

Análise e Projeto de Sistemas

Antônio da Mota Moura Júnior

jmoura.unibh@gmail.com

Introdução a Conceitos de Sistemas

Análise e Projeto de Sistemas

Parte 01

Conteúdo

- Definição de sistema
- Classificação dos sistemas
- Princípio geral dos sistemas

Introdução

- “a análise [de sistemas] é frustrante, repleta de relacionamentos entre pessoas, indefinida e difícil. Resumindo, é fascinante. Depois que você é fisgado, os velhos e fáceis prazeres da construção de sistemas nunca mais serão suficientes para satisfazê-lo.”

Tom DeMarco

Sistemas

- “...vivemos em um mundo de sistemas - e de sistemas dentro de outros sistemas, que são componentes de sistemas ainda maiores. Assim, tudo que fazemos em nossa vida pessoal e profissional tem impacto (muitas vezes imprevisto ou inesperado) nos diversos sistemas de que somos parte.”

Edward Yourdon

Definições de Sistema

- um grupo de itens que interagem entre si ou que sejam interdependentes, formando um todo unificado. Ex.: sistema numérico; gravitacional; digestivo; sistema de processamento de dados.
- um conjunto organizado de doutrinas, idéias ou princípios, habitualmente previsto para explicar a organização ou o funcionamento de um conjunto sistêmico. Ex.: sistema da mecânica newtoniana
- um procedimento organizado ou estabelecido. Ex: sistema de toques de digitação
- uma maneira de classificar, simbolizar ou esquematizar. Ex.: sistema taxonômico, sistema decimal
- organização harmoniosa ou modelo: ORDEM
- sociedade organizada ou situação social vista como indesejável

Definição de Sistema

(para o desenvolvimento de software)

- “Um sistema pode ser entendido como um mecanismo composto por um conjunto de partes interrelacionadas, onde cada parte está sempre relacionada a, pelo menos, uma das outras partes.”

S. Pompilho

Classificação dos Sistemas

- Sistemas Naturais
 - Sistemas físicos (sistemas estelares, geológicos, moleculares etc.)
 - Sistemas vivos (envolvem animais, plantas e seres humanos)
- Sistemas feitos pelo Homem
 - Sistemas sociais (organização de leis, doutrinas, costumes etc.)
 - coleção organizada e disciplinada de idéias (organização de livros)
 - Sistemas de transporte (redes rodoviárias, linhas aéreas etc.)
 - Sistemas de comunicações (telefone, sinais de fumaça/manuais...)
 - Sistemas de manufatura (fábrica, linhas de montagem etc.)
 - Sistemas financeiros (contabilidade, inventário, livro-razão etc.)
 - **Sistemas automatizados (Sistemas de Informação)**

Sistemas Automatizados

- Sistemas feitos pelo homem, que interagem com ou são controlados por um ou mais computadores.
- Composto por: hardware de computadores, softwares, pessoas, dados, procedimentos
- Classificação:
 - Sistemas on-line (sistema de processamento de transações)
 - Sistema de tempo-real (variação dos sistemas on-line)
 - Sistemas de apoio à decisão e planejamento estratégico
 - Sistemas baseados no conhecimento (sistemas especialistas, sistemas inteligentes)

