

# Infra-Estrutura de Software

## Apresentação da Disciplina

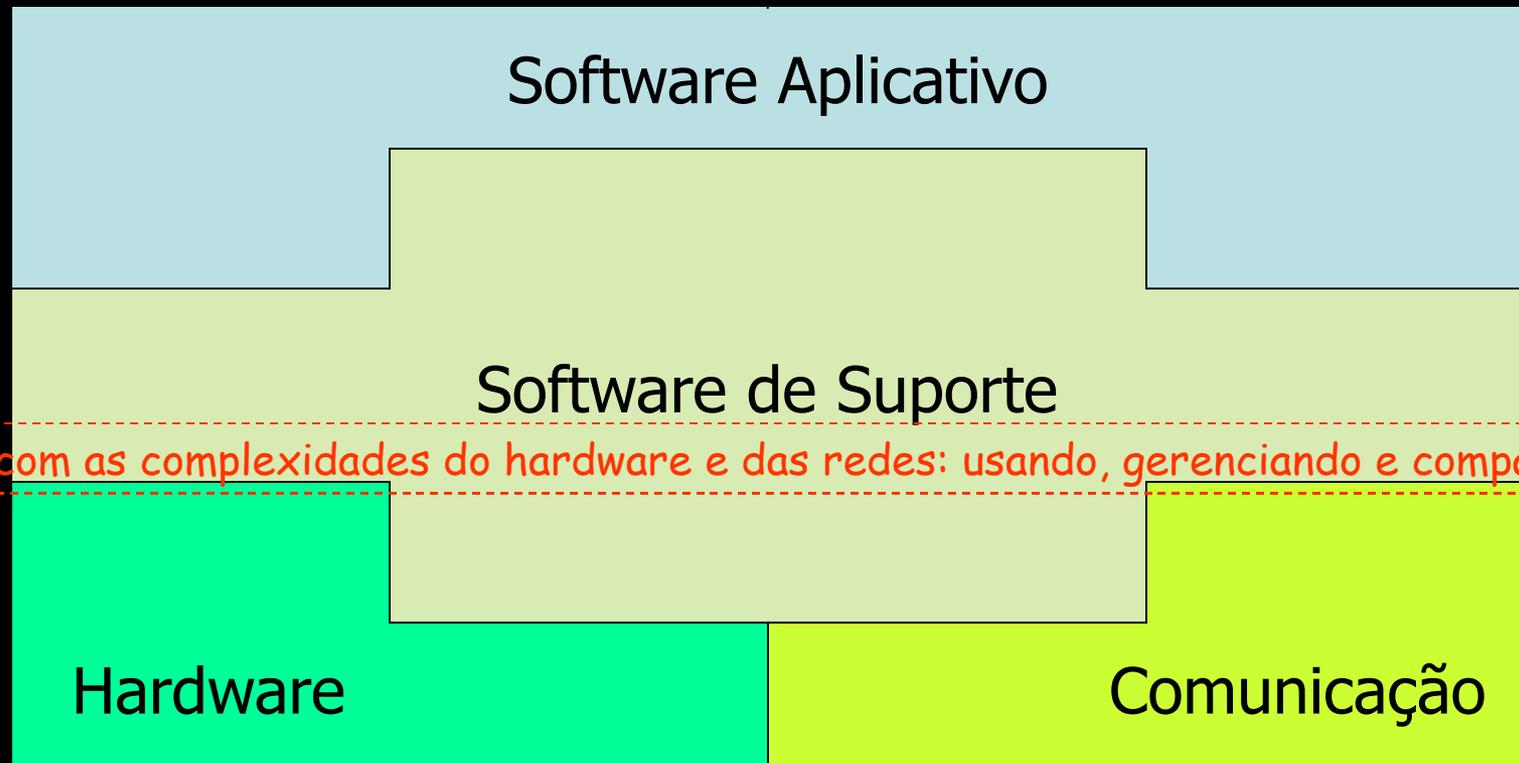
Fernando Castor

[castor@cin.ufpe.br](mailto:castor@cin.ufpe.br)

(com base em *slides* elaborados por Carlos Ferraz)

# Infra-Estruturas

Software+Hardware+Comunicação



- Existe aqui um programa (MS-PowerPoint) rodando
  - usando o **processador** da máquina,
- ...a **memória**,
- ...manipulando um **arquivo**,
- ...aparecendo na **tela**,
- ...recebendo comandos, via **teclado**

Como isso é possível?

