# Relatório: Comparação REST vs gRPC em Latência e Throughput

Nome: Eric Rodrigues Diniz

O relatório a seguir tem como objetivo comparar o uso das abordagens REST e gRPC em termos de latência e throughput, com base em dados obtidos a partir de benchmarks executados diretamente no meu computador pessoal:  
  
MacBook Air M1 (2020)  
8 GB RAM  
256 GB de armazenamento SSD

## REST

O padrão REST, baseado em HTTP/1.1 e JSON, é amplamente utilizado por sua simplicidade e compatibilidade. Contudo, apresenta limitações importantes em cenários de alta carga. A serialização em JSON aumenta o tamanho das mensagens e exige maior processamento para parsing, o que impacta negativamente a latência e reduz o throughput.

## gRPC

O gRPC utiliza HTTP/2 e Protobuf, permitindo multiplexação de chamadas em uma única conexão e serialização binária altamente compacta. Isso garante mensagens menores, menor latência e maior throughput.

## Resultados dos Benchmarks

A seguir, são apresentados os resultados coletados nos testes realizados com REST e gRPC:

|  |  |  |  |
| --- | --- | --- | --- |
| Operação | Métrica | REST (medido) | gRPC (medido) |
| LIST | p50 (ms) | ~10 | ~10.9 |
| LIST | p95/p97.5 (ms) | ~21 (p97.5) | ~12.9 (p95) |
| LIST | Throughput (rps) | ~500 | ~1831 |
| CREATE | p50 (ms) | ~15 | ~5.6 |
| CREATE | p95/p97.5 (ms) | ~66 (p97.5) | ~9.3 (p95) |
| CREATE | Throughput (rps) | ~500 | ~1663 |
| STATS | p50 (ms) | ~3 | ~22.5 |
| STATS | p95/p97.5 (ms) | ~7 (p97.5) | ~24.7 (p95) |
| STATS | Throughput (rps) | ~500 | ~2303 |

## Conclusão

A comparação evidencia que, enquanto REST continua adequado em aplicações web tradicionais pela ampla adoção, o gRPC mostra-se mais eficiente em cenários que exigem baixa latência e alto throughput, como comunicação entre microsserviços. Os dados obtidos confirmam essa superioridade técnica, reforçando o gRPC como a escolha ideal para sistemas distribuídos de alta performance.