

Engenharia de Software

Visão Geral

Profa. Ellen Francine
francine@icmc.usp.br

ICMC/USP



Conteúdo

TÓPICOS

- Software e sua importância
- Aplicações de Software
- Mitos de Software
- Processo de Software
- Modelos de Processo de Software (fases genéricas)

Algumas Dúvidas...

Two horizontal bars are positioned below the title. The top bar is a solid blue line. The bottom bar is a thicker, semi-transparent blue line with a gradient, appearing slightly darker and more textured.

- Como são desenvolvidos os softwares?
- Estamos sendo bem sucedidos nos softwares que construímos?
- Quão difícil é construir software?

O Veículo Espacial ARIANE 5




- Em 1996, o veículo espacial Ariane 5 saiu do curso e explodiu segundos após o seu lançamento.
- Levou uma década de desenvolvimento e custou 7 bilhões de dólares.

O Veículo Espacial ARIANE 5



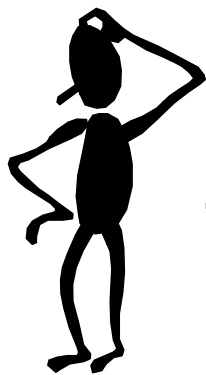
■ Motivo do acidente:

- Os testes em componentes reutilizados do veículo Ariane 4 foram a causa do acidente.
- Erro de software no cálculo da velocidade horizontal do foguete.
 - A variável que armazenava este valor tinha 64 bits (floating point) e foi erroneamente modificada para 16 bits (signed integer).
 - O valor era maior que 32.767 (maior inteiro), gerando uma falha de conversão!!!



Um Sistema Baseado em Computador (SBC) é um dos subsistemas do sistema empresa.


Qual é a finalidade de um SBC?



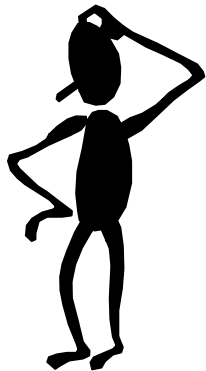
Sistema Baseado em Computador

- Conjunto de **elementos** organizados para executar certo método, procedimento ou controle ao processar informações.

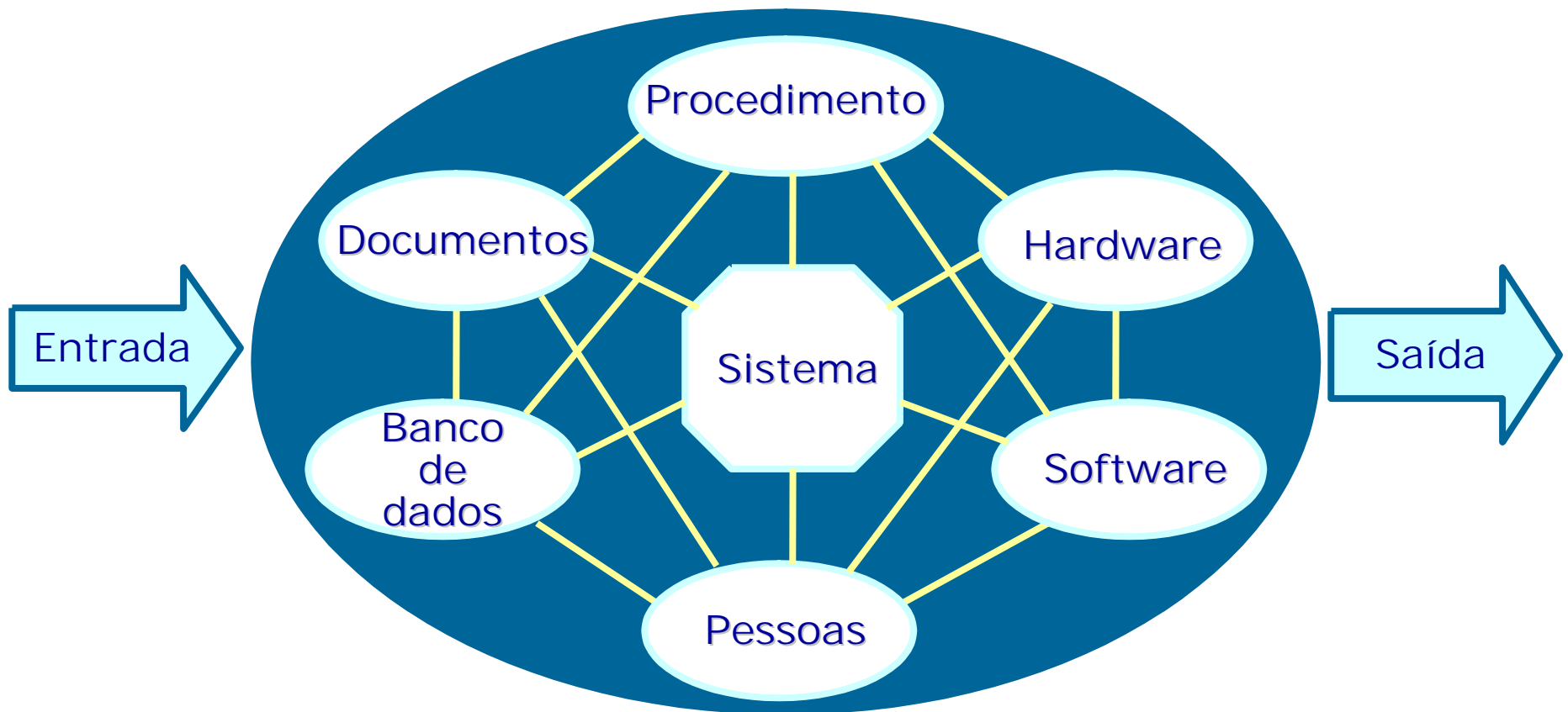




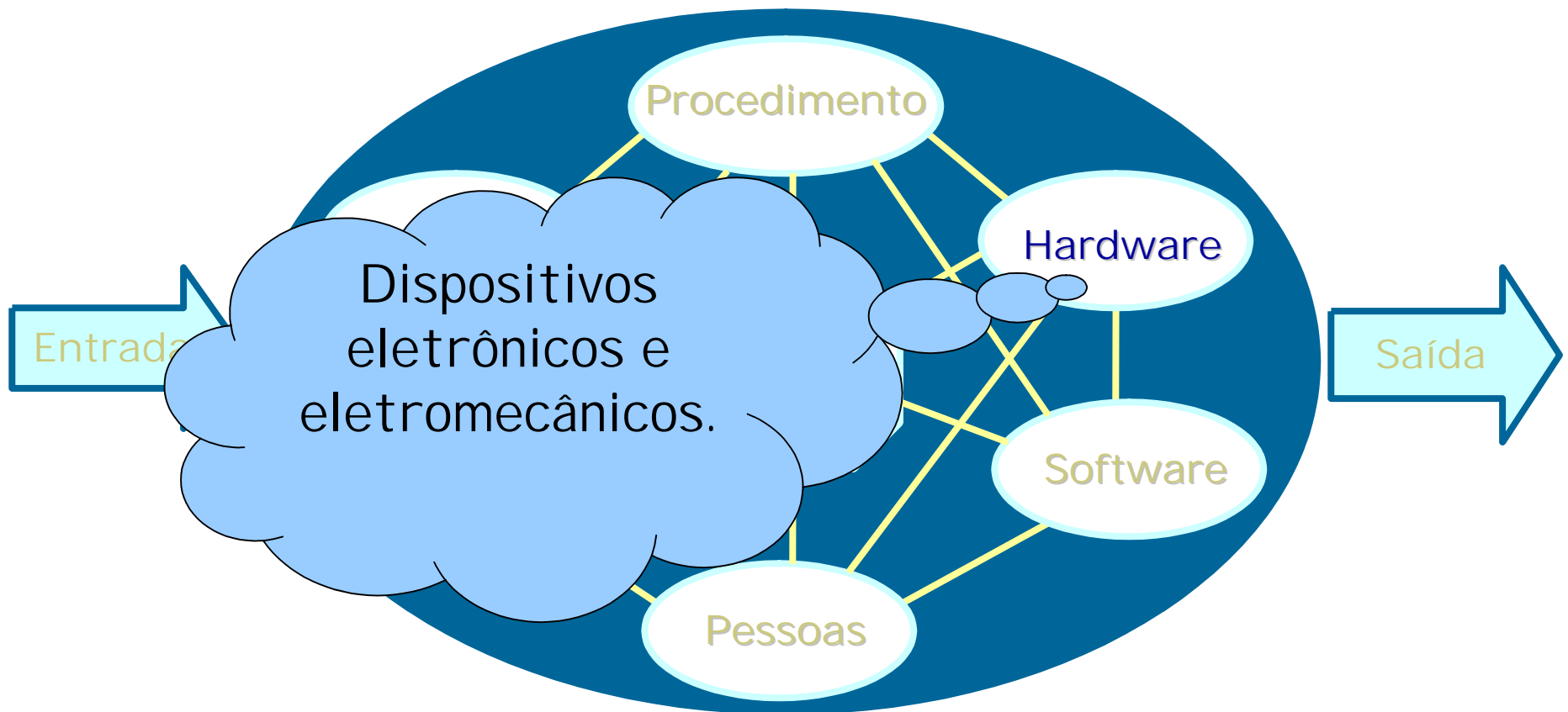
Quais são os elementos
de um SBC?



