



***“O Software ultrapassou o Hardware  
como chave para o sucesso de  
muitos sistemas baseados em  
computador” (Pressman)***

***Mas, será que Software é sinônimo  
de Sucesso??***



Prof. Dr. Alexandre Cardoso



**Crise do Software: Denver International Airport**



Custo do projeto: US\$ 4.9 bilhões

- 100 mil passageiros por dia
- 2x o de Manhattan
- 1.200 vôos
- 53 milhas quadradas
- 94 portões de embarque e desembarque
- 6 pistas de pouso / decolagem



Prof. Dr. Alexandre Cardoso



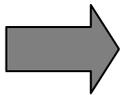
## Crise do Software: Denver International Airport

Erros no sistema automático de transporte de bagagens (*misloaded, misrouted, jammed*):

**Sistema deveria suportar:**

- ▶ 21 milhas de trilhas;
- ▶ 4000 rotas de carros teleguiados;
- ▶ 5000 olhos eletrônicos;
- ▶ 400 receptores de rádio - intercomunicação;
- ▶ 100 computadores conectados entre si;

**As consequências:**



- Atraso na abertura do aeroporto com custo total estimado em US\$360 Milhões
- 86 milhões para consertar o sistema



Prof. Dr. Alexandre Cardoso



## Crise do Software: Foguete Ariane V



Projeto da Agência Espacial Européia que custou:

10 anos.

US\$ 8 Bilhões.

*Capacidade:* 6 toneladas.

*Garante:* supremacia européia no espaço.

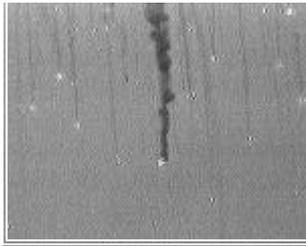
Vôo inaugural em 04/06/96:



Prof. Dr. Alexandre Cardoso



## Crise do Software: Foguete Ariane V



### Resultados

- Explosão 40 segundos após a decolagem.
- Destruição do foguete e carga avaliada em US\$ 500 milhões.

Fato: o veículo detonou suas cargas explosivas de autodestruição e explodiu no ar. Por que?

Porque ele estava se quebrando devido às forças aerodinâmicas.

Mas por que?

O foguete tinha perdido o controle de direção (atitude). Causa disso?

Os computadores principal e back-up deram *shut-down* ao mesmo tempo.



Prof. Dr. Alexandre Cardoso



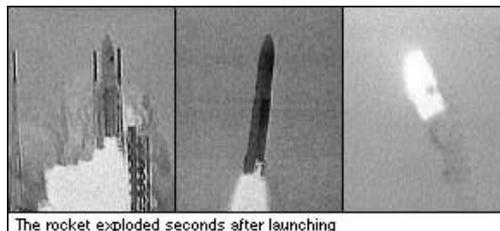
## Crise do Software: Foguete Ariane V

Por que o Shut-down? Ocorrerá um *run time error* (out of range, overflow, ou outro) e ambos computadores se desligaram. De onde veio este erro?

Um programa que convertia um valor em ponto flutuante para um inteiro de 16 bits recebeu como entrada um valor que estava fora da faixa permitida.

**Mas, por que???**

*O resultado desta conversão não era mais necessário após a decolagem...*



The rocket exploded seconds after launching



Prof. Dr. Alexandre Cardoso



## Crise do Software: O Caso FAA

FAA: Federal Aviation Administration - USA

O projeto:

- um novo sistema de controle de tráfego aéreo chamado AAS;
- programa arrojada: milhões de linhas de código distribuídas entre centenas de computadores com HW funcionando em tempo real;
- Contratada: *IBM*.



Prof. Dr. Alexandre Cardoso



## O Caso FAA

Custo Estimado:

- US\$ 500 por linha de código (5x a média de mercado).

Resultados:

- FAA pagando entre \$700 e \$900 por linha de código;
- FAA cancelou duas das quatro partes do projeto
- atraso na entrega
- o sistema (cheio de Bugs) está em análise pela *Carnegie Mellon* e *MIT*



Prof. Dr. Alexandre Cardoso



Há uma luz no  
fim do túnel???



Prof. Dr. Alexandre Cardoso



## Produto e Processo

- ▶ Ambos são aspectos da Engenharia de Software
- ▶ Ênfase: do produto para o processo;
- ▶ Necessidade: entregar produtos de software de qualidade para os clientes, desenvolvidos por processos consistentes, bem gerenciados e com controle efetivo de custos.

Prof. Dr. Alexandre Cardoso



## Produto e Processo

- ▶ Produto:
  - Qualidade
  - Atendimento à expectativas do cliente
  
- ▶ Processo:
  - Consistente
  - Bem gerenciado
  - Custo e tempo controlados



## Produto - Análise Inicial

### **Software:**

- uma força propulsora!!
- criar software é uma tarefa agonizante!
- é um fator de diferenciação!
- possui capacidades diferenciadas, como ser amigável ao ser humano.

