



ENGENHARIA DE SOFTWARE

Histórico

Produtos de software

Tipos de aplicações de software

Mitos do software

Kele Teixeira Belloze

kelebelloze@gmail.com

HISTÓRICO (ERA PRÉ-ES)

- **1940s:** Primeiro computador eletrônico de uso geral – ENIAC
 - Custo estimado de US\$ 500.000,00
 - Início da programação de computadores
- **1950s:** Primeiros compiladores e interpretadores
- **1960s:** Primeiro grande software relatado na literatura – OS/360
 - Mais de 1000 desenvolvedores
 - Custo estimado de US\$ 50.000.000,00 por ano
- **1968:** Crise do software – nasce a Engenharia de Software



HISTÓRICO (ERA PÓS-ES)

- 1970s:
 - Ciclo de vida cascata
 - Desenvolvimento estruturado
- 1980s:
 - Ciclo de vida espiral
 - Desenvolvimento orientado a objetos
- 1990s:
 - Processos
 - Modelagem



HISTÓRICO (ERA PÓS-ES)

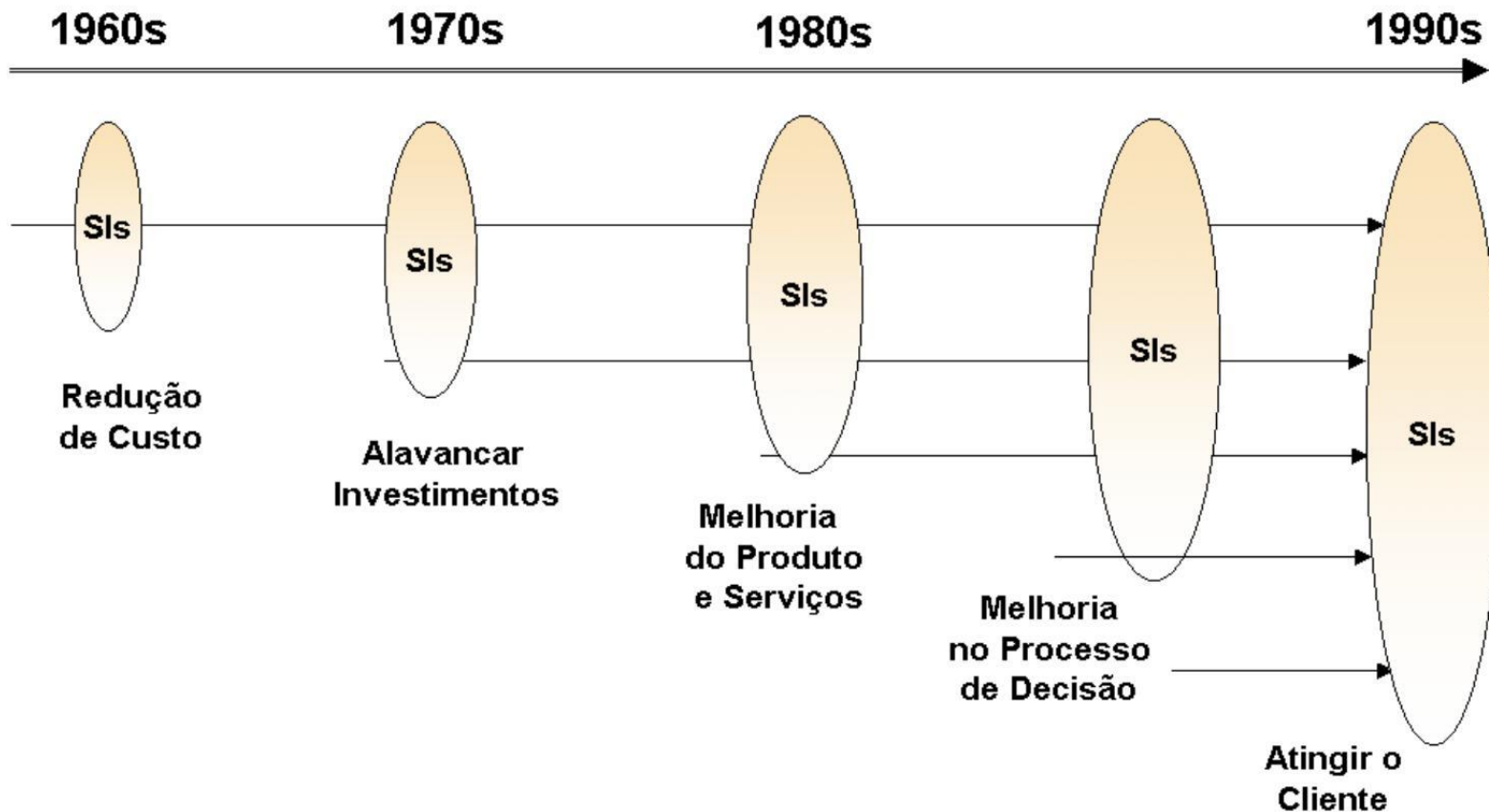
○ Século XXI:

- Desenvolvimento baseado na Web
- OO e componentes
- Qualidade
- Gestão do conhecimento
- Serviços web
- Web semântica
- Agentes
- Arquitetura orientada a serviço (SOA)
- Aspectos
-



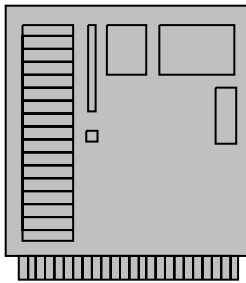
A IMPORTÂNCIA DO SOFTWARE

- A complexidade do software vem crescendo ao longo do tempo



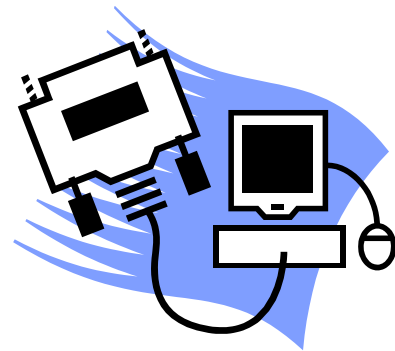
A IMPORTÂNCIA DO SOFTWARE

- As três primeiras décadas da era do computador:
 - Principal desafio: desenvolver um hardware que reduzisse o custo de processamento e o armazenamento de dados
 - Software na década de 60 era um único programa executado no mainframe.
 - Realizava uma única função bem definida



A IMPORTÂNCIA DO SOFTWARE

- Ao longo da década de 1980:
 - Avanços na microeletrônica resultaram em maior poder de computação a um custo cada vez mais baixo
 - O software atingia o processamento distribuído
 - Softwares cada vez mais complexos realizavam comunicações entre módulos de programa (Cliente/Servidor)



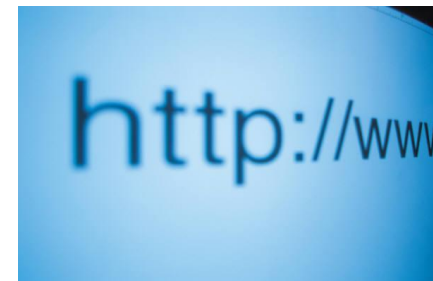
A IMPORTÂNCIA DO SOFTWARE

- Década de 1990 em diante
 - Principal desafio: melhorar a qualidade (e reduzir o custo) de soluções baseadas em computador



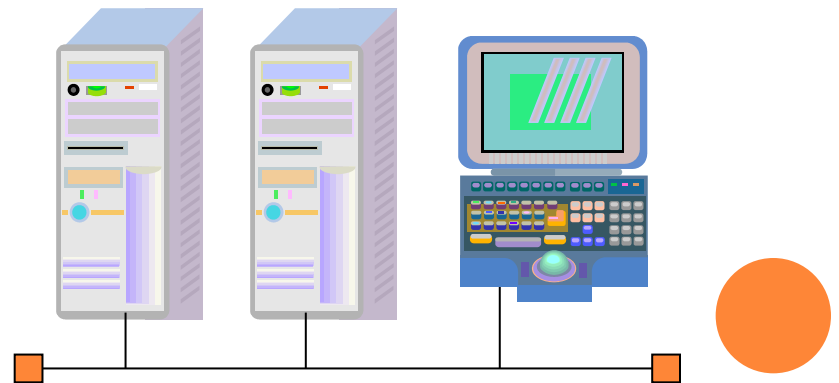
A IMPORTÂNCIA DO SOFTWARE

- No fim da década de 90 a Internet levou as operações de negócio para o mundo virtual de forma ampla
- Software cada vez mais complexos e vitais para as organizações (ERP, CRM, SCM, etc...)