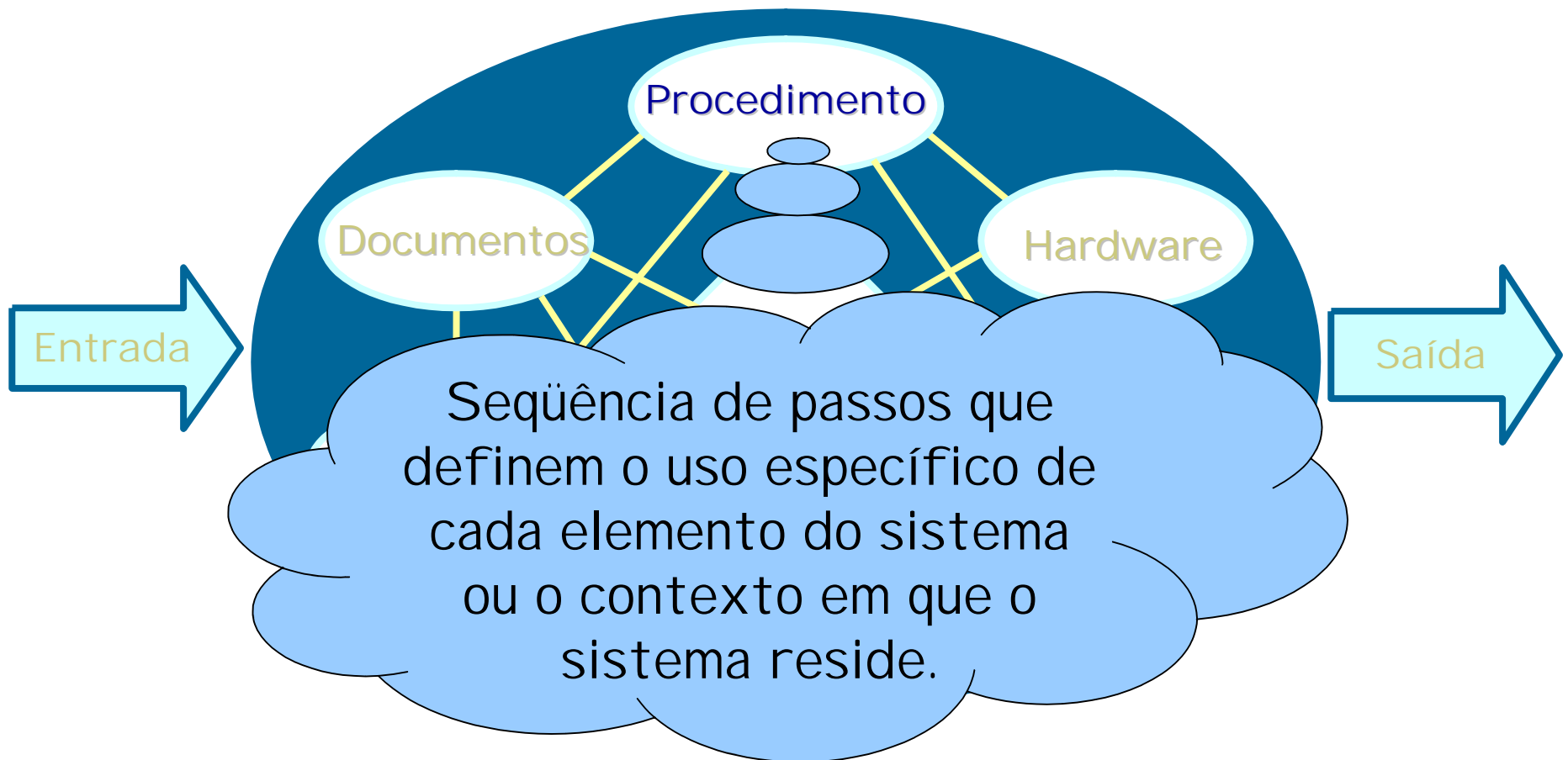
Elementos de um Sistema Baseado em Computador



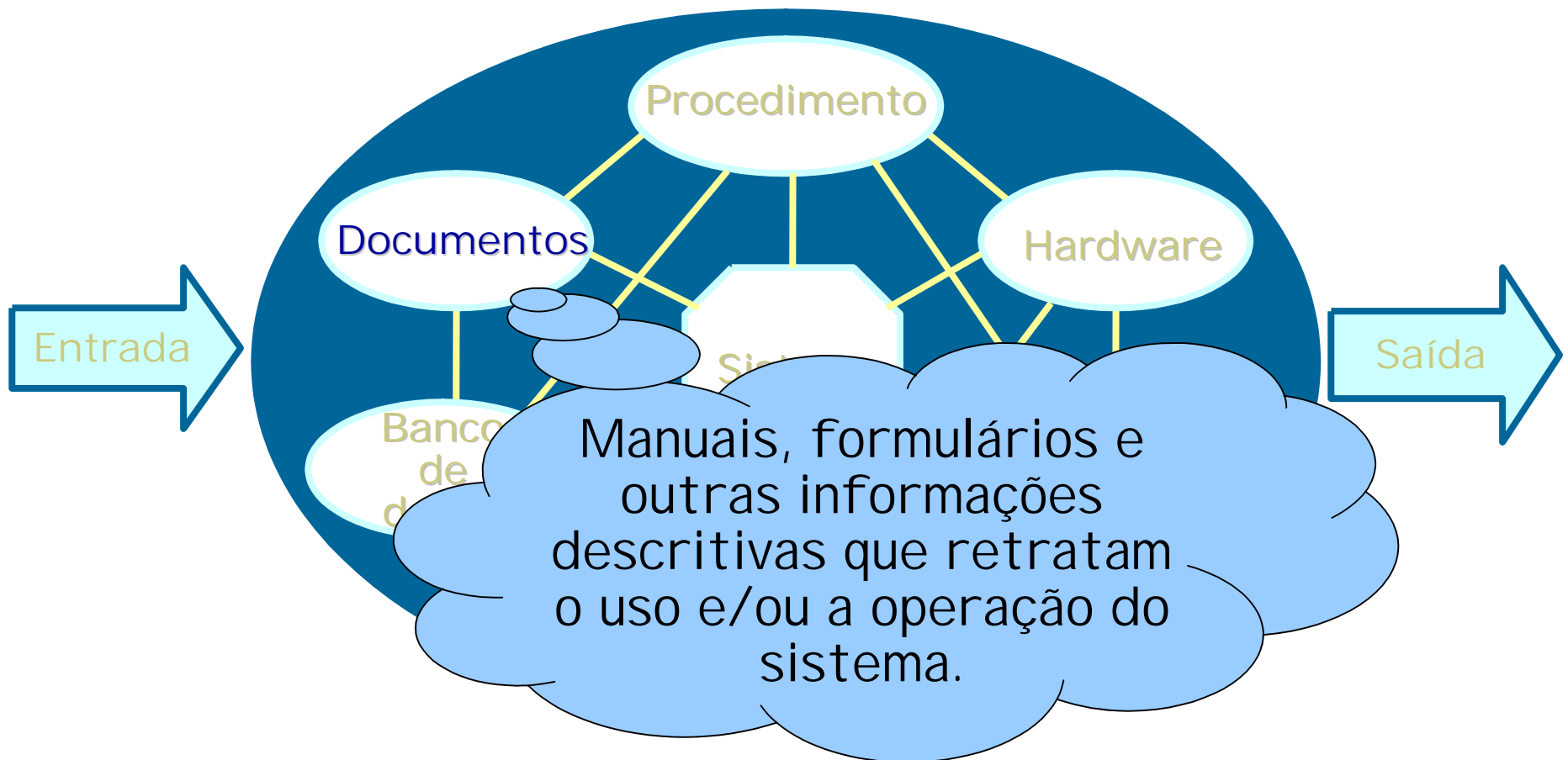
Elementos de um Sistema Baseado em Computador