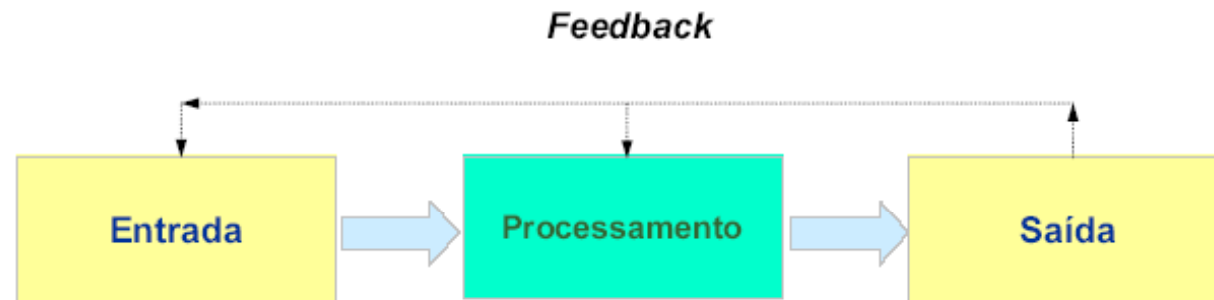
Sistema de Informação

- Em muitos casos é um tipo de sistema automatizado.
- Um SI é uma combinação de pessoas, dados, processos, interfaces, redes de comunicação e tecnologia que interagem com o objetivo de dar suporte e melhorar o processo de negócio de uma organização com relação às informações.
 - Vantagens do ponto de vista competitivo.
- Objetivo principal e final da construção de um SI: *adição de valor à organização.*

Eduardo Bezerra

Sistemas de Informação

- São sistemas que, através de processamento de coleta e tratamento de dados, geram e disseminam as informações necessárias aos diversos níveis e processos organizacionais.

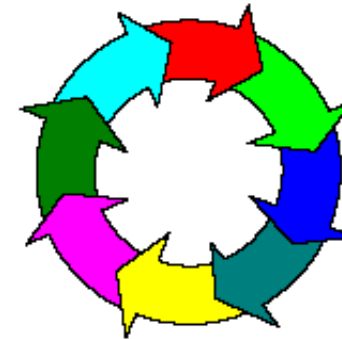


Conceitos

- **Dados:** seqüências ordenadas de símbolos das quais se pode extrair informação mas que, isoladamente, não contêm nenhum significado.
- **Informações:** dados tratados capazes de transmitir algum conhecimento ao receptor.

Tecnologia da Informação

- Para a construção de um SI utilizamos a tecnologia da informação.
- Diz respeito a tudo o que pode ser feito com a informação
- Ciclo do Processo Informativo:
 - Especificação (identificação de necessidades e requisitos)
 - Obtenção (origem e captura)
 - Armazenamento (classificação, estruturação, armazenamento e recuperação)
 - Manipulação (tratamento, formatação e apresentação)
 - Distribuição (identificação do destino e transmissão)
 - Uso (análise e uso)



Classificação dos Sistemas de Informação

- Os Sistemas de Informação podem ser classificados pelo tipo de informação gerado pelo sistema e qual o perfil de usuário que utiliza ou necessita daquelas informações.



Sistema de Software

- Um dos componentes de um SI é denominado **sistema de software**.
- Compreende os módulos funcionais computadorizados que interagem entre si para proporcionar a automatização de diversas tarefas.
- Característica intrínseca do desenvolvimento de sistemas de software: **complexidade** (cresce com o tamanho do software).

Princípios Gerais de Sistemas

- Quanto mais especializado é um sistema, menos capaz ele é de se adaptar a circunstâncias diferentes.
- Quanto maior for um sistema, maior o número de seus recursos que serão destinados à manutenção diária.
- Os sistemas sempre fazem parte de sistemas maiores e sempre podem ser divididos em sistemas menores.
- Os sistemas crescem.

Desenvolvimento de Sistemas

Análise e Projeto de Sistemas

Parte 02

Conteúdo

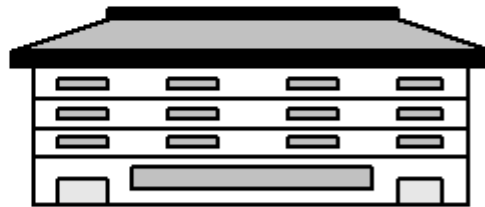
- Introdução ao desenvolvimento de sistemas
- Participantes do desenvolvimento de sistemas
- Problemas típicos do desenvolvimento de sistemas
- Modelagem de sistemas

