

## **Sistemas de Tempo Real**

Jean-Marie Farines  
Joni da Silva Fraga  
Rômulo Silva de Oliveira

LCMI - Laboratório de Controle e Microinformática  
DAS - Departamento de Automação e Sistemas  
UFSC - Universidade Federal de Santa Catarina

## **Apresentação**

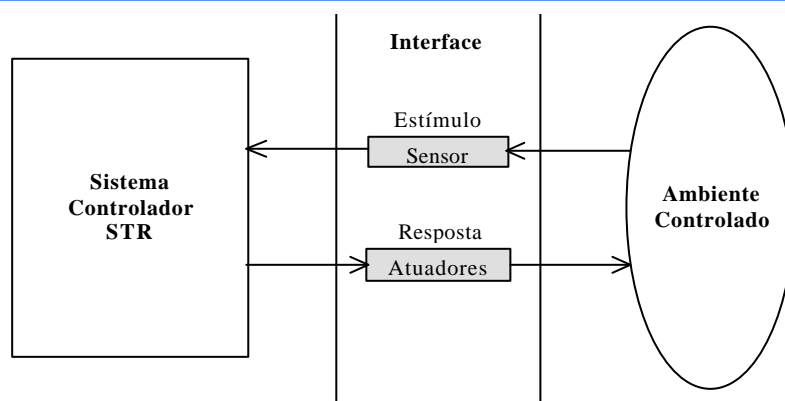
---

- Caracterização
- Mercados
- Concepções Erradas
- Conceitos Básicos
- Diversos Aspectos
- Organização do livro e do curso

## Caracterização

- **Sistemas computacionais de tempo real:**
  - Submetidos a requisitos de natureza temporal
  - Resultados devem estar corretos lógicamente e temporalmente
  - Requisitos definidos pelo ambiente físico
- **Aspectos temporais**
  - NÃO estão limitados a uma questão de maior ou menor desempenho
  - Estão diretamente associados com a funcionalidade
- **Sistemas em geral:**
  - “Fazer o trabalho usando o tempo necessário”
- **Sistemas de tempo real:**
  - “Fazer o trabalho usando o tempo disponível!”

## Caracterização



## **Mercados**

---

- **Telecomunicações**
  - Estabelecimento de conexões, videoconferência, groupware
- **Aeroespacial**
  - Automação em aeronaves, sondas espaciais
- **Defesa**
  - Radar, sonar, sistema guia em mísseis
- **Indústria**
  - Controle de processos, robôs, aquisição de dados
- **Financeiro**
  - Transações em bolsa, negociação automática
- **Entretenimento**
  - Vídeo games, vídeo sob demanda

## **Concepções Erradas**

---

- Tempo real significa execução rápida
- Computadores mais rápidos vão resolver todos os problemas
- Sistemas de tempo real são pequenos, escritos em assembly
- Sistemas de tempo real são formados apenas por
  - Tratadores de interrupção e
  - Device drivers
- Não existem problemas específicos da área de tempo real
- Sistemas de tempo real operam em ambientes estáticos

## Conceitos Básicos

---

- **Tarefa ( task )**
  - Segmento de código cuja execução possui atributo temporal próprio
  - Exemplo: método em OO, subrotina, trecho de um programa
- **Deadline**
  - Instante máximo desejado para a conclusão de uma tarefa
- **Tempo real crítico ( **hard real-time** )**
  - Falha temporal pode resultar em consequências catastróficas
  - Necessário garantir requisitos temporais em projeto
  - Exemplo: usina nuclear, indústria petroquímica, mísseis
- **Tempo real não crítico ( **soft real-time** )**
  - Requisito temporal descreve apenas comportamento desejado
  - Exemplo: multimídia