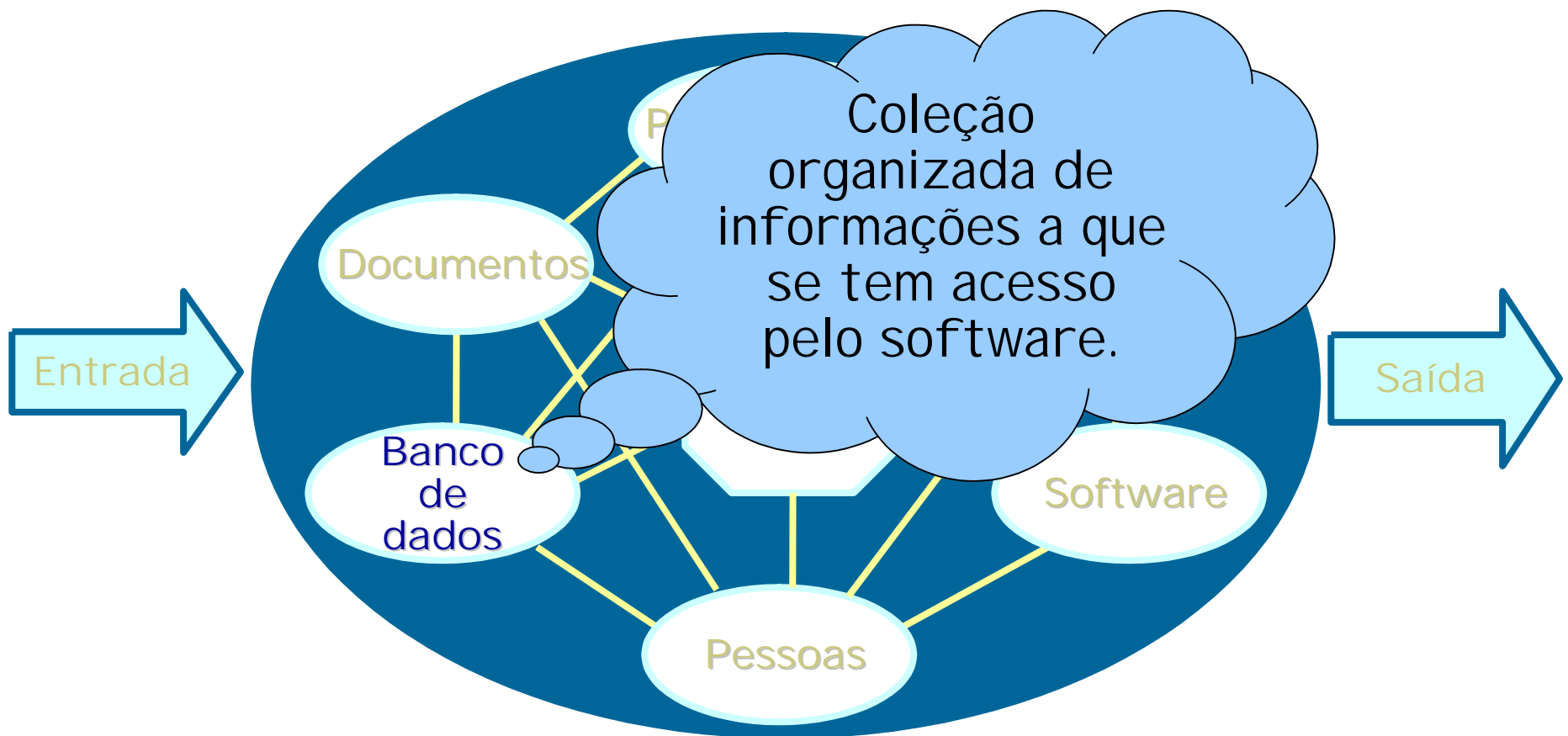
Elementos de um Sistema Baseado em Computador



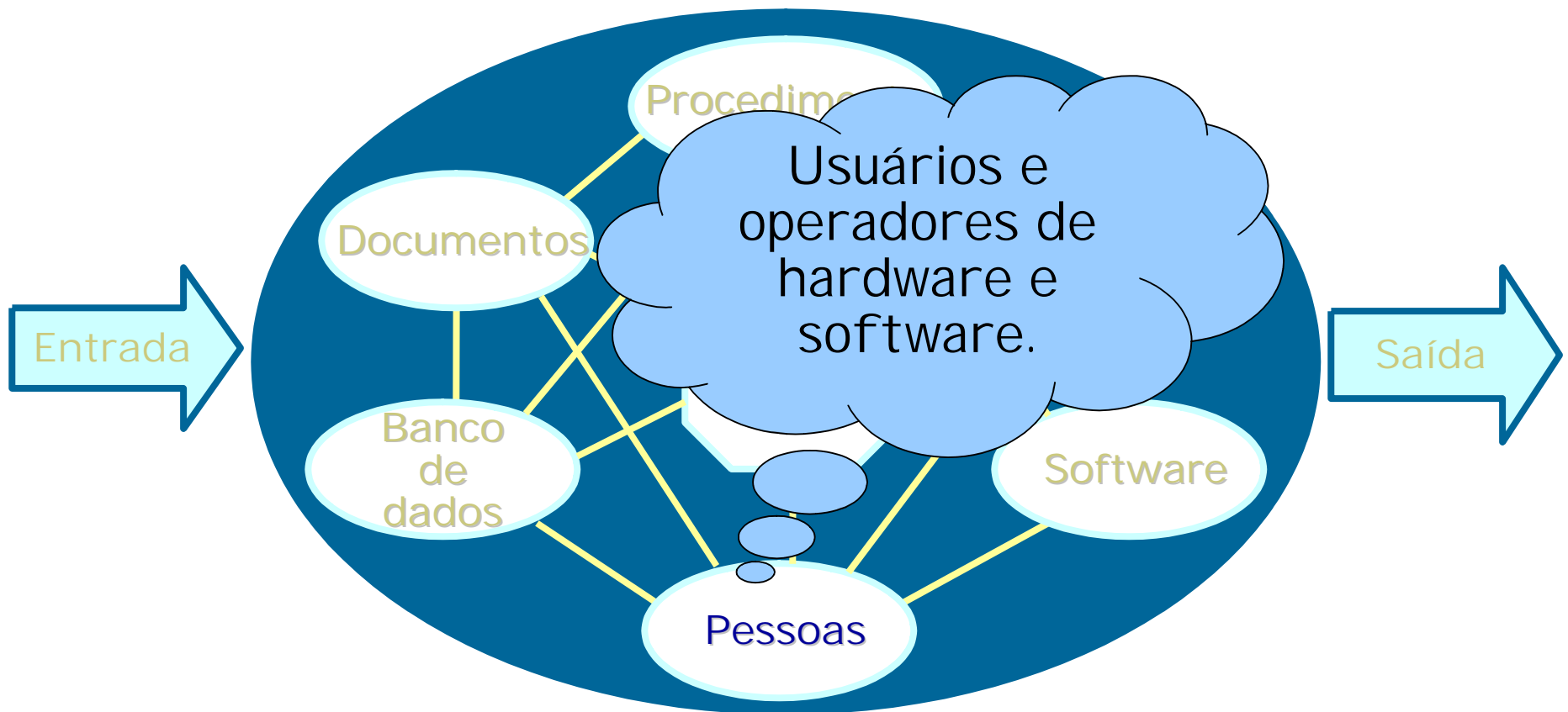
Elementos de um Sistema Baseado em Computador



