de computação em nuvem e na borda, considerando as limitações de latência, escalabilidade e eficiência energética. A pesquisa pretende preencher a lacuna que existe da viabilização da utilização de plataformas FaaS, como o Globus Compute, em arquiteturas hierárquicas, para o Cloud Continuum.

Palavras-chave—Provisionamento de recursos; Cloud Continuum; Globus Compute.

I. INTRODUÇÃO

A evolução da Internet das Coisas (IoT) representa um dos marcos mais transformadores da era digital, caracterizada pela interconexão exponencial de dispositivos que geram volumes massivos de dados em tempo real. Projeções indicam que até 2030 haverá mais de 25 bilhões de dispositivos IoT em operação, conforme citado por [1], criando desafios significativos para as infraestruturas computacionais convencionais em termos de processamento, armazenamento e análise de dados.

Neste contexto, o conceito de *IoT Edge-Cloud Continuum* surge como uma solução inovadora para superar as limitações de abordagens isoladas, como computação de borda ou nuvem. Essa arquitetura integra diferentes camadas de infraestrutura

computacional, desde dispositivos na borda, que processam dados próximos à sua origem, até sistemas de nuvem que oferecem capacidade de processamento e armazenamento praticamente ilimitadas. Essa continuidade entre borda, nevoeiro (*fog computing*) e nuvem minimiza a latência, otimiza o consumo de banda e oferece escalabilidade para aplicações críticas [2].

Plataformas como o Globus Compute¹ destacam-se por sua abordagem baseada em Function-as-a-Service (FaaS), permitindo a execução remota e escalável de funções em ambientes distribuídos [3]. Originalmente desenvolvido para computação científica, o Globus Compute oferece capacidades de federação de recursos que podem ser exploradas em contextos IoT mais amplos.

Apesar do potencial do IoT Edge-Cloud Continuum, existe uma lacuna significativa na literatura relacionada à aplicação de plataformas FaaS como o Globus Compute em arquiteturas hierárquicas integradas. Estudos existentes exploram escalabilidade em nuvem ou borda isoladamente, mas não abordam como soluções baseadas em FaaS podem criar clusters hierárquicos que combinam funcionalidades específicas e gerais [4].

Este trabalho tem como objetivo explorar a viabilidade da utilização do Globus Compute em arquiteturas escaláveis para o IoT Edge-Cloud Continuum, propondo uma arquitetura hierárquica baseada em *clusters* especializados.

II. PROPOSTA

A arquitetura proposta implementa o conceito de Cloud Continuum através de uma estrutura hierárquica composta por quatro camadas principais: Camada de dispositivos IoT, Computação em Borda, Computação em Nevoeiro e Nuvem. Cada camada utiliza variantes do Kubernetes adequadas aos

¹Uma plataforma distribuída de FaaS que permite execução remota de funções. <https://docs.globus.org/compute/>

seus recursos computacionais, desde K3s (Distribuição leve certificada de Kubernetes para dispositivos que possuam restrição de poder computacional) para dispositivos com recursos limitados até Kubernetes completo para clusters robustos na nuvem. A arquitetura é exemplificada pela Figura 1

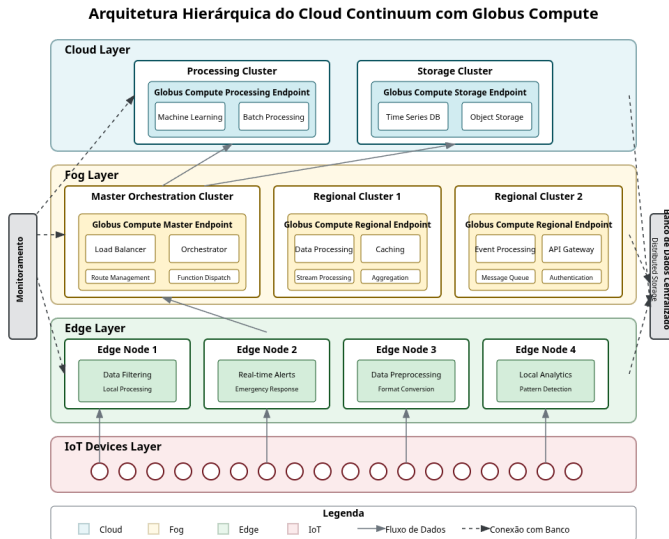


Fig. 1. Arquitetura conceitual

A. Especificação das Camadas

Conforme visto na Figura 1, a arquitetura hierárquica é composta por diversas camadas, sendo elas as seguintes:

