

Como FaaS integrados a DBaaS se comportam em diferentes regiões: uma avaliação por meio da Framework Orama

Bruno Abreu Kamienski¹, Leonardo Rebouças de Carvalho¹, Aleteia Araujo¹

¹ Departamento de Ciência da Computação – Universidade de Brasília (UnB)
Brasília – DF – Brazil

{brunosabreu, leouesb}@gmail.com, aleteia@unb.br

Abstract. *Public cloud providers have made efforts to expand the coverage of their services based on the serverless paradigm in order to meet the need of next generation of cloud computing. However, it is very important studying the behavior of FaaS environments in different regions of providers. This work presents a study aided by the Orama framework in order to evaluate the performance of the main FaaS integrated with DBaaS services in different regions spread across the globe. The results indicate that the Alibaba provider was able to guarantee good equivalence between its regions, in addition to a lower average execution time. AWS and GCP had similar results, and Azure, on the other hand, had the worst performance and significant failure rates.*

Resumo. *Os provedores de nuvem pública têm feito esforços para expandir a cobertura de seus serviços baseados no paradigma sem servidor, a fim de atender à necessidade da próxima geração de computação em nuvem. No entanto, é muito importante estudar o comportamento dos ambientes FaaS em diferentes regiões de provedores. Este trabalho apresenta um estudo auxiliado pela framework Orama a fim de avaliar o desempenho dos principais FaaS integrados aos serviços DBaaS em diferentes regiões espalhadas pelo globo. Os resultados indicam que o provedor Alibaba conseguiu garantir uma boa equivalência entre suas regiões, além de um menor tempo médio de execução. AWS e GCP tiveram resultados semelhantes, enquanto a Azure, por outro lado, teve o pior desempenho além de taxas de falha significativas.*

1. Resumo estendido

A computação sem servidor [Nupponen and Taibi 2020] como o paradigma padrão de programação em nuvem tem se tornado uma ideia cada vez mais presente nas publicações recentes e isso mostra a importância que ela ganhou para a computação em nuvem. Function-as-a-Service (FaaS) [Schleier-Smith et al. 2021] permite que os usuários publiquem funções escritas em alguma linguagem de programação suportada pelo provedor e configurem um gatilho que, quando acionado, cabe ao provedor garantir o devido processamento, mesmo diante de altos níveis de concorrência.

Visto a necessidade dos provedores estarem o mais próximo possível do usuário final, frequentemente é utilizada pelas empresas de nuvem a estratégia de implantar infraestruturas geograficamente distribuídas ao redor do mundo. Além disso, é comum que

diferentes serviços de nuvem sejam combinados para compor a solução. Portanto, provedores de nuvem oferecem diversas soluções para armazenamento de dados, entre as quais se destacam Database-as-a-Service (DBaaS) [ZHENG 2018] em que os provedores entregam ambientes de banco de dados totalmente gerenciados por eles.

Levando em consideração a perspectiva de crescimento da adoção de FaaS, bem como a possibilidade de diferentes implementações entre regiões impactarem o desempenho de aplicações operando em ambientes dessa natureza, foram avaliados os principais FaaS em diferentes regiões. Cinco importantes regiões do planeta onde Amazon WebServices (AWS), Google Cloud Platform (GCP), Azure (AZF) e Alibaba (AFC) possuem infraestruturas implantadas foram escolhidas para receber um dos casos de uso disponíveis do framework Orama [Carvalho. and Araujo. 2022]. Utilizando FaaS integrado com o respectivo DBaaS, foram executadas diversas baterias de testes simulando acessos simultâneos a serviços desde 1 requisição simultânea até 4096 acessos paralelos. Os processos de provisionamento de ambientes FaaS, execução de testes, análise de resultados e desprovisionamento de ambientes foram realizados utilizando o framework Orama.

Os resultados indicam que o Alibaba aparentemente implementa uma estratégia de gestão mais eficiente para sua plataforma FaaS em todas as regiões avaliadas, pois seu tempo médio de execução foi o menor entre os provedores, assim como sua taxa de falhas. AWS e GCP obtiveram resultados intermediários e muito próximos. Já o AFC registrou os piores resultados, tanto em tempo médio de execução quanto em taxas de falhas.