### **O DESAFIO:**

- Aprimorar a Qualidade do Software que produzimos.





## Produto – uma análise inicial

### A Discussão:

- Como prover melhorias?
- Quem será agente das melhorias: o programador, as ferramentas, o gerente?



## A importância do Software

**Fato:** Durante as 3 primeiras décadas da era do computador, o principal desafio era desenvolver um **HARDWARE** de baixo custo e alto desempenho.

### Compõem o cenário atual:

- Cinco décadas de evolução dos computadores;
- Hardware de grande desempenho;
- Hardware de menor custo;
- Hardware com grandes e rápidas modificações...

**Proposição:** hoje o desafio é melhorar a qualidade (e reduzir os custos) das soluções baseadas em **SOFTWARE!**





## Eras do Software



### Primeira Era: 1950/1960

- software: uma reflexão posterior
- desenvolvimento de software sem administração
- orientação batch (programas em lote)
- hardware orientado a uma (ou poucas) aplicações
- software destinado a uma aplicação bem específica



Prof. Dr. Alexandre Cardoso



## Eras do Software

### Segunda Era: 1960/1970

- multiprogramação
- sistemas multi-usuários
- aparecimentos de sistemas de tempo real
- primeira geração de banco de dados
- advento das “Software Houses”
- software voltado à ampla distribuição
- horizonte incerto: milhares de linhas de programas/ instruções teriam de ser corrigidas (ou alteradas)
- manutenção dispendiosa e cara



Prof. Dr. Alexandre Cardoso



## Eras do Software

### Terceira Era: 1970/1985

- sistemas distribuídos: aumento de complexidade e uso de muitos computadores
- redes locais e globais
- demanda por acesso instantâneo
- poderosas estações de trabalho (*workstations*)
- computador pessoal (PC) como catalisador de crescimento de muitas empresas



Prof. Dr. Alexandre Cardoso



## Eras do Software

### Quarta Era: 1985/...

- tecnologias orientadas a objetos: na análise, no projeto, na criação de bancos de objetos
- sistemas especialistas e inteligência artificial
- redes neurais
- computação paralela
- linguagens de programação abertas
- sistemas para a Internet
- usuário desenvolvedor
- softwares sofisticados e com múltiplas funções



Prof. Dr. Alexandre Cardoso



## Análise do Histórico

### Início:

- programação era uma forma de arte
- poucos métodos formais eram aplicados
- esquema baseado em tentativa e erro
- mundo indisciplinado

### Atualmente:

- software é o item de maior custo.
- por que a demora na conclusão, se são aplicados métodos e ferramentas?
- por que há tantos erros?
- por que é difícil medir nossos erros?
- necessidade de atualização das aplicações
- empresas e outsourcing



Prof. Dr. Alexandre Cardoso



## O Software (produto)

### **Software:**

- 1º- instruções (programas de computador) que, quando executadas, produzem a função e o desempenho desejados;
- 2º- estruturas de dados que permitem a manipulação das informações;
- 3º - documentos que descrevem a operação e uso dos programas.



Prof. Dr. Alexandre Cardoso



## Características do Software

***O Software é desenvolvido ou projetado por engenharia, não manufaturado no sentido clássico:***

1. Custos são concentrados no trabalho de engenharia.
2. Projetos não podem ser geridos como projetos de manufatura.
3. “Fábrica de Software!”