- **Camada de dispositivos IoT:** Composta por sensores, atuadores e dispositivos conectados responsáveis pela coleta contínua de dados do ambiente físico. Esta camada representa a fonte primária de dados do sistema;
- **Camada da borda:** Implementa o primeiro nível de processamento, posicionada geograficamente próxima aos dispositivos IoT. Possui diversas funcionalidades, entre elas, as possibilidades são a filtragem e pré-processamento de dados em tempo real, detecção de anomalias e geração de alertas críticos e a realização de análises locais que não requerem poder computacional intensivo;
- **Camada da névoa:** Atua como ponte entre edge e cloud, implementando um cluster mestre para orquestração e clusters regionais especializados, responsáveis por processamentos que requerem um pouco mais de poder computacional;
- **Camada da nuvem:** Oferece a maior capacidade computacional.

B. Integração das Camadas com Globus Compute

A característica distintiva da arquitetura proposta está na integração das diferentes camadas utilizando o *Globus Compute* como plataforma Function-as-a-Service (FaaS) distribuída. Esta integração cria um ambiente de execução contínuo, onde funções podem ser implantadas e executadas de forma transparente em qualquer nível da hierarquia, dependendo dos requisitos específicos de cada tarefa. A Figura 3 ilustra detalhadamente este fluxo de dados e comunicação:

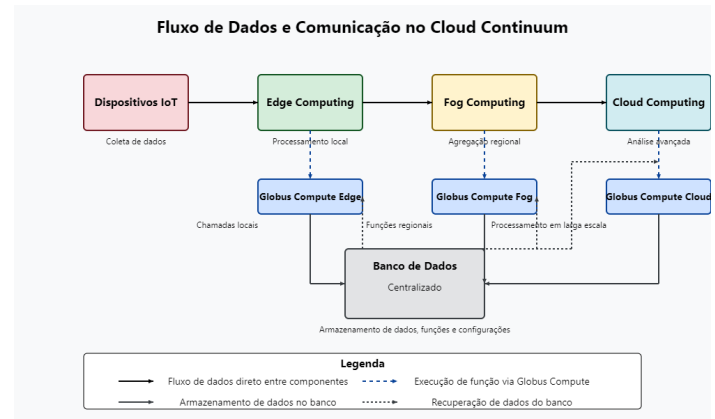


Fig. 2. Representação do fluxo de dados

Como observado por [3], o Globus Compute se destaca por sua capacidade de federação e execução remota de funções em ambientes heterogêneos. Explorando esta característica, a arquitetura implementa endpoints Globus Compute em cada camada.

Na camada da borda, os endpoints são configurados para possibilitar o processamento local de baixa latência, executando funções simples que exigem respostas imediatas. Além disso, realizam tarefas de filtragem e transformação inicial, como limpeza e formatação dos dados antes que sejam transmitidos para camadas superiores. Esses endpoints são otimizados para operar em dispositivos com restrições significativas de poder computacional e energia, o que demanda soluções eficientes e resilientes para enfrentar ambientes instáveis — aspectos fundamentais considerados na implementação.

Na camada da névoa, os endpoints são projetados para atuar na agregação e processamento regional, consolidando dados provenientes de múltiplos dispositivos dentro de uma mesma região. Eles também desempenham funções de caching e buffering, que auxiliam no gerenciamento intermediário dos dados, reduzindo assim a pressão sobre a infraestrutura em nuvem. Essa camada busca oferecer um equilíbrio entre latência e capacidade de processamento. [5] destacam que o uso de plataformas FaaS em ambientes intermediários como a névoa

permite elasticidade e isolamento das funções, características essenciais para adaptação rápida frente a variações na carga de trabalho.

Ja para a camada da nuvem, os endpoints serão otimizados para operações de análise em larga escala, capazes de processar grandes volumes de dados e integrar-se a sistemas especializados, como frameworks de machine learning e processamento distribuído. Essa camada também dá suporte ao processamento de longa duração, necessário para análises mais profundas e sofisticadas. Segundo [6], o verdadeiro potencial das arquiteturas *serverless* na nuvem reside justamente na capacidade de escalar instantaneamente, sem a necessidade de provisão prévia de recursos, para atender picos de demanda — princípio que guia a implementação dos endpoints Globus Compute nessa camada.