O framework Orama [Carvalho. and Araujo. 2022] é uma ferramenta cujo objetivo é auxiliar na execução de benchmarks em ambientes FaaS. A estrutura permite alguns casos de uso integrados que podem ser provisionados e desprovisionados automaticamente. Além disso, o framework coordena a execução dos benchmarks a partir dessas configurações. O framework pode ser configurado para funcionar de forma autônoma, porém sua capacidade de ativar o FaaS estará limitada à quantidade de recursos disponíveis na máquina onde está instalado. Também é possível configurá-lo para atuar de forma distribuída seguindo a arquitetura “master/workers” em que os workers serão responsáveis por ativar o FaaS e assim a carga de concorrência pode ser dividida entre os nós workers configurados no ambiente do framework, aumentando assim a capacidade de concorrência da plataforma.

Considerando que a fase de análise de resultados é uma etapa crucial para entender o desempenho de ambientes FaaS, o framework Orama fornece duas ferramentas de análise estatística. O *design* fatorial [Jain 1991] auxilia na identificação de fatores que influenciam os resultados. No framework Orama, dois fatores são considerados: nível de concorrência e provedor, então o planejamento fatorial implementado está no formato 2^2 e caso os resultados sejam compostos por mais de uma rodada (repetição), então será possível analisar o erro estatístico do *design* fatorial. Se o erro estatístico for alto, isso indica a existência de outro fator além dos inicialmente mapeados. Outra ferramenta estatística disponível na estrutura Orama é o teste T pareado. Neste teste, analisa-se a significância estatística da diferença encontrada em dois resultados justapostos.

Soluções envolvendo o uso de FaaS em conjunto com banco de dados são comuns, justificando a escolha do caso de uso do framework Orama que implanta FaaS em diferentes provedores integrados com soluções DBaaS em provedores como DynamoDB

da AWS, Firestore do GCF, CosmosDB da AZF e TableStore da AFC. Com o objetivo de submeter os ambientes FaaS a diferentes níveis de concorrência, foram definidos 13 cenários de teste com concorrência de 1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048 e 4096 solicitações simultâneas para FaaS. Cada bateria de testes foi configurada no framework Orama para ser repetida 10 vezes a fim de construir uma média de tempos de execução.

Uma vez implementado o framework, foram solicitados os deploys dos respectivos ambientes do caso de uso de FaaS com DBaaS para cada região de cada provedor, totalizando 20 provisionamentos. O Orama foi configurado para realizar uma requisição de pré-acionamento aos serviços e separando-a das demais requisições, permitindo uma análise da partida a frio. Cada provisionamento foi submetido a uma bateria de repetições, para que os resultados pudessem ser analisados.

Os resultados mostraram que, de maneira geral, as diferentes regiões dos provedores avaliados entregam desempenhos equivalentes, com exceção da região US-East de Lambda, cujos resultados superaram as demais regiões do provedor, possivelmente devido ao seu maior nível de maturidade.

A análise dos tempos médios de execução mostrou que o AFC liderou os resultados, apresentando os menores tempos médios em todos os níveis de concorrência avaliados, seguido pelo Lambda (AWS) e GCF, praticamente iguais em seus desempenhos. Por fim, AZF registrou os maiores tempos médios em todas as regiões testadas.

A análise da taxa de falha confirmou a liderança da AFC nesta avaliação, registrando as menores taxas seguidas pela GCF. AWS e AZF forneceram taxas de falha notavelmente altas, especialmente em níveis mais altos de concorrência.

A análise da partida a frio mostrou mais uma vez a eficiência das estratégias adotadas pela AFC. AWS e GCF registraram tempos de partida a frio equivalentes, enquanto o AZF teve um tempo muito alto para implantar seu ambiente de processamento FaaS em todas as suas regiões. Diante desses resultados, ficou evidente que a AFC registrou os melhores resultados em todos os itens analisados nesta avaliação, seguida da AWS e GCF em nível de qualidade equivalente, e por fim os menores resultados foram registrados pela AZF.

Referências

- Carvalho., L. and Araujo., A. (2022). Orama: A benchmark framework for function-as-a-service. In *Proceedings of the 12th CLOSER*, pages 313–322. INSTICC, SciTePress.
- Jain, R. (1991). *The art of computer systems: Techniques for experimental design, measurement, simulation, and modeling*.
- Nupponen, J. and Taibi, D. (2020). Serverless: What it is, what to do and what not to do. In *2020 IEEE ICSCA-C*, pages 49–50.
- Schleier-Smith, J., Sreekanti, V., Khandelwal, A., Carreira, J., Yadwadkar, N. J., Popa, R. A., Gonzalez, J. E., Stoica, I., and Patterson, D. A. (2021). What serverless computing is and should become: The next phase of cloud computing. *ACM*, 64(5):76–84.
- ZHENG, X. (2018). Database as a service - current issues and its future. *CoRR*, abs/1804.00465.