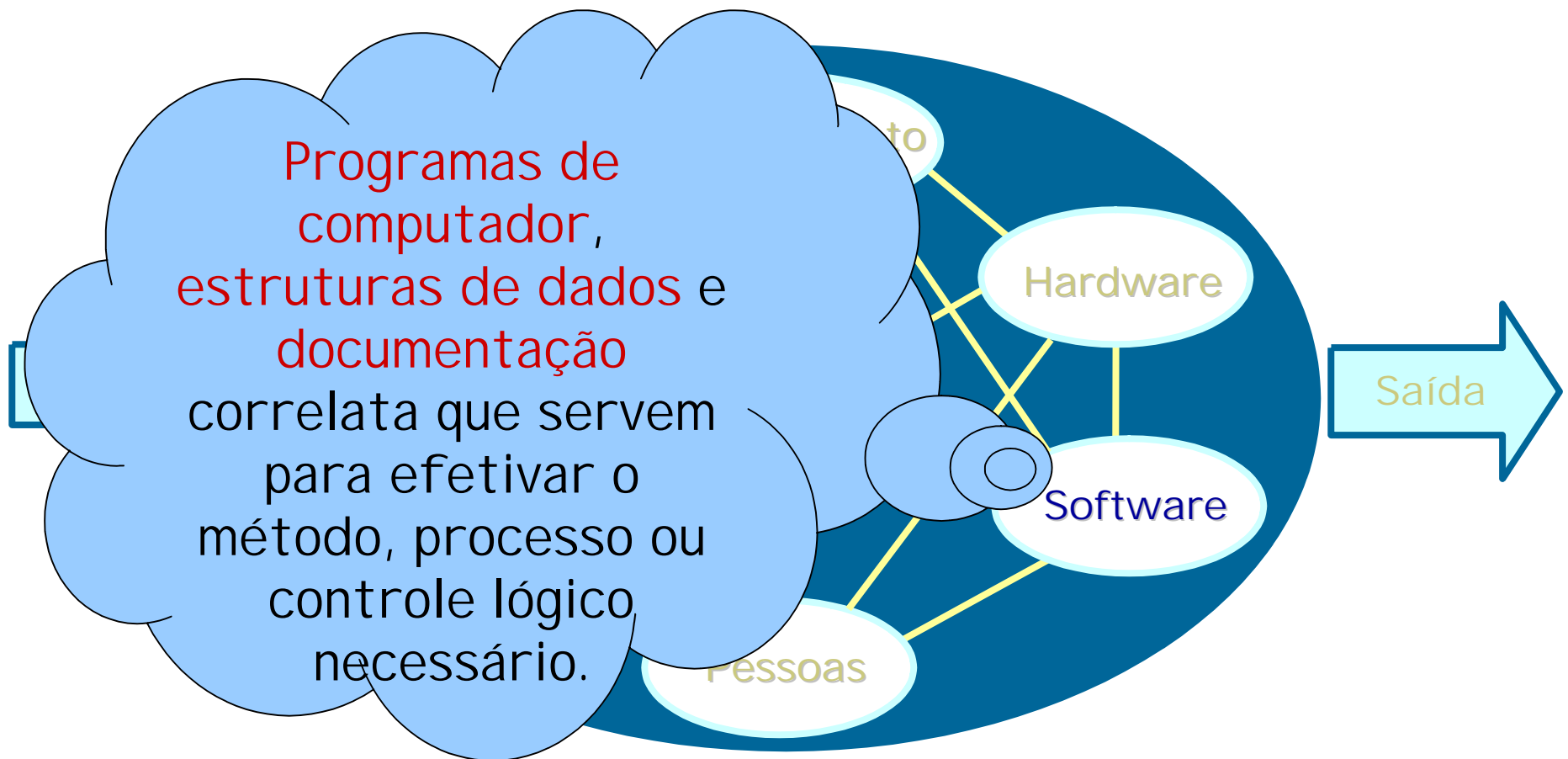
Elementos de um Sistema Baseado em Computador




Elementos de um Sistema Baseado em Computador



Elementos de um Sistema Baseado em Computador



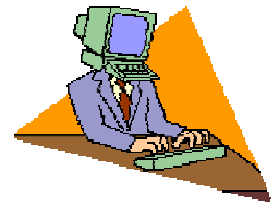


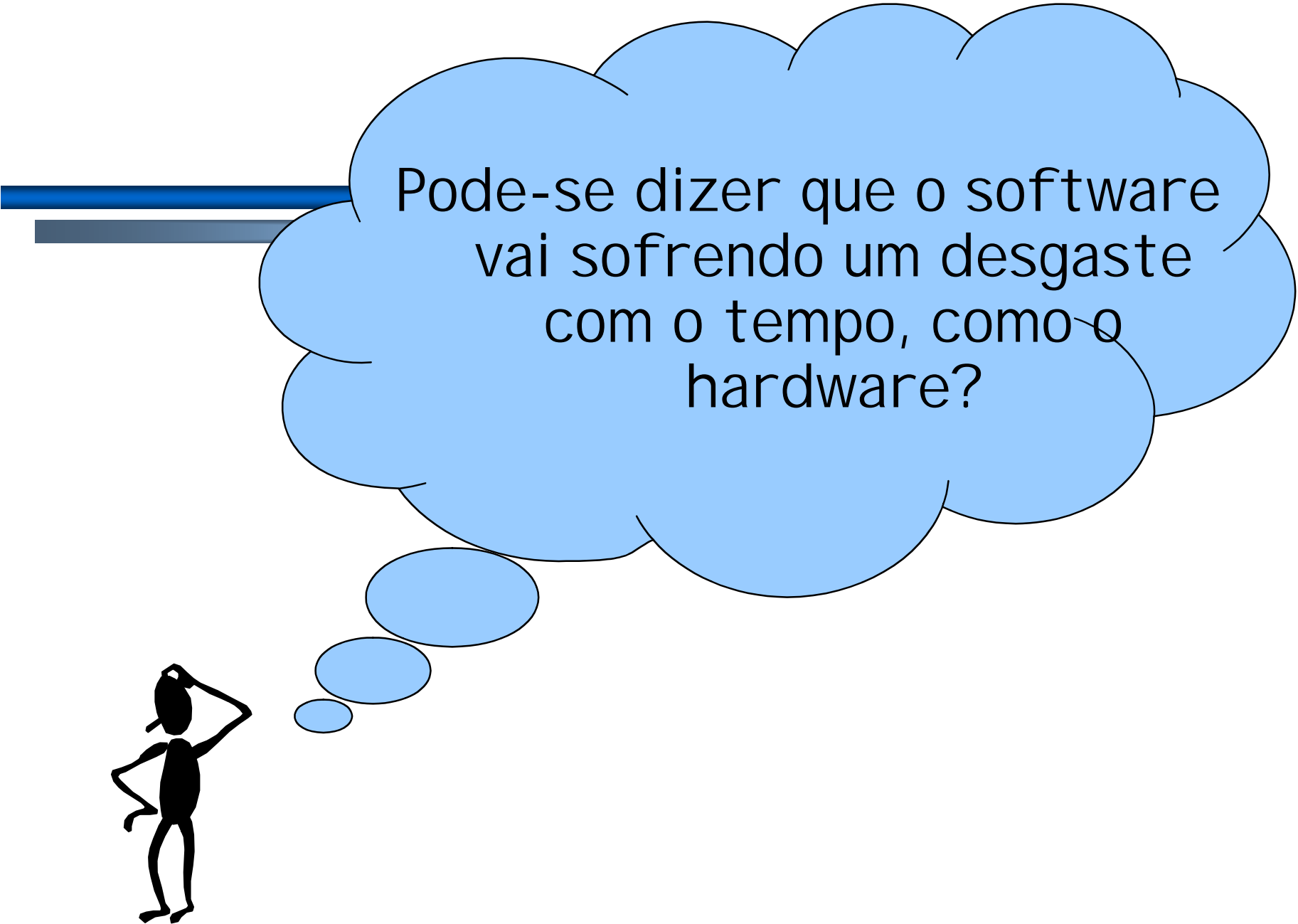
O que é o software?