A IMPORTÂNCIA DO SOFTWARE

- Hoje a arquitetura dos softwares é em três camadas (cliente, aplicação e servidor)
- Uso de componentes que se comunicam entre si e são reutilizáveis (reuso de software)





2016-1

SOFTWARE:

CONCEITOS E APLICAÇÕES

CONCEITOS



- O que é Software?
 - Instruções (programas de computador) que, quando executadas, produzem a função e o desempenho desejados
 - Estruturas de dados que possibilitam que os programas manipulem adequadamente a informação
 - Documentos que descrevem a operação e o uso dos programas
 -



APLICAÇÕES DO SOFTWARE

- O software pode ser aplicado a qualquer situação em que um conjunto previamente especificado de passos procedimentais (algoritmo) tiver sido definido



CARACTERÍSTICAS DO SOFTWARE

- Software é desenvolvido ou projetado por engenharia, não manufaturado no sentido clássico
 - os custos do software estão centrados no trabalho de engenharia, no processo
- Software não se “desgasta”
 - o software se “deteriora”
 - a manutenção do software é complexa
- A maioria dos software é feita sob medida
 - o crescente uso da OO e técnicas de reutilização tem contribuído para o reuso de componentes




PROJETOS DE SOFTWARE

- O desenvolvimento de software é ainda imprevisível
 - Somente 10% dos projetos de software são entregues com sucesso dentro das estimativas de orçamento e custo
 - O nível de software jogado fora e que tem necessidade de re-trabalho é um indicativo de processo imaturo
- Problemas
 - Falta de dados históricos sobre o processo de desenvolvimento
 - Insatisfação de clientes
 - Qualidade suspeita do software
 - Manutenção difícil



TIPOS DE APLICAÇÕES

1. Aplicações stand-alone: são aplicações executadas em um computador local, tal como um PC. Incluem toda a funcionalidade necessária e não precisam estar conectadas a uma rede.
 2. Aplicações interativas baseadas em transações: são aplicações executadas em um computador remoto e são acessadas pelos usuários a partir dos seus próprios PCs ou terminais. Essas incluem aplicações web tais como para e-commerce.
 3. Sistemas de controle embutidos: são sistemas de software que controlam e gerenciam dispositivos de hardware. Por exemplo, funções digitais em um automóvel como controle de combustível.
 4. Sistemas de entretenimento: são, principalmente, para uso pessoal e se destinam a entreter o usuário.
- 

TIPOS DE APLICAÇÕES

5. Sistemas de processamento de lotes: são sistemas corporativos projetados para processar dados em grandes lotes. Eles processam um grande número de entradas individuais para criar saídas correspondentes.
6. Sistemas de modelagem e simulação: são desenvolvidos por cientistas e engenheiros para modelar processos físicos ou situações, que incluem muitos, objetos separados que interagem entre si.
7. Sistemas de coleta de dados: são sistemas que coletam dados do seu ambiente usando um conjunto de sensores e envia esses dados para outros sistemas, para processamento.
8. Sistemas de sistemas: são sistemas compostos por uma série de outros sistemas de software. Por exemplo, componentes de sistemas operacionais, softwares de rede, etc.



PRODUTOS DE SOFTWARE

○ Produtos genéricos

- Sistemas autônomos que são comercializados e vendidos a qualquer cliente que deseje comprá-los.
- **Exemplos:** softwares para PC tais como programas gráficos, ferramentas de gerenciamento de projetos; software CAD; software para propósitos específicos, tais como sistemas de registros odontológicos, controle de estoque, etc.

○ Produtos sob encomenda

- O software que é encomendado por um cliente específico para atender suas próprias necessidades.
- **Exemplos:** sistemas de controle integrado, software de controle de tráfego aéreo, sistemas de monitoramento de tráfego.