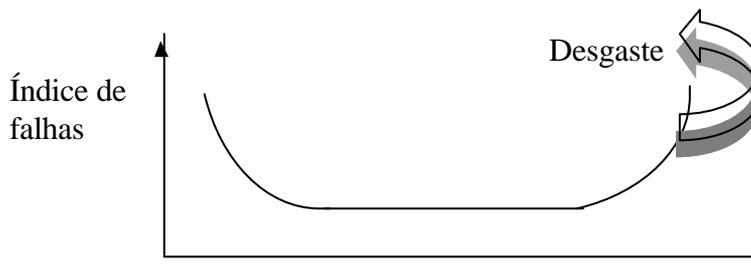
Prof. Dr. Alexandre Cardoso



## Características do Software

***Software não desgasta!***

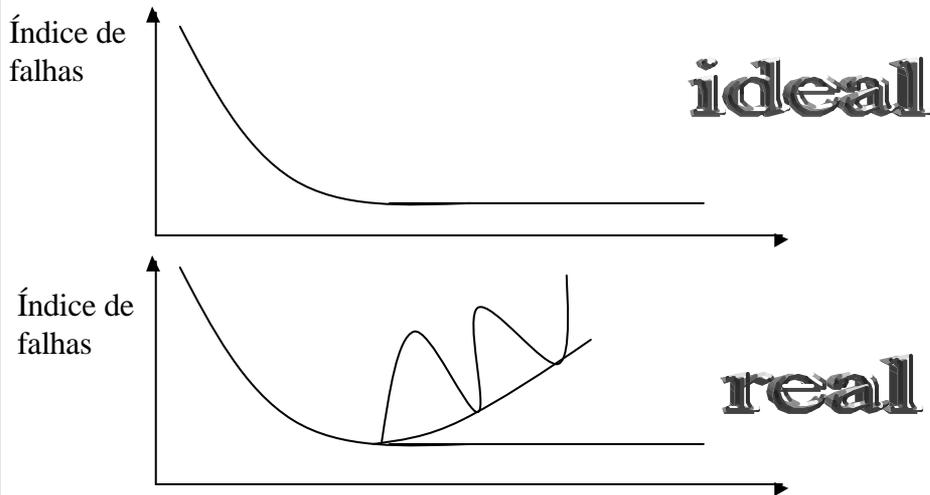
- Software não é sensível aos problemas ambientais que fazem com que o hardware se desgaste.
- Toda falha indica erro de projeto ou implementação: manutenção do SW é mais complicada que a do HW.



Prof. Dr. Alexandre Cardoso



## Curvas do Software



Prof. Dr. Alexandre Cardoso



## Características do Software

***A maioria dos softwares é feita sob medida e não montada a partir de componentes existentes.***

- Bem diferente do Hardware.
- A situação esta mudando:
  - a) Orientação a objetos.
  - b) Reusabilidade é o “Santo Graal”(diminui custos e melhora projetos).



Prof. Dr. Alexandre Cardoso



## Componentes do Software

### Duas formas básicas:

1. Componentes não executáveis em máquinas
2. Componentes executáveis em máquinas

### Linguagens:

1. de máquina
2. de alto nível
3. especializadas



Prof. Dr. Alexandre Cardoso



## Aplicações de Software

- ✓ Software Básico
- ✓ Software de Tempo Real
- ✓ Software Comercial
- ✓ Software Científico ou de Engenharia
- ✓ Software Embutido
- ✓ Software de Computador Pessoal
- ✓ Software de Inteligência Artificial



Prof. Dr. Alexandre Cardoso



# Mitos do Software

## I. Mitos Administrativos

1º MITO: Já possuímos um manual com padrões e procedimentos para a construção de um Software  
- Isto não é suficiente?

REALIDADE:

- será que o manual é usado? É conhecido pela equipe? É completo?



Prof. Dr. Alexandre Cardoso



# Mitos Administrativos

2º MITO: Temos boas ferramentas, já que possuímos a última geração de computadores!

REALIDADE:

- Só computadores não são suficientes - Ferramentas de Engenharia de Software acabam sendo mais importantes.



Prof. Dr. Alexandre Cardoso



# Mitos Administrativos

3º MITO: Se estamos atrasados, vamos contratar mais programadores

REALIDADE:

- O desenvolvimento de Software não é um processo manufaturado, assim, mais programadores acabam por atrasar ainda mais o processo.

*“Nove mulheres não conseguem fazer um filho em um mês”*



Prof. Dr. Alexandre Cardoso



# Mitos do Software

## II. Mitos do Cliente

1º MITO: Uma declaração geral de objetivos é suficiente para se começar a escrever programas

REALIDADE:

-Má definição inicial é a maior causa de fracasso do SW

- há necessidade de cuidadosa comunicação entre desenvolvedor e cliente.



Prof. Dr. Alexandre Cardoso



## Mitos do Cliente

2º MITO: Os requisitos do Software modificam-se continuamente, mas, as mudanças podem facilmente serem acomodadas

REALIDADE:

- O impacto da mudança varia de acordo com o tempo em que ela é introduzida;
- Uma séria atenção na definição inicial possibilita uma boa acomodação de solicitações de mudança;
- Mudanças durante o projeto do Software impacta fortemente os custos, já que recursos foram comprometidos.



Prof. Dr. Alexandre Cardoso



## Mitos do Software

### III. Mitos do Profissional

1º MITO: O trabalho se resume a escrever e colocar um programa em funcionamento

REALIDADE:

- Quanto mais cedo se inicia o código, mais tempo é despendido no seu término: de 50 a 70% do esforço é despendido depois da primeira entrega.



