

Sharding para Acesso Rápido a Dados Móveis e Distribuídos: Uma Aplicação para Smart Cities com Fog Computing

Daniel Lopes Ferreira¹, Rodrigo da Rosa Righi¹

¹PPGCA – Universidade do Vale do Rio dos Sinos (UNISINOS)
93.022-750 – São Leopoldo – RS – Brasil

danielferreiral7@edu.unisinos.br, rrrighi@unisinos.br

***Resumo.** Sharding é uma estratégia de armazenamento de dados distribuídos em uma arquitetura horizontal. Dessa maneira, essa prática tem um enorme potencial de reduzir a latência no acesso de dados distribuídos em cidades inteligentes. Nessa pesquisa, trazemos uma implementação de Sharding para ser a base prática de uma arquitetura de armazenamento distribuído e rastreabilidade de dados através de Fog Computing para Smart Cities.*

1. Introdução

Com o crescimento da população urbana nos grandes centros as *Smart Cities* tornam-se uma realidade cada vez mais próxima. A possibilidade de integração entre as diversas áreas de uma cidade trazem a expectativa de melhorias na gestão pública em todas as suas atribuições como mobilidade, saúde, educação e infraestrutura [Lai et al. 2020].

Contudo, as tarefas de armazenamento e consulta desses dados precisam ser o mais performáticas e transparentes possível. Neste contexto a gestão dos dados é o fator chave para o sucesso de um ecossistema inteligente. Isso se deve a enorme quantidade de dados gerados nas *Smart Cities* a partir de seus sensores [Sasubilli et al. 2020]. Estes necessitam ser armazenados de forma competente a fim de garantir a eficiência no processamento e análise desses dados.

Por consequência disso, se faz necessária a elaboração de uma arquitetura robusta capaz de atender a todos os serviços da melhor maneira possível, preocupando-se com tolerância a falhas, consistência, integridade, rastreabilidade e confiabilidade. Contudo, não se encontram muitos trabalhos na literatura acadêmica que lidem especificamente com a arquitetura de armazenamento distribuído de dados e outros aspectos como alto desempenho, hierarquia, rastreabilidade e cache.

Manipular dados na borda da rede reduz a latência, aumenta a segurança e diminui a carga na rede quando comparada a um modelo distante como o baseado apenas em Cloud Computing [Vilela et al. 2020]. Uma maneira de implementar essa manipulação mais próxima dos dados é através de nós de Fog Computing [Naeem et al. 2019] que distribuem-se na cidade e são mais eficientes nos tratamentos de dados.

A fim de lidar com a mobilidade de um usuário em uma cidade, este trabalho expressa uma parte da pesquisa sobre a rastreabilidade desses dados através do armazenamento distribuído. O recurso chave para este armazenamento chama-se Sharding. Sharding é uma técnica de divisão de dados entre diferentes servidores objetivando o armazenando distribuído, redução da carga de trabalho em cada nó da rede e acesso paralelo aos dados que necessitam ser escritos e/ou lidos.

A presente pesquisa cita alguns trabalhos relacionados na seção 2. Na seção 3 temos o modelo central da pesquisa. Os testes e resultados preliminares através de uma abordagem do Sharding estão na seção 4. E, por fim, na seção 5, apresentamos uma visão de trabalhos futuros e contribuições esperadas.

2. Trabalhos Relacionados

As decisões dessa pesquisa foram tomadas baseando-se em trabalhos relacionados que através das metodologias e dos resultados apresentados são considerados aderentes ao objetivo de armazenar dados de maneira distribuída e com técnicas de alto desempenho. Em [Kudo 2018] há a proposição de armazenamento distribuído e hierárquico entre Fog e Cloud no Banco de Dados NoSQL MongoDB, através do recurso de GridFS que divide grandes documentos em mais de uma parte para melhorar o desempenho de leitura e escrita. Em [Sinaeepourfard et al. 2018] há a preocupação com todo o ciclo de vida dos dados e indica a estrutura Fog-to-Cloud como uma ótima opção de armazenamento de dados distribuídos em uma Smart City. Em [Shwe et al. 2016] é apresentado um *framework* (Database as a Service) para gestão dos dados fragmentados e hierarquizados em Smart Cities.