Desenvolvimento de Sistemas

- Complexidade X Tamanho



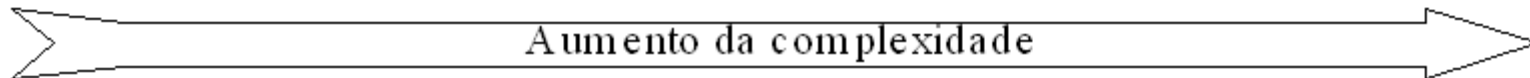
Casa de cachorro



Casa



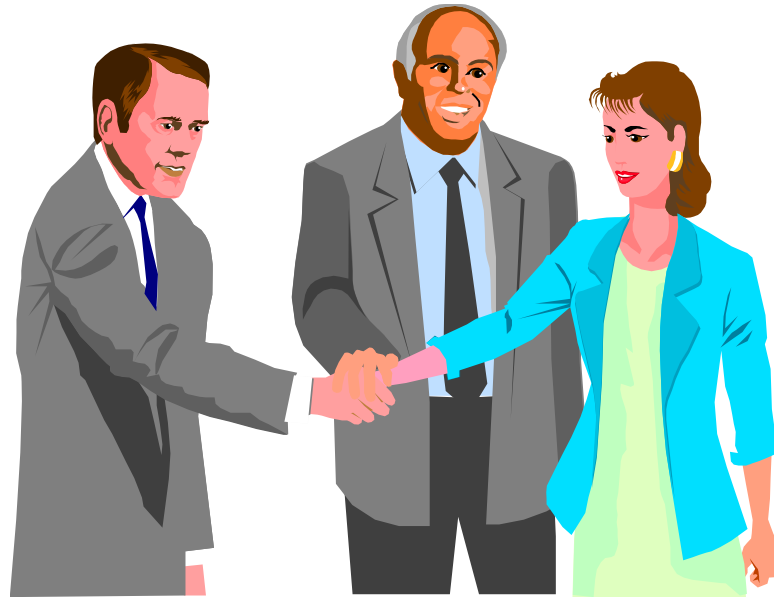
Arranha-Céus



Desenvolvimento de Sistemas

- Resumidamente, podemos afirmar que para desenvolvermos sistemas de software de uma maneira controlada e organizada, que reduza os problemas típicos desta atividade, é necessário obedecermos a uma metodologia, baseada em processo de software, que define atividades e papéis dos participantes; e utilizarmos ferramentas para a confecção dos modelos que representarão o software a ser construído.

Análise e Projeto de Sistemas I



Participantes do
Desenvolvimento de Sistemas

Participantes no Desenvolvimento de Sistemas

- Usuários
- Gerentes
- Auditores, pessoal do controle de qualidade e “mantenedores dos padrões”
- Analistas de sistemas
- Projetistas de sistemas
- Programadores
- Operadores

Classificação dos Usuários

- Por função:
 - operativos
 - supervisores
 - executivos
- Por nível de experiência com processamento de dados:
 - amadores
 - novatos
 - peritos

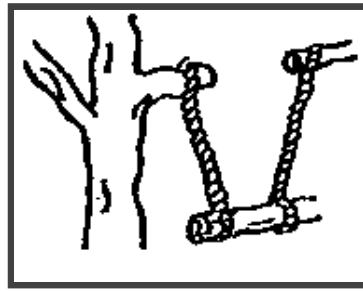
Participação do Usuário

- A participação do usuário durante o desenvolvimento de um sistemas é extremamente importante.

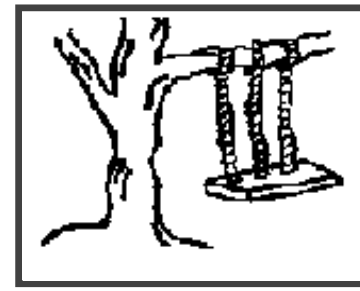
História de um Desenvolvimento



O que o usuário pediu



Como o analista viu



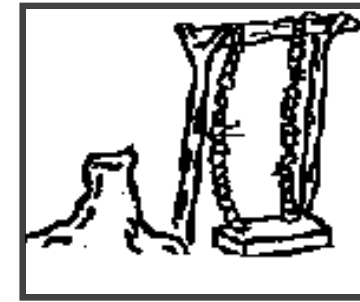
Como foi projetado



Como foi programado



O que o usuário queria



Como funciona de fato

Tarefas do Analista de Sistemas

- “... analisar ou estudar o sistema para determinar-lhe a essência: seu comportamento exigido independentemente da tecnologia utilizada em sua implementação. Na maioria dos casos, só estaremos em situação de decidir se faz sentido usar um computador para executar as funções do sistema depois de modelarmos seu comportamento.” (Edward Yourdon)
- “... o papel do analista de sistemas é especificar quais são os requisitos do sistema do ponto de vista da eficácia, ou seja, garantir que o sistema alcance os objetivos globais da empresa. Trata-se de certificar-se de que o sistema fará o que precisa ser feito, fará o que é certo ser feito, independentemente da instrumentação que será usada para se chegar a esse objetivo.” (S.Pompilho)