Prof. Dr. Alexandre Cardoso



## Mitos do Profissional

2º MITO: Enquanto o programa não estiver funcionando,  
não há como avaliar sua qualidade

REALIDADE:

- A revisão técnica formal é um bom mecanismo de filtro de qualidade



Prof. Dr. Alexandre Cardoso

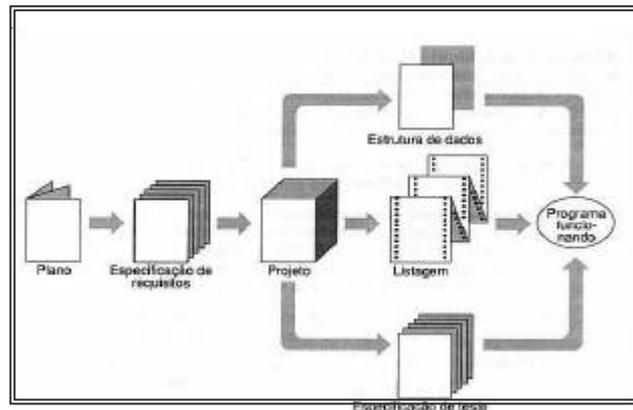


## Mitos do Profissional

3º MITO: Só deve ser entregue o programa em funcionamento

REALIDADE:

- O programa é uma parte de uma configuração de Software



Prof. Dr. Alexandre Cardoso



# Engenharia de Software

*Engenharia de Software: é o estabelecimento e uso de sólidos princípios de Engenharia para que se possa obter, economicamente, um software que seja confiável que funcione eficientemente em máquinas reais*

Componentes { métodos: como fazer;  
ferramentas: como os CASE;  
procedimentos: seqüência de aplicação de métodos

Há diversos paradigmas (conjuntos de etapas que envolvem métodos, ferramentas e procedimentos) relacionados com Engenharia de Software.

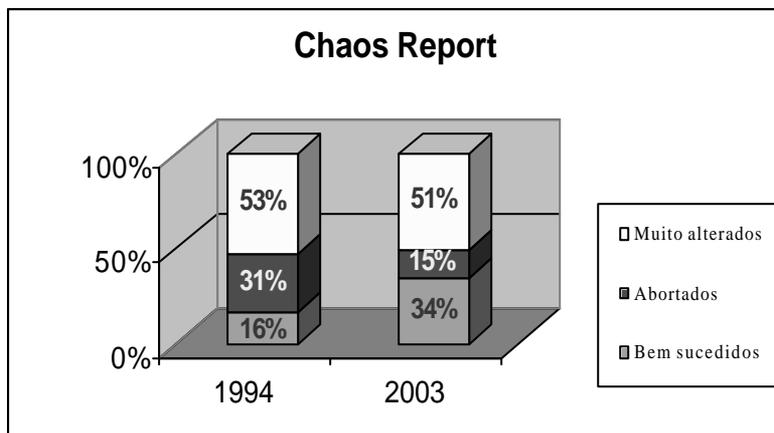


Prof. Dr. Alexandre Cardoso



## O Processo - cenário

**Pesquisas realizadas em 1994 e 2003 pelo Standish Group sobre projetos de software:**



Prof. Dr. Alexandre Cardoso



## O Processo - discussão

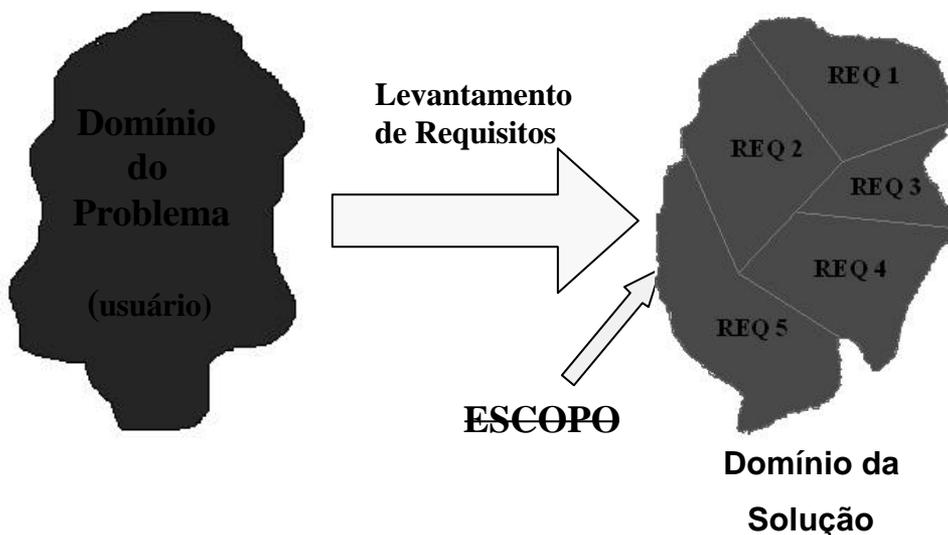
- ▶ Alguns motivos de insucesso:
  - Dificuldade em captar os requisitos dos usuários;
  - Excesso de alterações na especificação do escopo;
  - Dificuldade em estimar prazos e recursos.



Prof. Dr. Alexandre Cardoso



## Os Domínios – Problema e Solução



Prof. Dr. Alexandre Cardoso