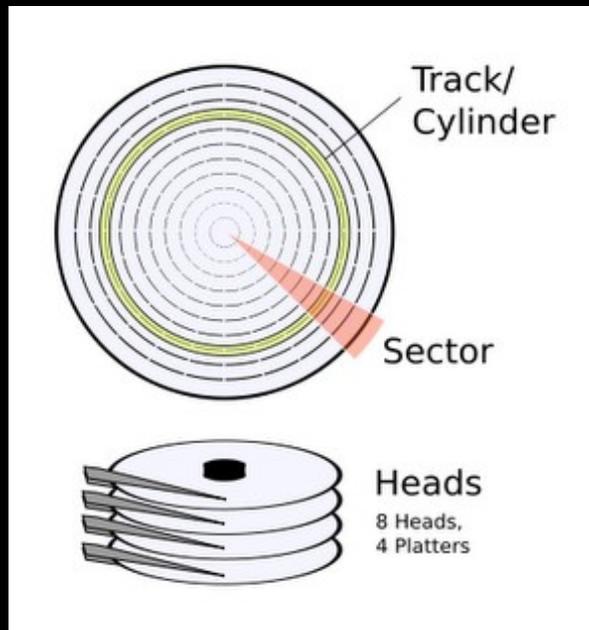
# Um sistema operacional...

- é um gerenciador de recursos de hardware
  - CPU/processador
  - Memória
  - Disco
  - Dispositivos de entrada/saída – teclado, tela, mouse etc.

- Gerência de processo
- Gerência de entrada/saída
- Gerência de disco/  
armazenamento – Sistema  
de Arquivos
- Gerência de memória

# Um sistema operacional...

- Também é um mecanismo de abstração dos dispositivos subjacentes
  - Ex.



Pin out	
1	39
2	40
Pin 1	Reset
Pin 2	Ground
Pin 3	Data 7
Pin 4	Data 8
Pin 5	Data 6
Pin 6	Data 9
Pin 7	Data 5
Pin 8	Data 10
Pin 9	Data 4
Pin 10	Data 11
Pin 11	Data 3
Pin 12	Data 12
Pin 13	Data 2
Pin 14	Data 13
Pin 15	Data 1
Pin 16	Data 14
Pin 17	Data 0
Pin 18	Data 15
Pin 19	Ground
Pin 20	Key or VCC_in
Pin 21	DDRQ
Pin 22	Ground
Pin 23	I/O write
Pin 24	Ground
Pin 25	I/O read

- E se o sistema for distribuído em rede?
  - Ex.: Web browser e servidor
- É preciso gerenciar também recursos de rede/comunicação

## Um middleware...

dá suporte a sistemas de software distribuídos

# Objetivos do curso

- Apresentar como funciona um computador típico, pela ótica da **infra-estrutura de software**
  - Sistema operacional
    - Conceitos
      - Ex. Concorrência
    - Funcionalidades
    - Prática
  - Middleware

# Ao final do curso você deverá ser capaz de

- **Explicar** o funcionamento de um SO
  - Dos pontos de vista de gerenciamento de recursos e mecanismo de abstração
- **Aplicar** vários dos conceitos discutidos, como processos, *threads*, interrupções e escalonamento
- **Usar** infra-estruturas existentes para computação distribuída

...E não deverá ser capaz de

- **Projetar** um novo sistema operacional
- **Implementar** um novo sistema operacional
- **Estender** um sistema operacional existente
- **<os mesmos verbos>** uma plataforma de middleware

...E não deverá ser capaz de

- **Projetar** um novo sistema operacional
- **Implementar** um novo sistema operacional
- **Estender** um sistema operacional existente
- **<os mesmos verbos>** uma plataforma de middleware
- Existem disciplinas mais apropriadas para isso:
  - IF709 -IMPLEMENTACAO SIST. OPERACIONAIS
  - IF749 -TOPICOS AVANC.SIST.DISTRIBUIDOS

# Metodologia

- Aulas expositivas
- Demonstrações de conceitos em laboratório

# Estrutura

- Parte I: Sistemas Operacionais
  - Suporte a sistemas de software convencionais
- Parte II: Middleware
  - Suporte a sistemas de software distribuídos

# Avaliação

- Provas (EE1 e EE2)
- Projeto integrado (EE3 = Parte I + Parte II)

Jogo:

- Por que?
- Mas esta não é uma disciplina de jogos!

Entregas:

- E1 = abertura do jogo em Assembly;
- E2 = jogo-partel;
- E3 = serviços de middleware (testados com clientes-falsos);
- E4 = jogo distribuído-parteII;

Nota do projeto = pesos das partes a definir

- Nota Final =  $(EE1 + EE2 + EE3) / 3$

•Haverá especificação de requisitos obrigatórios  
•Na 1ª. prova haverá uma questão, baseada no material da monitoria

# Material de Estudo

- Transparências das aulas
  - [www.cin.ufpe.br/~if677](http://www.cin.ufpe.br/~if677)
- Livros
  - Parte I: Sistemas Operacionais Modernos – 3ª Edição. A. Tanenbaum, 2008
  - Parte II:
    - Distributed Systems: Principles and Paradigms. Andrew Tanenbaum, Maarten van Steen. Prentice-Hall, 2002
    - Distributed Systems: Concepts and Design (3rd / 4th Edition). George Coulouris, Jean Dollimore, Tim Kindberg. Addison-Wesley, 2001 / 2005