O software é apenas
um conjunto de
programas?

Definição de Software

- É o produto que os **Engenheiros de Software** projetam e constroem.
- Engloba:
 - **PROGRAMAS** que quando executados 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.





Pode-se dizer que o software
vai sofrendo um desgaste
com o tempo, como o
hardware?

Características do Software

- Desenvolvido ou projetado por engenharia, não manufaturado no sentido clássico.
- A maioria é feita sob medida em vez de ser montada a partir de componentes existentes.
- Não se desgasta mas se deteriora.

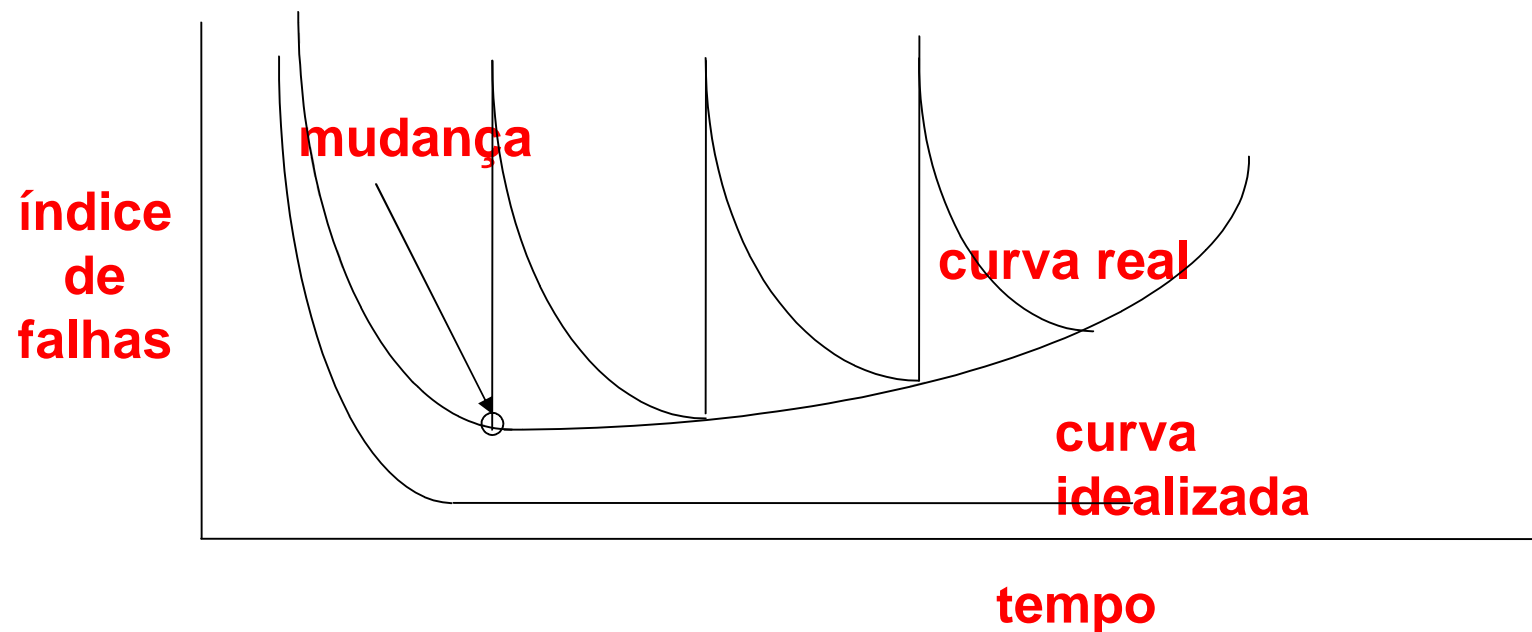


Características do Software



CURVA DE FALHAS DO HARDWARE

Características do Software



CURVA DE FALHAS DO SOFTWARE

Aplicações de Software

■ Software Básico

- Coleção de programas escritos para dar apoio a outros programas.

■ Software de Tempo Real

- Monitora, analisa e controla eventos do mundo real à medida que estes ocorrem.

■ Software Comercial

- Sistemas de operações comerciais e tomadas de decisões administrativas (sistemas de informação).

Aplicações de Software

- Software Científico e de Engenharia
 - Caracterizado por algoritmos de processamento numérico.
- Software Embutido
 - Usado para controlar produtos e sistemas para os mercados industriais e de consumo.
- Software de Computador Pessoal
 - Envolve processamento de textos, planilhas eletrônicas, aplicações gráficas, diversões, etc.

Aplicações de Software

■ Software para Web

- Páginas da Web recuperadas por um browser, incorporando instruções executáveis e dados.

■ Software para Inteligência Artificial

- Faz uso de algoritmos não numéricos para resolver problemas que não são passíveis de computação ou análise direta.
 - Sistemas especialistas, de reconhecimento de padrões, de redes neurais, etc.

Evolução do Software

(1950 - 1965)

- ⇒ O hardware sofreu contínuas mudanças.
- ⇒ O software era uma arte "secundária" para a qual havia poucos métodos sistemáticos.
- ⇒ O hardware era de propósito geral.
- ⇒ O software era específico para cada aplicação.
- ⇒ Não havia documentação.

Evolução do Software

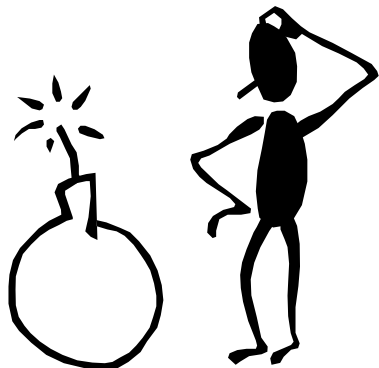
(1965 - 1975)

- ⇒ Multiprogramação e sistemas multiusuários.
- ⇒ Técnicas interativas.
- ⇒ Sistemas de tempo real.
- ⇒ 1ª geração de SGBD's.
- ⇒ Produto de software - software houses.
- ⇒ Bibliotecas de software.

Evolução do Software

(1965 - 1975)

- ⇒ Crescimento do número de sistemas baseados em computador.
- ⇒ Manutenção quase impossível.