ESPECIFICAÇÃO DOS PRODUTOS

○ Produtos genéricos

- A especificação do que o software deve fazer é de propriedade do desenvolvedor de software e as decisões sobre as mudanças de software são feitas pelo desenvolvedor.

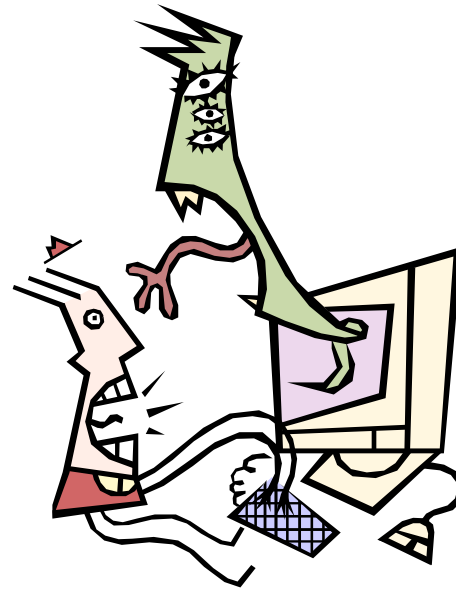
○ Produtos sob encomenda

- A especificação do que o software deve fazer é propriedade do cliente para o software e eles tomam decisões sobre as mudanças necessárias no software.



MITOS DO SOFTWARE

- Antigas atitudes e hábitos que são difíceis de serem modificados



MITOS ADMINISTRATIVOS

○ Mito:

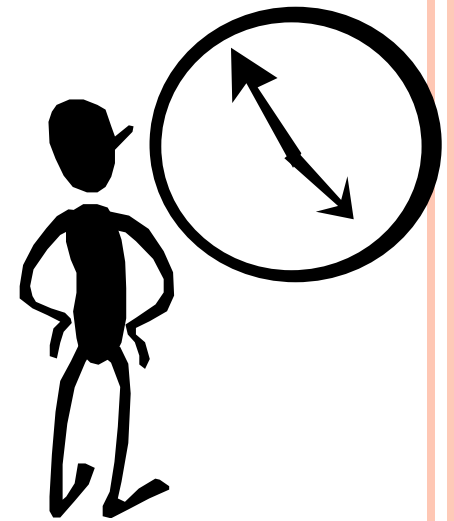
- Temos um manual repleto de padrões e procedimentos para a construção de software. Isso não oferecerá ao meu pessoal tudo o que eles precisam saber?

○ Realidade:

- O manual pode existir, mas ele será usado? Os profissionais de software tem conhecimento de sua existência? Ele reflete a moderna prática de desenvolvimentos de software? É completo?



MITOS ADMINISTRATIVOS



○ Mito:

- Se nós estamos atrasados nos prazos, podemos adicionar mais programadores e tirar o atraso

○ Realidade:

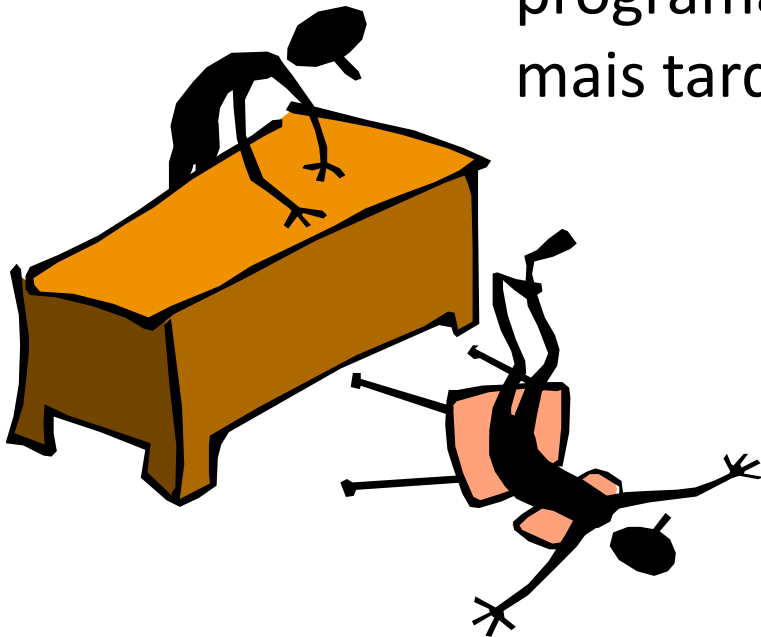
- O desenvolvimento de software não é mecânico como a manufatura. Acrescentar pessoas em projetos atrasados pode o tornar ainda mais atrasado
 - Quando novas pessoas são acrescentadas, as pessoas que estavam trabalhando devem gastar tempo educando os recém-chegados, o que reduz o tempo despendido num esforço de desenvolvimento produtivo.
 - Pessoas podem ser acrescentadas mas de forma planejada e coordenada.



