

Instituto Superior de Engenharia de Lisboa

Área Departamental de Engenharia de Electrónica e Telecomunicações e de Computadores

Mestrado em Engenharia Informática e de Computadores (MEIC)

Mestrado em Engenharia Informática e Multimédia (MEIM)

Computação Distribuída (CD)

Semestre Inverno 2021/2022

Luís Assunção (lass@isel.ipl.pt ; luis.assuncao@isel.pt)

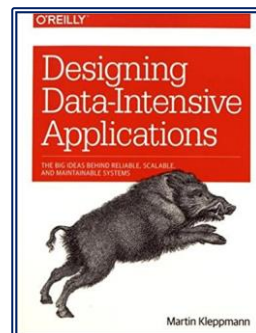
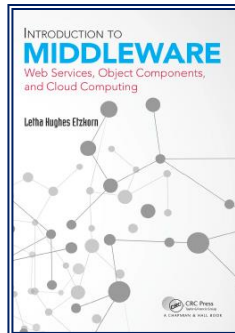
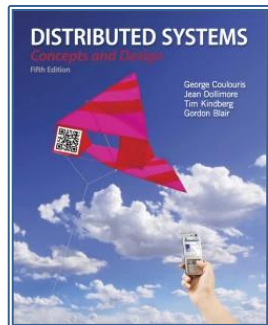
José Simão (jsimao@cc.isel.ipl.pt ; jose.simao@isel.pt)

1. Descrever e discutir as vantagens, os problemas e desafios que se colocam no desenvolvimento de aplicações usando o paradigma da computação distribuída;
2. Conhecer os padrões de arquitetura, de interação e comunicação entre as partes das aplicações distribuídas;
3. Tempo, ordenação e coordenação de eventos e consensos em computação distribuída
4. Desenvolver aplicações, usando o paradigma da computação distribuída com recurso a *middleware* e API para acesso a componentes e serviços;
5. Alojamento de serviços em infraestruturas distribuídas, locais ou em *Clouds* públicas com recurso a *Virtual Machines* e *Containers*

- ✓ **Caracterização da Computação Distribuída:**
 - **Potencialidades; Desafios; Comunicação; Resiliência; Escalabilidade; Elasticidade; Replicação; Teorema CAP**
- ✓ **Arquiteturas e modelos de interação na computação distribuída: Cliente/Servidor; *N-tier*; *Peer-to-Peer*; Clusters HPC; *Publish/Subscribe***
- ✓ **Infraestruturas de suporte à execução das componentes distribuídas: Máquinas físicas; Máquinas virtuais; Containers; Serviços de infraestrutura Cloud (IaaS)**
- ✓ **Granularidade, padrões *middleware* de comunicação e interação: Sockets; *Remote Procedure Call*; Objetos distribuídos; Serviços REST, SOAP; *Microservices*; Comunicação assíncrona e *Callbacks*; Componentes *Stateless/Stateful*; *Message Queuing*; *Publish/Subscribe***
- ✓ **Tempo e ordenação de eventos. Relógios lógicos e vetoriais na ordenação de mensagens. Algoritmos distribuídos de exclusão mútua e de consenso baseados em eleições. Comunicação por grupos com *multicast***
- ✓ **Aulas práticas com concretização de aplicações sobre plataformas tecnológicas, tanto em infraestruturas locais como em infraestruturas de *Cloud* pública**

Bibliografia

- [B1] ***Distributed Systems: Concepts and Design***, George Coulouris, Jean Dollimore, Tim Kindberg and Gordon Blair, Fifth Edition, published by Addison Wesley, May 2011
- [B2] ***Introduction to Middleware: Web Services, Object Components, and Cloud Computing***, Letha Hughes Etzkorn, CRC Press, 2017
- [B3] ***Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems***, Martin Kleppmann, 1st Edition, 2017



Moodle Geral: <https://2122moodle.isel.pt/course/view.php?id=4522>

Moodle Turma Diurna: <https://2122moodle.isel.pt/course/view.php?id=4996>

Moodle Turma Noturna: <https://2122moodle.isel.pt/course/view.php?id=4997>

- ✓ Artigos e *Links* sobre temas específicos
- ✓ Sumários, enunciados e cópia dos slides de acompanhamento das aulas

Avaliação Prática (trabalho em grupo):