..... **CRISE DE SOFTWARE**

Evolução do Software

(1975 - hoje)

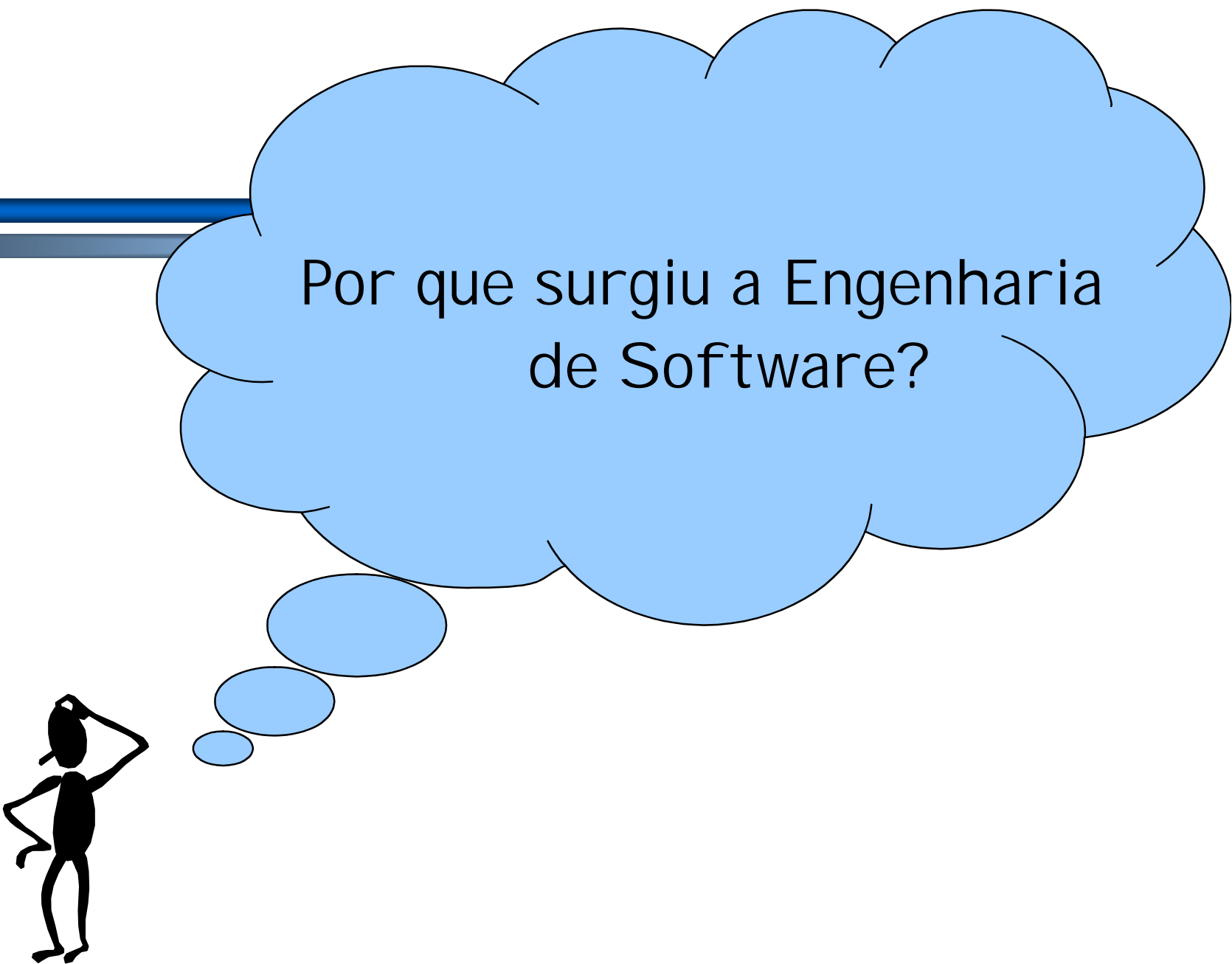
- ⇒ Sistemas distribuídos.
- ⇒ Redes locais e globais.
- ⇒ Uso generalizado de microprocessadores - produtos inteligentes.
- ⇒ Hardware de baixo custo.
- ⇒ Impacto de consumo.

Evolução do Software

(Quarta era do software de computador)

- ⇒ Tecnologias orientadas a objetos, a aspectos, a serviços.
- ⇒ Sistemas especialistas e software de inteligência artificial usados na prática.
- ⇒ Software de rede neural artificial.
- ⇒ Computação paralela.

...



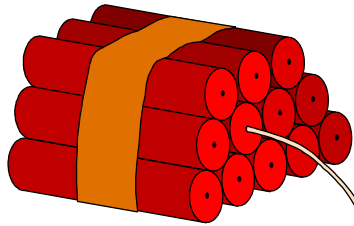
Por que surgiu a Engenharia
de Software?

Evolução do Software

(1965 - 1975)

AFLIÇÃO CRÔNICA

- ⇒ Crescimento baseado em computador.
- ⇒ Manutenção quase impossível.



CRISE DO SOFTWARE

Refere-se a um conjunto de **problemas** encontrados no desenvolvimento de software.

Crise do Software



- As **estimativas** de prazo e de custo freqüentemente são imprecisas.
- Freqüente **insatisfação do cliente** com o sistema concluído.
- A **qualidade do software** é, em geral, menos que adequada.
- O software existente é muito **difícil de manter**.
- Falhas das **pessoas** responsáveis pelo desenvolvimento de software.

Crise do Software: Problemas

- As **estimativas** de prazo e de custo freqüentemente são imprecisas.
 - Não dedicamos tempo para coletar dados sobre o processo de desenvolvimento de software.
 - Sem nenhuma indicação sólida de produtividade, não podemos avaliar com precisão a eficácia de novas ferramentas, métodos ou padrões.

Crise do Software: Problemas

- Freqüente insatisfação do cliente com o sistema concluído.
 - Os projetos de desenvolvimento de software normalmente são executados apenas com um vago indício das exigências do cliente.
 - A comunicação entre o cliente e o desenvolvedor é muito fraca.

Crise do Software: Problemas

- A **qualidade do software** às vezes é menos que adequada.
 - Falta de confiança.
 - Apenas recentemente começaram a surgir conceitos quantitativos sólidos de garantia de qualidade de software.

Crise do Software: Problemas

- O software existente é muito **difícil de manter**.
 - A tarefa de manutenção devora a maior parte do orçamento destinado ao software.
 - A facilidade de manutenção não foi enfatizada como um critério importante para a aceitação do software.

Crise do Software: Problemas

- Falhas das **pessoas** responsáveis pelo desenvolvimento de software.
 - Gerentes sem nenhum background em software.
 - Profissionais da área de software têm pouco treinamento em novas técnicas para o desenvolvimento de software.
 - A “anarquia” ainda reina.
 - Maus hábitos: qualidade e manutenibilidade deficientes.
 - Resistência a mudanças.

Crise do Software:

Causas dos Problemas

■ Mitos do Software

- Atitudes enganosas as quais provocam sérios problemas tanto a gerentes como ao pessoal técnico.
- Propagaram desinformação e confusão.
- Atitudes e hábitos antigos são difíceis de modificar.

Mitos do Software:

Administrativos

Mito 1:

- Já 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?

Mitos do Software:

Administrativos

Realidade:

- Será que o manual é usado?
- Os profissionais sabem que ele existe?
- Ele reflete a prática moderna de desenvolvimento de software?
- Ele é completo?

Mitos do Software:

Administrativos

Mito 2:

- Meu pessoal tem ferramentas de desenvolvimento de software de última geração.

Mitos do Software:

Administrativos

Realidade:

- É preciso muito mais do que os mais recentes computadores e ferramentas para se fazer um desenvolvimento de software de alta qualidade.

Mitos do Software:

Administrativos

Mito 3:

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

Mitos do Software:

Administrativos

Realidade:

- O desenvolvimento de software não é um processo mecânico igual à manufatura.
 - Acrescentar pessoas em um projeto pode torná-lo ainda mais atrasado.
- Pessoas podem ser acrescentadas, mas somente de uma forma planejada.

