

Implementação de Broker para alta disponibilidade no consumo de software externo via API REST

Pedro Figueiredo Dias, Nathan Silva Macena, Ye Wei Jiang, Claudio Ramos

Prof.(a) Ana Cláudia Rossi

Motivação e objetivo

Em engenharia de software, delegar certas funcionalidades que não são o núcleo do negócio para provedores externos permite focar no que realmente pode agregar ao produto. Por exemplo, funções de comunicação e pagamento em um e-commerce.

Para viabilizar a contratação, os provedores precisam garantir o bom funcionamento do serviço, com altos níveis de disponibilidade e outros compromissos nos chamados *Service Level Agreements* (SLA).

No entanto, a utilização de um único provedor para cada necessidade implica em pontos de falha única no sistema, dado que a garantia de funcionamento em 100% do tempo não é uma realidade, o que pode causar prejuízos (BEYER et al., 2018). Logo, é importante ter redundância de provedores externos.

O objetivo, portanto, é desenvolver uma solução para gerenciar múltiplos provedores e contribuir para manter o nível de disponibilidade desejado por sistemas que dependem de parceiros externos em eventual falha destes.

Metodologia

Pelo ciclo do design de (WIERINGA, 2014), às fases definidas para a pesquisa são:

- Investigação do problema
- Proposta de solução para o problema
- Validar a solução
- Implementação do artefato
- Avaliação da implementação

Nesta etapa foi endereçada uma proposta de solução baseada no conceito de Broker de aplicações distribuídas de (BUSCHMANN; HENNEY; SCHMIDT, 2011), no padrão de design para alta disponibilidade com redundância ativo-passivo de (AHLUWALIA; JAIN, 2006), nos conceitos de SRE de (BEYER et al., 2018) e com base na arquitetura baseada em espaço, para alto desempenho (RICHARDS, 2015).

Resultados preliminares

A Figura 1 apresenta a arquitetura da solução proposta.

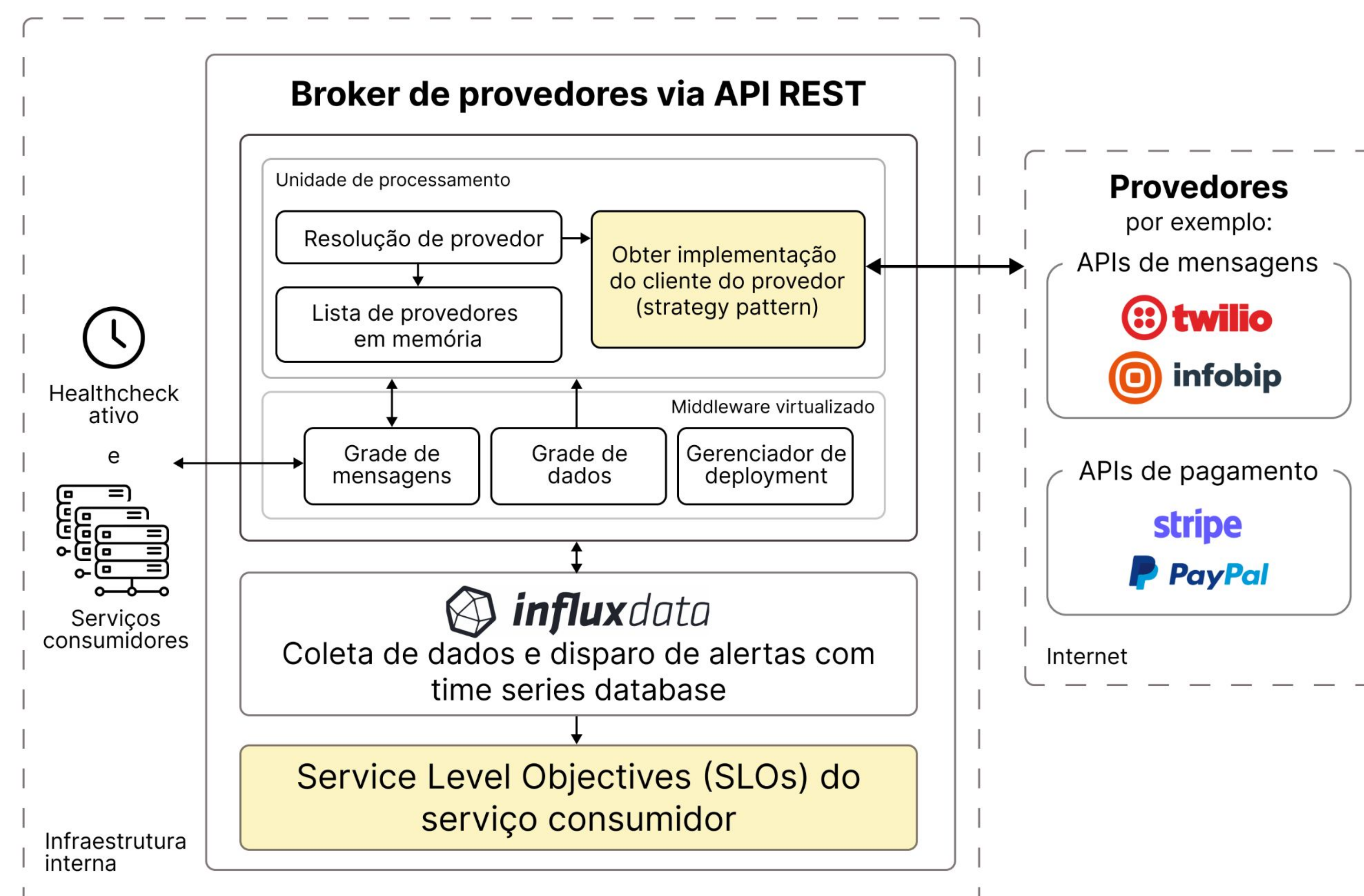


Figura 1: Arquitetura proposta como solução.

Fonte: autores

Conclusões e próximos passos

Ao final desta etapa foi proposto um modelo conceitual de solução voltado para escalabilidade e alta performance para atingir o objetivo.

Os próximos passos da pesquisa serão a implementação da proposta e a validação do artefato para determinar sua eficiência.

Referências

- AHLUWALIA, K. S.; JAIN, A. High availability design patterns. Proceedings of the 2006 conference on Pattern languages of programs. Portland, Oregon: ACM Press, 2006.
- BEYER, B. et al. (EDS.). The site reliability workbook: practical ways to implement SRE. Sebastopol, CA: O'Reilly Media, 2018.
- BUSCHMANN, F.; HENNEY, K.; SCHMIDT, D. C. A pattern language for distributed computing. Reprinted August 2011 ed. Chichester: Wiley, 2011.
- RICHARDS, M. Software architecture patterns. California: O'Reilly Media, 2015.
- WIERINGA, R. J. Design Science Methodology for Information Systems and Software Engineering. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014.