

Paradigmas de Sistemas Distribuídos

Exame¹

28 de Janeiro de 2019

Duração: 1h30m

- 1 Compare os modelos de despacho de invocações remotas *thread por conexão* e *thread por invocação*, e descreva uma situação em que seja preferível o uso do primeiro.
- 2 Existem diferentes formatos binários para serialização de estruturas de dados. Descreva diferentes aspectos vulgarmente considerados no desenho de um formato binário e respectivas alternativas. Indique ainda em que situações usaria um formato binário, em detrimento de um formato baseado em texto.
- 3 Descreva as características fundamentais dos actores, e explique porque estas ultrapassam vários problemas dos sistemas de threads+objectos.
- 4 O modelo de actores tornou-se popular com a linguagem Erlang, tendo depois sido oferecido em Java através de bibliotecas como Akka e Quasar. Compare estas três alternativas.
- 5 Descreva em que consiste um modelo de subscrição no paradigma *publish/subscribe*. Explique como estes modelos podem ser aproveitados para melhorar a escalabilidade de um sistema de notificação distribuído. Dê um exemplo concreto, no contexto do ZeroMQ e modelo de subscrição suportado.
- 6 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.
- 7 Suponha que pretende implementar um serviço que envolva acesso por clientes em sessões de curta duração, com pedidos independentes envolvendo computação intensiva, e também para acesso sobretudo para consulta de informação estruturada. Diga, justificando, qual/quais dos seguintes paradigmas/tecnologias escolheria para o servidor: multithreaded em Java, actores em Erlang, ZeroMQ em Java, API REST em Java.