## Necessidade de Diferentes Abordagens

---

- Mercado para tempo real é amplo
- Sistemas de tempo real variam enormemente
  - Sistema de emergência em usina petroquímica
  - Controle de temperatura do freezer
  - Vídeo game
- Principais variações:
  - Crítico ou não crítico
  - Carga estática ou dinâmica
  - Importância associada com os deadlines
- Diferentes abordagens são necessárias

## Classificação das Abordagens

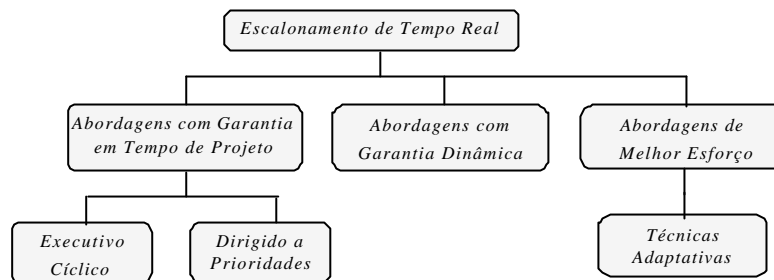


Figura 2.3: Abordagens de Escalonamento de Tempo Real

## Abordagens com Garantia em Projeto

- Oferece previsibilidade determinista
- Análise feita em projeto
  - Carga limitada e conhecida em projeto ( Hipótese de Carga )
  - Suposto um limite para faltas ( Hipótese de Faltas )
- Vantagens
  - Determina em projeto que todos os deadlines serão cumpridos
  - Necessário para aplicações críticas
  - Teoria serve de base para abordagens sem garantia
- Desvantagens
  - Necessário conhecer exatamente a carga
  - Necessário reservar recursos para o pior caso
  - Difícil determinar o pior caso em soluções off-the-shelf
  - Gera enorme subutilização de recursos

## **Abordagens com Garantia Dinâmica e Melhor Esforço**

---

- Não existe garantia que os deadlines serão cumpridos
- Sempre que uma tarefa é ativada ocorre uma análise da sua escalonabilidade
- Capaz de fornecer análise probabilista
  - Simulação, teoria das filas de tempo real, etc
- Algumas abordagens oferecem Garantia Dinâmica
  - Garante o deadline (ou não) no início da ativação

## **Diversas Áreas**

---

- Sistemas de Tempo Real
  - Escalonamento recebe o maior destaque
  - Existem vários aspectos envolvidos
- Sistemas de Tempo Real possuem diferentes aspectos
  - Escalonamento
  - Modelos e linguagens de programação
  - Protocolos de comunicação
  - Arquitetura de computadores
  - Metodologias de desenvolvimento
  - etc
- Impossível tratar todos os aspectos dentro do escopo de um livro da Escola de Computação

## **Nosso Curso/Livro**

---

- Livro dividido em 6 capítulos
  - 1) Introdução (segunda & terça)
  - 2) Escalonamento (terça & quarta)
  - 3) Suportes (quinta)
  - 4) Abordagem Síncrona (sexta)
  - 5) Exemplos (sábado a noite)
  - 6) Tendências

## **Evolução Rápida**

---

- Área em rápida evolução
- Teoria de escalonamento desenvolvida nos anos 90
- Prática começa a mudar em função da teoria
- Existe muito a ser feito
  
- Evolução acontece em ondas
  - Pesquisa teórica ( teoremas, modelos abstratos )
  - Desenvolvimento tecnológico ( algoritmos, modelos realistas )
  - Empacotamento da tecnologia ( compilador, sistema operacional )
  - Construção de aplicações

## Internet

---

- <http://www.lcmi.ufsc.br>
- <http://www.lcmi.ufsc.br/gtr/livro/principal.htm>
  
- [romulo@lcmi.ufsc.br](mailto:romulo@lcmi.ufsc.br)
  
- Home Page da IEEE Computer Society, Technical Committee on Real-Time Systems (IEEE-CS TC-RTS):  
  
<http://www.cs.bu.edu/pub/ieee-rts>