

## Paradigmas de Sistemas Distribuídos

Teste<sup>1</sup>

14 de Janeiro de 2019

Duração: 1h30m

---

**1** Suponha que pretende utilizar um middleware de RPC, para fazer invocações remotas a um serviço multi-threaded, por clientes multi-threaded, envolvendo um baixo ritmo de criação de threads com tempo de vida longo nos clientes, possivelmente sincronização entre operações e pedidos de tamanho muito variável. Identifique potenciais problemas, e justifique o que seria a escolha apropriada em termos de gestão das conexões.

**2** Em tom de brincadeira, Cap'n Proto descreve-se como sendo “INFINITY TIMES faster than Protocol Buffers.”. Explique a filosofia destes formatos com representação neutra em memória, e enumere algumas vantagens face ao modo clássico de fazer serialização.

**3** Descreva as características fundamentais dos actores, e explique porque estas ultrapassam vários problemas dos sistemas de threads+objects.

**4** Diga em que consiste o *selective receive* suportado vulgarmente por sistemas de actores, e explique a sua importância.

**5** Descreva alguns papéis típicos de um *messaging-oriented middleware*, exemplificando a sua ocorrência na interação entre diferentes tipos de sockets ZeroMQ, usados para obter um dado padrão de interação.

**6** Explique o papel de um socket ZeroMQ do tipo ROUTER, e o modo como estes exploram a possibilidade das mensagens serem *multi-part* para desempenharem o seu papel.

**7** Explique em que consiste a restrição “interface uniforme” no padrão arquitectural REST, descrevendo os seus benefícios, e compare (neste aspecto) o desenvolvimento de uma aplicação usando REST versus usando o paradigma de invocação remota de procedimentos/métodos.