- Aulas práticas laboratoriais com atividades de consolidação de conhecimentos
- Dois trabalhos de avaliação:
 - (TP1): trabalho prático com apresentação e demonstração de operacionalidade
 - (TP2): trabalho de projeto com entrega de relatório, apresentação e demonstração de operacionalidade;
- Avaliação individual em Exame Final

Classificação Final:

(20%) (TP1): trabalho prático

(30%) (TP2): trabalho de projeto obrigatório (nota ≥ 10)

(50%) Exame final em época de exames (nota ≥ 10)

O relatório terá um peso importante nos critérios de avaliação do TP2, nomeadamente na clareza da descrição e justificação da arquitetura desenvolvida

- O exame final terá uma duração de 2 Horas com a seguinte estrutura:

45 Minutos – **[Sem Consulta]** Questões sobre conceitos fundamentais e aspetos das tecnologias utilizadas;

75 Minutos – **[Com Consulta]** Questões práticas de aplicação dos conhecimentos adquiridos e de aspetos de implementação relacionados com as tecnologias utilizadas.

Nota: Entende-se por **[Com Consulta]** a possibilidade de consultar o material em papel (livros, cópias dos slides; exemplos, etc.) que individualmente, cada aluno leve para o exame, excluindo equipamentos eletrónicos.

Formato do Exame Época Especial

- A avaliação por exame especial é completamente independente do funcionamento da disciplina, visto que um aluno, que tenha esse direito, pode realizar o exame especial sem ter realizado qualquer frequência e/ou trabalhos práticos.
- Na época especial, pode prestar provas de exame final todo o aluno que, estando inscrito na disciplina, reúna as condições previstas em diploma legal, e nos casos dos pontos 3.5.1 e 3.5.2 das Normas de avaliação de conhecimentos do ISEL
- O exame de época especial tem uma componente teórica e uma componente prática que poderá ser diferente dos trabalhos práticos realizados durante o semestre

➤ Pré-requisitos:

- Programação orientada aos objetos com prática em linguagem Java
- Programação *multithreading* e controlo de concorrência
- Conhecimentos de protocolos de comunicação TCP/IP e HTTP
- Conhecimentos do sistema operativo Linux (*file system* e permissões rwx)

➤ Tecnologias e *tools*

- Plataforma Java (**JDK 11**)
- IntelliJ IDEA Community Edition 2021.X.X
- *Build System* (suportado pelo *IntelliJ*)
 - Maven (<https://maven.apache.org/>)
Maven in 5 minutes: <https://maven.apache.org/guides/getting-started/maven-in-five-minutes.html>
- *Google Cloud Platform Tools*

Computação Distribuída

- ✓ Um único computador
- ✓ Um único processador
- ✓ RAM escassa e cara
- ✓ Disco escasso e caro
- ✓ Comunicação em rede lenta ou inexistente
- ✓ Latência de segundos
- ✓ Poucos utilizadores concorrentes
- ✓ Dados de pequena dimensão