3. Modelo

O modelo apresentado nesta seção é apenas uma parte de um modelo mais abrangente que é focado na distribuição e rastreabilidade dos dados armazenados em Fog Nodes. Pelo andamento da pesquisa, será apresentado o armazenamento de uma coleção de dados fragmentada em dois bancos de dados. Isso possibilitará a realização de testes preliminares quanto à performance do uso ou não de Sharding.

Em resumo, para utilizar a estratégia de Sharding em bancos de dados é necessário definir-se uma chave de divisão dos registros. Neste caso, definiu-se como o próprio identificador do servidor (campo *shard_server*). Já que no caso do modelo dessa pesquisa os dados são gerados a partir da conexão a uma parte do banco de dados em um local da cidade, a chave será o próprio identificador do nó na rede.

Na Figura 1 é demonstrada a organização de uma estrutura básica de Sharding com replicação. Diferentemente de como é apresentado na Figura 2 com o armazenamento centralizado em uma Cloud distante dos usuários, essa técnica permite a distribuição do armazenamento em diversos Fog Nodes próximos geograficamente dos usuários.

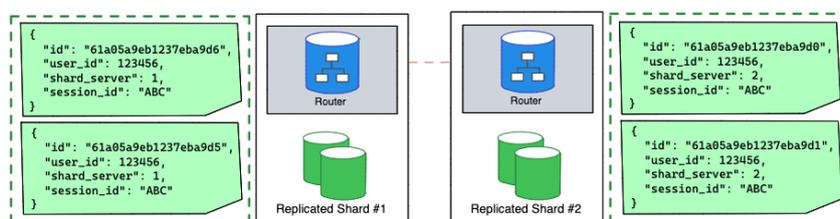


Figura 1. Sharding: dados distribuídos e próximos dos usuários

4. Testes e Resultados Preliminares

Os testes para validar a eficiência do modelo utilizando Sharding foram realizados em uma infraestrutura baseada em containerização Docker para individualização do proces-

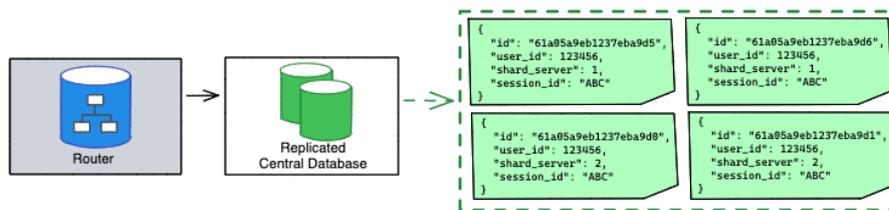


Figura 2. Implementação sem Sharding: centralizado e distante dos usuários

samento em um único computador. A configuração do host é composta por um processador hexa-core Intel Core i7 com clock de 2,6 GHz, 16 GB de memória RAM DDR4 Dual-Channel com frequência de 2667 MHz e SSD com taxa de leitura e escrita de 3200 MB/s e 2200 MB/s, respectivamente.

Foi utilizado MongoDB, pois é um banco de dados que implementa Sharding de uma maneira muito simples e eficiente. Conforme demonstrado na Figura 3, por ser um requisito do uso de Sharding no MongoDB, foi agregado ao modelo um servidor de configuração que armazena os metadados de localização dos registros na estrutura.

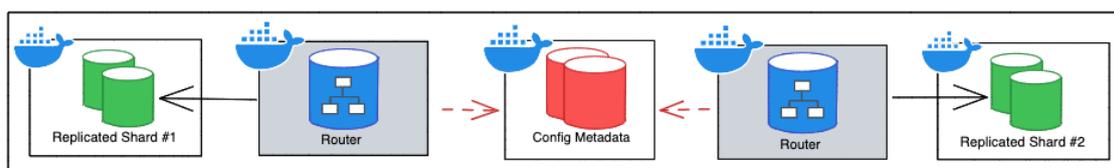


Figura 3. Aplicação de Sharding utilizando MongoDB em Docker

Foram estabelecidos dois cenários de testes preliminares para validar a eficiência do modelo utilizando Sharding (com dois nós) em comparação ao de um banco de dados centralizado. Conforme a Figura 4, o primeiro cenário (1) realiza uma busca utilizando um filtro pelo servidor na consulta, já no segundo (2) a consulta foi realizada sem filtragem, ou seja, retornando toda a base de dados. Ambos foram executados para um banco de dados com pouco mais de 2 milhões de registros e a partir de uma amostragem de resultados obtida após 10 execuções de consulta.