C. Solução para a Limitação de 10MB do Globus Compute

Um dos desafios técnicos mais significativos enfrentados na implementação da arquitetura foi a limitação de 10MB para entrada e saída de dados no Globus Compute. Esta restrição poderia comprometer seriamente a capacidade de processar volumes maiores de dados, comuns em aplicações IoT. Para superar este obstáculo, uma solução elegante baseada em um banco de dados centralizado foi desenvolvida conforme ilustrado na Figura 3

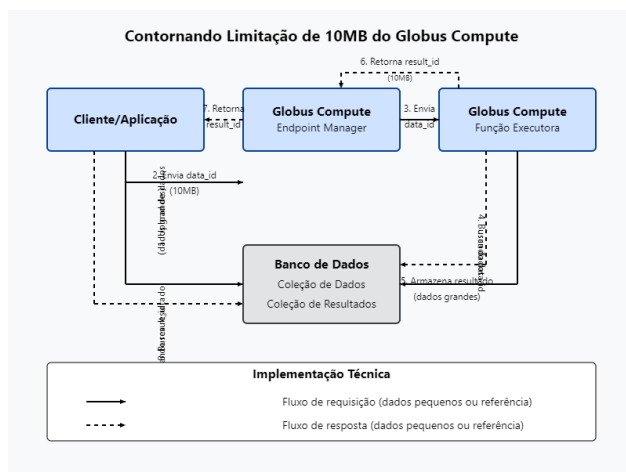


Fig. 3. Representação da solução

A essência da solução está no uso de referências leves (IDs) para representar conjuntos de dados potencialmente grandes. Esta abordagem oferece desacoplamento entre dados e processamento, persistência natural e redução significativa do tráfego de rede.

III. RESULTADOS E CONTRIBUIÇÕES ESPERADAS

Com o desenvolvimento desse trabalho, espera-se que seja possível contribuir tecnicamente com a criação de um modelo de referência, desenvolvendo uma arquitetura padrão para a implementação de soluções FaaS no Cloud Continuum, bem como a expansão da aplicabilidade do Globus Compute em cenários IoT, expandindo seu uso além de ambientes científicos. Espera-se também que seja possível validar estratégias de balanceamento de carga inteligente, baseado em características específicas de cada camada, usando como métricas principais algumas métricas de desempenho, como: latência, número de funções executadas em dado período de tempo e taxa de utilização de recursos.

REFERÊNCIAS

- [1] I. D. R. Zyrianoff, "Architecting the edge-cloud continuum for iot-based monitoring applications," Ph.D. dissertation, alma, Luglio 2024. [Online]. Available: <http://amsdottorato.unibo.it/11496/>
- [2] S. Moreschini, F. Pecorelli, X. Li, S. Naz, D. Hästbacka, and D. Taibi, "Cloud continuum: The definition," *IEEE Access*, vol. 10, pp. 131 876–131 886, 2022.
- [3] Z. Li, R. Chard, Y. Babuji, B. Galewsky, T. J. Skluzacek, K. Nagaitsev, A. Woodard, B. Blaiszik, J. Bryan, D. S. Katz *et al.*, "funcx: Federated function as a service for science," *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 12, pp. 4948–4963, 2022.
- [4] A. Ullah, T. Kiss, J. Kovács, F. Tusa, J. Deslauriers, H. Dagdeviren, R. Arjun, and H. Hamzeh, "Orchestration in the cloud-to-things compute continuum: taxonomy, survey and future directions," *Journal of Cloud Computing*, vol. 12, no. 1, p. 135, Sep 2023. [Online]. Available: <https://doi.org/10.1186/s13677-023-00516-5>
- [5] E. Jonas, J. Schleier-Smith, V. Sreekanti, C. Tsai, A. Khandelwal, Q. Pu, V. Shankar, J. Carreira, K. Krauth, N. J. Yadwadkar, J. E. Gonzalez, R. A. Popa, I. Stoica, and D. A. Patterson, "Cloud programming simplified: A berkeley view on serverless computing," *CoRR*, vol. abs/1902.03383, 2019. [Online]. Available: <http://arxiv.org/abs/1902.03383>
- [6] G. Adzic and R. Chatley, "Serverless computing: economic and architectural impact," in *Proceedings of the 2017 11th joint meeting on foundations of software engineering*, 2017, pp. 884–889.