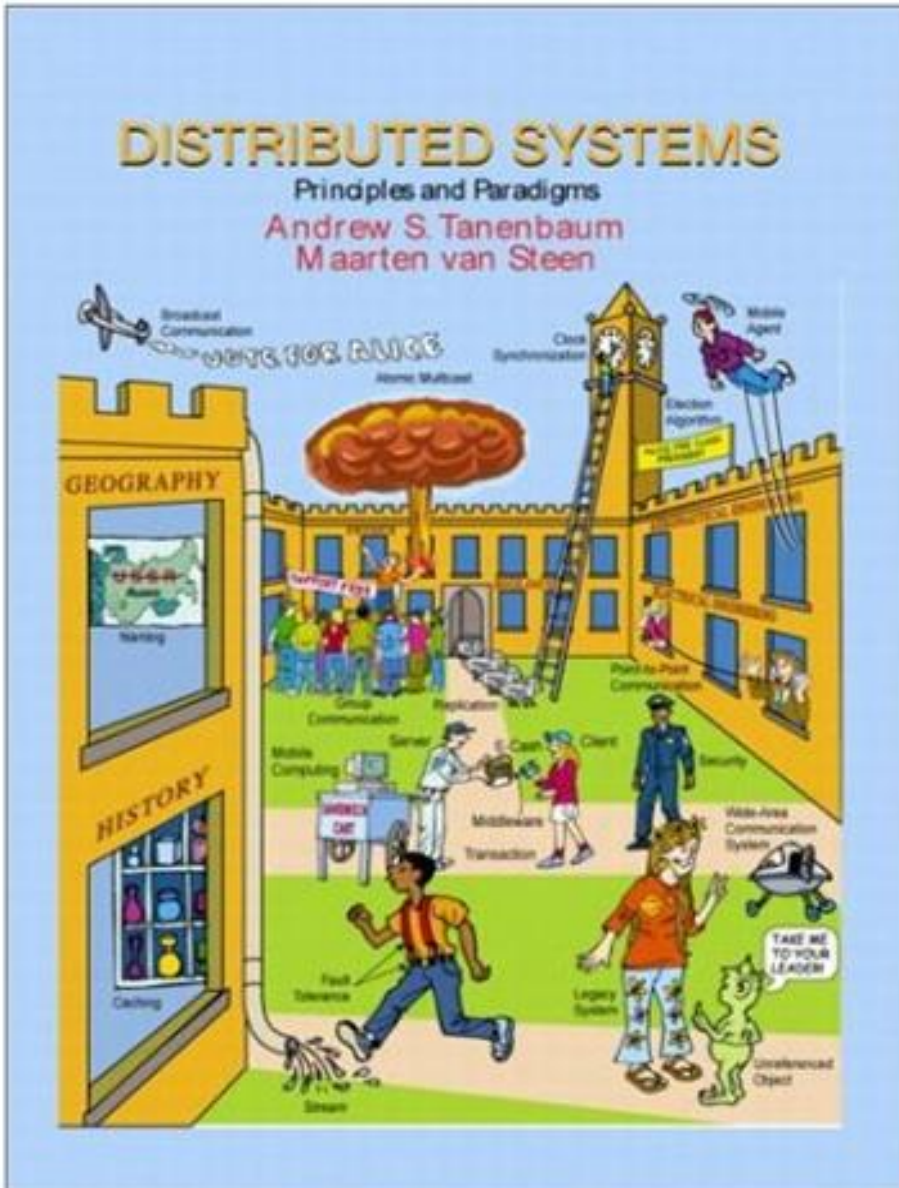
Passado



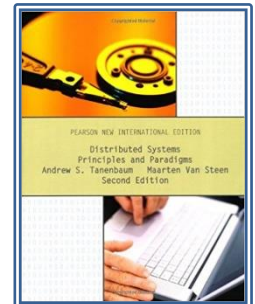
Presente

- ✓ Clusters de computadores
- ✓ Múltiplos processadores (*Multicore*)
- ✓ Virtualização com VMs e *Containers*
- ✓ RAM com grande dimensão e baixo preço
- ✓ Discos com grande dimensão e baixo preço
- ✓ Comunicação em rede de banda larga
- ✓ Latência de milissegundos
- ✓ Muitos (mesmo muitos) utilizadores concorrentes
- ✓ Dados de grande dimensão e estrutura (*BigData*)
- ✓ Recursos computacionais *on-demand* (*Clouds*)
- ✓ Larga Escala e Elasticidade

Sistemas distribuídos



- ✓ Clock synchronization
- ✓ Election Algorithm
- ✓ Broadcast/Multicast communication
- ✓ Group Communication
- ✓ Caching
- ✓ Replication
- ✓ Peer-to-Peer Communication
- ✓ Fault Tolerance
- ✓ Transactions
- ✓ Security
- ✓ Grid Computing
- ✓ Mobile Computing
- ✓ Cloud Computing



Distributed Systems, Principles and Paradigms,
Andrew S. Tanenbaum, Pearson Education, 2014

Computação Distribuída: *Buzzwords* !

Sockets TCP/IP

Client/Server applications

Distributed Objects

Java Remote Method Invocation

Remote Procedure Calls - gRPC

Web Services

Web applications

JSON

Protobuf

XML

SOAP

ReST (Representational State Transfer)

Model View Control (MVC)

RESTful Service

Web Sockets

Message Oriented Middleware

Monolithic

Web API

Publish/Subscribe

Microservices

Relational DB

Message Broker

Containers

Virtual machines

NoSQL

Stateless/Stateful

Kafka

Docker

Orchestration

Choreography

Cloud services

Cluster

Kubernetes

Serverless - FaaS

HPC – High Performance Computing