Mitos do Software:

Cliente

Mito 1:

- Uma declaração geral dos objetivos é suficiente para se começar a escrever programas.
 - Podemos preencher os detalhes mais tarde.

Mitos do Software:

Cliente

Realidade:

- Uma definição inicial ruim é a principal causa de fracassos dos esforços de desenvolvimento de software.
- É fundamental uma descrição formal e detalhada do domínio da informação, função, desempenho, interfaces, restrições de projeto e critérios de validação.

Mitos do Software:

Cliente

Mito 2:

- Os requisitos de projeto modificam-se continuamente, mas as mudanças podem ser facilmente acomodadas, porque o software é flexível.

Mitos do Software:

Cliente

Realidade:

- Uma mudança, quando solicitada tardiamente num projeto, pode ser maior do que a ordem de magnitude mais dispendiosa da mesma mudança solicitada nas fases iniciais.

Mitos do Software: Profissional

Mito 1:

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

Mitos do Software: Profissional

Realidade:

- Os dados da indústria indicam que entre 50 e 70% de todo esforço gasto em um programa são despendidos depois que ele é entregue pela primeira vez ao cliente.

Mitos do Software: Profissional

Mito 2:

- Enquanto não tiver o programa "funcionando", eu não terei realmente nenhuma maneira de avaliar sua qualidade.

Mitos do Software: Profissional

Realidade:

- Um programa funcionando é somente uma parte de uma Configuração de Software que inclui todos os itens de informação produzidos durante a construção e manutenção do software.

Resposta à Crise de Software

Engenharia de Software

A aplicação de uma abordagem **sistemática**, **disciplinada** e possível de ser **medida** para o **desenvolvimento**, **operação** e **manutenção** do software (*IEEE*).



Processo de
Software

O Processo de Software

- Elementos fundamentais:
 - Métodos
 - Ferramentas
 - Procedimentos
- Para projetar, construir e manter grandes sistemas de software de forma profissional.

O Processo de Software

- **MÉTODOS:** proporcionam os detalhes de como fazer para construir o software.

Gerenciamento do Projeto

Análise de sistemas

Análise de requisitos

Projeto do software

Construção do algoritmo

Geração do código

Teste

Manutenção

...

O Processo de Software

- **FERRAMENTAS:** dão suporte automatizado aos métodos.
 - Existem atualmente ferramentas para **apoiar** cada um dos métodos.
 - **CASE** (Computer Aided Software Engineering)
 - Quando as ferramentas são integradas é estabelecido um sistema de suporte ao desenvolvimento de software.



O Processo de Software

- **PROCEDIMENTOS:** constituem o elo de ligação entre os métodos e ferramentas.
 - **Seqüência** em que os métodos serão aplicados.
 - **Produtos** que se exige que sejam entregues.
 - **Controles** que ajudam assegurar a qualidade e coordenar as alterações.
 - **Marcos de referência** que possibilitam administrar o progresso do software.

Um Processo de Software com Qualidade

■ Inteligibilidade

- O processo é definido e inteligível.

■ Visibilidade

- O progresso do processo é visível externamente.

■ Suportabilidade

- O processo pode ser apoiado por ferramentas CASE.

Um Processo de Software com Qualidade

■ Aceitabilidade

- O processo é aceito por todos os envolvidos.

■ Confiabilidade

- Os erros do processo são descobertos antes que resultem em erros no produto.

■ Robustez

- O processo pode continuar a despeito de problemas inesperados.

Um Processo de Software com Qualidade

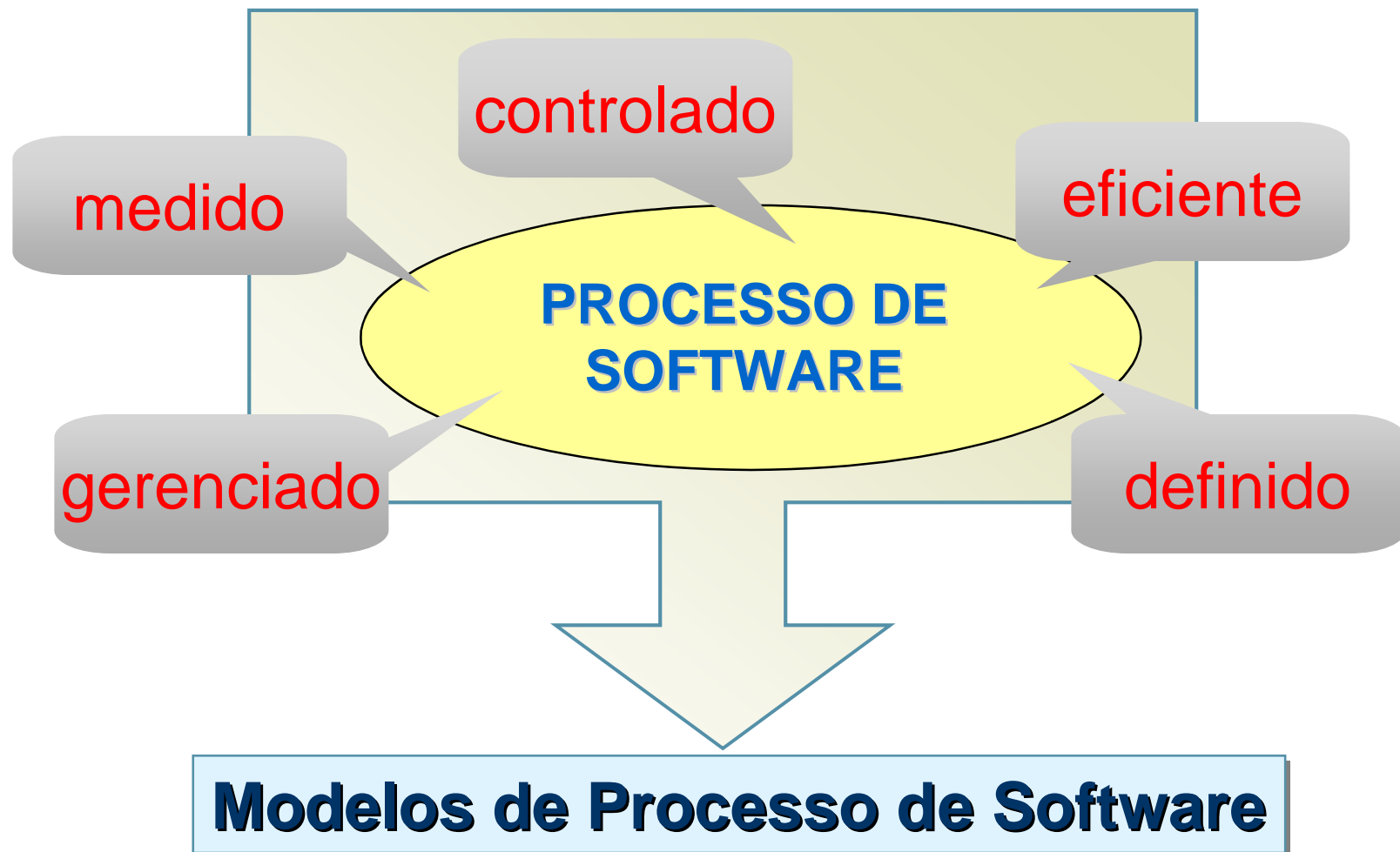
■ Manutenibilidade

- O processo pode evoluir para atender alterações das necessidades organizacionais.