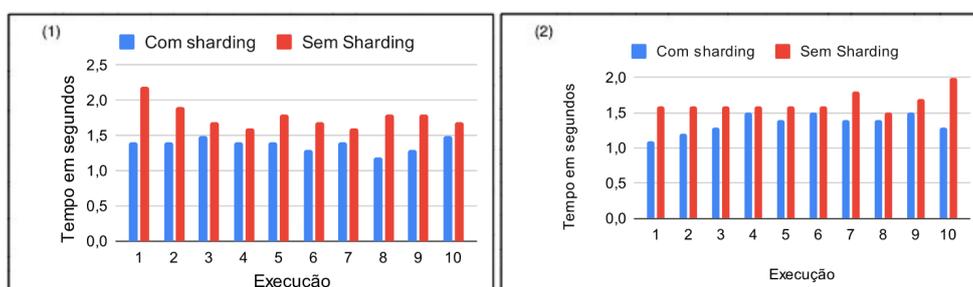


Figura 4. Desempenho dos testes preliminares

Os resultados preliminares apontam a estrutura com Sharding com uma melhora de aproximadamente 20% no tempo de resposta das consultas. Vale ressaltar que neste teste não pode ser considerado o fator proximidade com os dados que em um cenário real

traria mais vantagens ainda ao uso de Sharding. Além disso, com Sharding obteve-se um tempo médio de retorno de 1,36 e 1,38 segundos (com desvio padrão de 0,1350 e 0,0919 segundos) no primeiro e segundo teste, respectivamente e sem Sharding de 1,66 e 1,78 segundos (com desvio padrão de 0,1430 e 0,1751 segundos), respectivamente.

Um fato interessante que se pode analisar é que na implementação com Sharding o tempo de busca de todos os registros nos dois nós ou de um único nó é praticamente o mesmo. Isso ocorre porque como os dados estão distribuídos, a consulta é realizada de maneira concorrente para ambos os servidores, tendo a carga de trabalho dividida.

5. Conclusão

Especificamente nesta validação preliminar, foi empregado o uso de Docker. Contudo, para os trabalhos futuros, o foco é realizar testes de carga com os nós dispostos em distância geográfica equivalente a de bairros em uma cidade. Esta abordagem produzirá resultados mais fiéis a aplicação final do modelo proposto.

Neste trabalho foi possível verificar um pouco da eficiência do Sharding em relação ao armazenamento centralizado. Ao fim de toda a pesquisa, a contribuição esperada à comunidade é a proposição de um padrão eficiente de arquitetura de armazenamento distribuído. Essa arquitetura, aplicada em cidades inteligentes, buscará proporcionar fácil rastreabilidade, baixa latência e alta disponibilidade na busca de dados.

Referências

- Kudo, T. (2018). Fog computing with distributed database. In *2018 IEEE 32nd International Conference on Advanced Information Networking and Applications (AINA)*, pages 623–630.
- Lai, C. S., Jia, Y., Dong, Z., Wang, D., Tao, Y., Lai, Q. H., Wong, R. T. K., Zobia, A. F., Wu, R., and Lai, L. L. (2020). A review of technical standards for smart cities. *Clean Technologies*, 2(3):290–310.
- Naeem, R. Z., Bashir, S., Amjad, M. F., Abbas, H., and Afzal, H. (2019). Fog computing in internet of things: Practical applications and future directions.
- Sasubilli, S. M., Kumar, A., and Dutt, V. (2020). Improving health care by help of internet of things and bigdata analytics and cloud computing. In *2020 International Conference on Advances in Computing and Communication Engineering (ICACCE)*, pages 1–4.
- Shwe, H. Y., Jet, T. K., and Chong, P. H. J. (2016). An iot-oriented data storage framework in smart city applications. In *2016 International Conference on Information and Communication Technology Convergence (ICTC)*, pages 106–108.
- Sinaeepourfard, A., Garcia, J., Masip-Bruin, X., and Marin-Tordera, E. (2018). Data preservation through fog-to-cloud (f2c) data management in smart cities. In *2018 IEEE 2nd International Conference on Fog and Edge Computing (ICFEC)*, pages 1–9.
- Vilela, P. H., Rodrigues, J. J. P. C., Righi, R. d. R., Kozlov, S., and Rodrigues, V. F. (2020). Looking at fog computing for e-health through the lens of deployment challenges and applications. *Sensors*, 20(9).