MITOS DO CLIENTE

○ Mito:

- Uma declaração geral dos objetivos é suficiente para se começar a escrever programas – podemos preencher o detalhes mais tarde.



MITOS DO CLIENTE

○ Realidade:

- Uma definição inicial ruim é a principal causa de fracasso dos esforços de desenvolvimento de software.
- Uma definição formal e detalhada do domínio da informação, função, interfaces, validação,..., é fundamental.
- É necessária uma cuidadosa comunicação inicial entre o cliente e o desenvolvedor



MITOS DO CLIENTE

○ Mito:

- Os requisitos de projeto modificam-se continuamente, mas as mudanças podem ser facilmente acomodados pois o software é flexível



MITOS DO CLIENTE

○ Realidade:

- Os requisitos de software se modificam sim, mas o impacto de mudança varia de acordo com o tempo em que ela é introduzida
- Se uma séria atenção for dada à definição inicial, os primeiros pedidos de mudança podem ser acomodados facilmente. O cliente pode rever as exigências e recomendar modificações sem causar grande impacto



MITOS DO CLIENTE

○ Realidade (cont.)

- Quando são exigidas mudanças durante o projeto, o impacto sobre os custos eleva-se rapidamente. Recursos foram comprometidos e uma estrutura de projeto foi estabelecida. A mudança pode causar sublevações que exijam recursos adicionais e grandes modificações, isto é, custo adicional
- Mudanças na função, desempenho, interfaces, ..., durante a implementação (código e teste) exercem forte impacto sobre o custo



MITOS DO PROFISSIONAL

○ Mito:

- Assim que escrevermos o programa e o colocarmos em funcionamento, nosso trabalho estará completo.

○ Realidade:

- Alguns dados indicam que entre 50 e 70% de todo o esforço gasto num programa serão despendidos depois que ele for entregue pela primeira vez ao cliente



MITOS DO PROFISSIONAL

○ Mito:

- Enquanto não tiver o programa “funcionando”, eu não terei nenhuma maneira de avaliar sua qualidade



MITOS DO PROFISSIONAL

○ Realidade:

- Um dos mecanismos mais efetivos de garantia de qualidade pode ser aplicado desde o começo de um projeto – a revisão técnica formal, que tem sido considerada mais eficiente do que a realização de testes para a descoberta de certas classes de defeitos de software



MITOS DO PROFISSIONAL

○ Mito:

- A única coisa a ser entregue em um projeto bem sucedido é o programa funcionando



MITOS DO PROFISSIONAL

○ Realidade:

- Um programa funcionando é somente uma parte de uma configuração de software que inclui vários elementos: plano, especificação de requisitos, projeto, estrutura de dados, teste, ...
- A documentação forma os alicerces para um desenvolvimento bem sucedido e o que é mais importante, fornece um guia para a tarefa de manutenção de software.



PONTOS IMPORTANTES

- A engenharia de software é uma disciplina da engenharia que se preocupa com todos os aspectos da produção de software.
- Atributos essenciais do produto de software são a manutenibilidade, confiança, proteção, eficiência e aceitabilidade.
- As atividades de alto nível de especificação, desenvolvimento, validação e evolução fazem parte de todos os processos de software.
- As ideias fundamentais da engenharia de software são universalmente aplicáveis a todos os tipos de desenvolvimento do sistema.
- Existem muitos tipos diferentes de sistemas e cada um requer ferramentas de engenharia de software e técnicas apropriadas para o seu desenvolvimento.