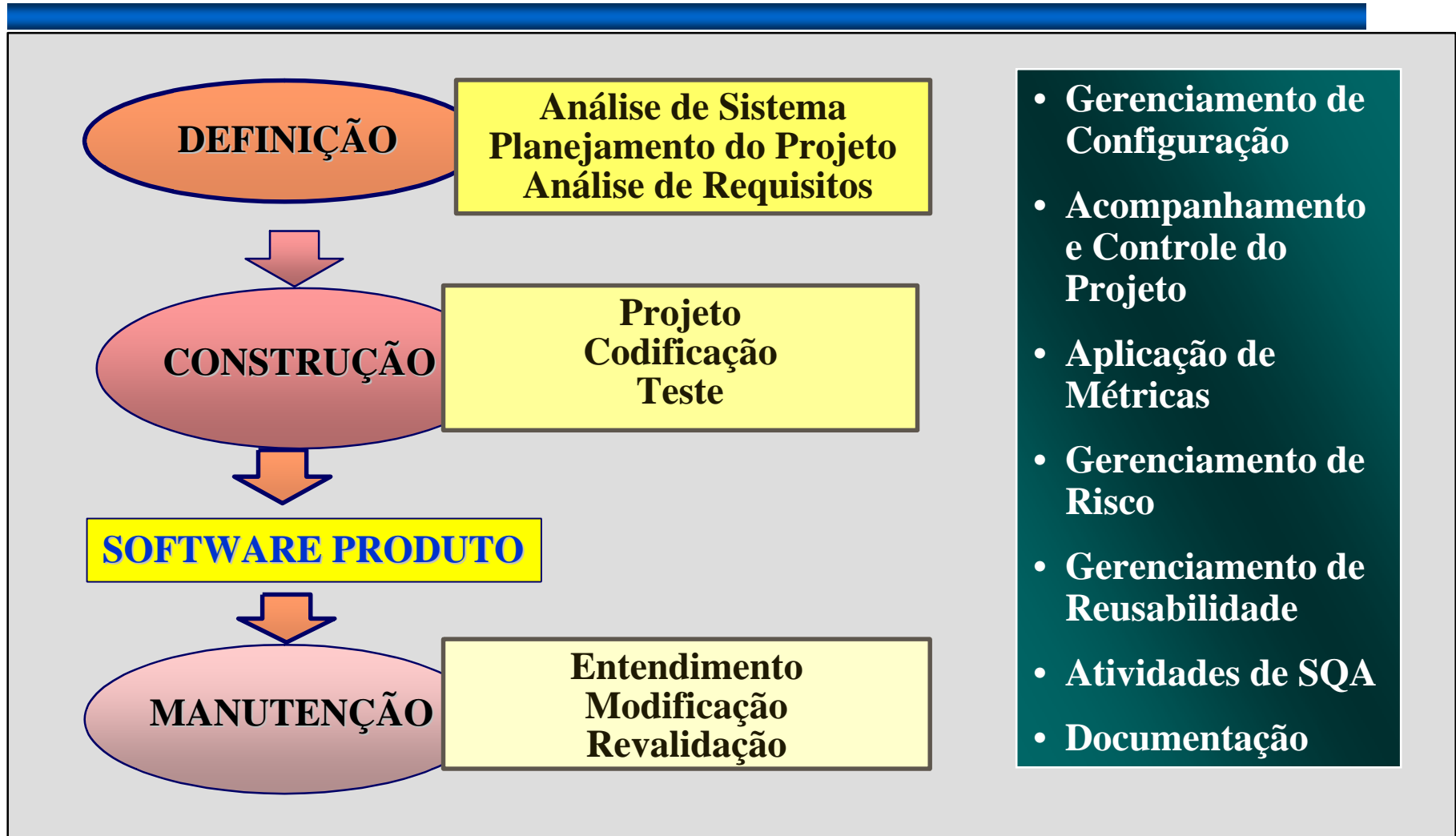
■ Velocidade

- Quão rápido o sistema pode ser produzido.

Um Processo de Software com Qualidade



Fases Genéricas dos Modelos de Processo de Software



Fase de Definição

Focaliza **o quê** será desenvolvido.

- Que informação vai ser processada.
- Que função e desempenho são desejados.
- Que comportamento pode ser esperado do sistema.
- Que interfaces vão ser estabelecidas.
- Que restrições de projeto existem.
- Que critérios de validação são exigidos para definir um sistema bem-sucedido.
- Que tarefas serão realizadas.

Fase de Definição

Focaliza **o quê** será desenvolvido.

Três tarefas principais ocorrem de alguma forma:

- Análise de sistemas.
- Planejamento do projeto de software.
- Análise de requisitos.

Fase de Desenvolvimento

Focaliza **como** o software será desenvolvido.

- Como os dados devem ser estruturados.
- Como a função deve ser implementada dentro da arquitetura do software.
- Como os detalhes procedimentais devem ser implementados.
- Como as interfaces devem ser caracterizadas.
- Como o projeto será traduzido em uma linguagem de programação.
- Como os testes serão realizados.

Fase de Desenvolvimento

Focaliza **como** o software será desenvolvido.

Três tarefas técnicas específicas devem sempre ocorrer:

- Projeto de software.
- Geração de código.
- Teste de software.

Fase de Manutenção

Focaliza as **mudanças** que ocorrerão depois que o software for liberado para uso operacional.

- A fase de manutenção reaplica os passos das fases de definição e desenvolvimento, mas faz isso no contexto de um software já existente.

Fase de Manutenção

Focaliza as **mudanças** que ocorrerão depois que o software for liberado para uso operacional.

- **Correção** de erros/defeitos.
- **Adaptações** exigidas conforme o ambiente do software evolui.
- Mudanças devido a **melhoramentos** ocorridos por alterações nos requisitos dos clientes.

Atividades de Apoio

- As três fases genéricas do processo de software são complementadas por uma série de atividades de apoio.
 - Atividades guarda-chuva.
- As atividades de apoio são aplicadas durante toda a engenharia do software.

Atividades de Apoio

Two horizontal bars are positioned below the title. The top bar is a solid blue line. The bottom bar is a thicker, semi-transparent blue line with a gradient effect.

- Gerenciamento de configuração de software.
- Controle e acompanhamento do projeto de software.
- Aplicação de métricas.
- Gerenciamento de riscos.
- Gerenciamento de reusabilidade.
- Garantia de qualidade de software.
- Preparação e produção de documentos.
- ...

Concluindo...😊

O Produto Software....

- Tem uma natureza um pouco diferente dos “produtos tradicionais”.
- O software é abstrato.
 - É a expressão, traduzida em instruções para computadores, da solução desenvolvida por um ser humano (ou vários!) para um problema.
 - Pode ser tão simples quanto calcular o total de uma lista de compras, ou tão complexo quanto calcular a fatura de um telefone celular.

Assim, é preciso

- Compreender bem o problema do cliente.
- Desenvolver uma solução.
- Garantir que ela funcione.
- Transformar a solução em um software.
- Entregar o software ao cliente no custo e prazo combinados.
- Oferecer treinamento aos usuários.
- Realizar alterações quando o cliente pedir.
- Manter o controle da versão do software que cada cliente possui
- **Tudo enquanto...**
 - Negocia preços e prazos.
 - Paga o salário dos funcionários.
 - Capacita os funcionários.
 - Se mantém atualizado com o mercado tecnológico.
- **E ainda** busca maneiras de fazer tudo isso cada vez melhor, mais rápido e mais barato do que a concorrência.

Assim, é preciso

- Compreender bem o problema do cliente.
- Desenvolver uma solução.
- Garantir a qualidade.
- Traçar o plano de trabalho.
- Entender o contexto e o prazo combinados.
- Oferecer o melhor preço.
- Realizar alterações quando o cliente pedir.
- Manter o controle da versão do software que cada cliente possui
- **Tudo enquanto...**
 - Negocia preços e condições
 - Paga o salário dos funcionários
 - Capacita os funcionários
 - Se mantém atualizado
- **E ainda** busca maneiras de fazer tudo isso cada vez melhor, mais rápido e mais barato do que a concorrência.

Parece não ser simples...

E não é!!!



Engenharia de Software

Visão Geral