Habilidades do Analista de Sistemas

- “... [o analista de sistemas] deverá ter maiores habilitações que um programador: além do conhecimento do hardware e do software do computador, [ele] deve ser capaz de se comunicar com pessoas leigas em computação e estar familiarizado com as aplicações comerciais dessas pessoas.” (Edward Yourdon)
- Habilidades:
 - **Comunicação**
 - **Capacidade de análise**
 - **Conhecimento da área usuária**
 - **Capacidade de negociação**
 - **Administração de projetos**
 - **Conhecimento técnico**

Tarefas do Projetista de Sistemas

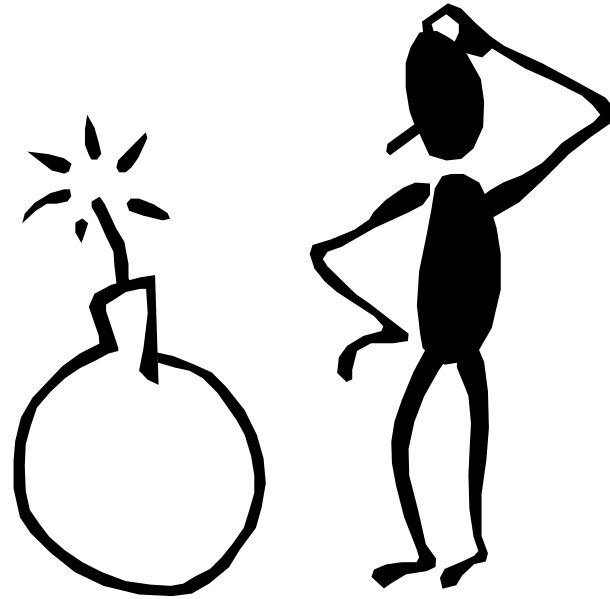
- “... transformar uma lista isenta de tecnologia dos requisitos do usuário em um projeto arquitetural de alto-nível que fornecerá a estrutura com a qual os programadores poderão trabalhar.” (Edward Yourdon)
- “... o projetista [de sistemas] tem um papel voltado para a eficiência, isto é, voltado para obtenção do melhor desempenho individual dos componentes do sistema.” (S.Pompilho)
- “...(1) avaliar as alternativas de solução (da definição) do problema resultante da análise e (2) gerar a especificação de uma solução computacional detalhada” (Eduardo Bezerra)

Habilidades do Projetista de Sistemas

- Habilidades:
 - **Comunicação**
 - **Conhecimento do ambiente tecnológico da área usuária**
 - **Capacidade de negociação**
 - **Conhecimento técnico**
- Há vários tipos de projetistas: de interface, de rede, de banco de dados etc.

Analista X Projetista

- “... é importante para o analista e o projetista de sistemas permanecerem em estreito contato durante todo o projeto.”
(Edward Youdon)
- Em muitos casos, o analista de sistemas e o projetista são a mesma pessoa.
- Em outros casos, parte das atividades do projetista são executadas pelo analista e parte pelo programador.



Problemas Típicos do Desenvolvimento de Sistemas

Problemas no Processo de Desenvolvimento de Sistemas

- Má qualidade dos **produtos** gerados (sistemas)
- Baixa produtividade dos **processos** responsáveis pelas atividades de desenvolvimento e manutenção de sistemas.

Exemplos de Problemas

- Problemas de qualidade dos produtos:
 - Sistemas que não contribuem para os objetivos da organização
 - Sistemas não-confiáveis
 - Sistemas ineficientes
 - Sistemas de manutenção constante, difícil e onerosa
- Problemas na Produtividade do Processo:
 - Prazos não cumpridos
 - Custos acima da previsão
 - Perdas de oportunidades
 - Aumento do **backlog** documentado e invisível

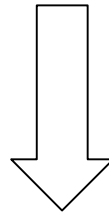


Causa dos Problemas

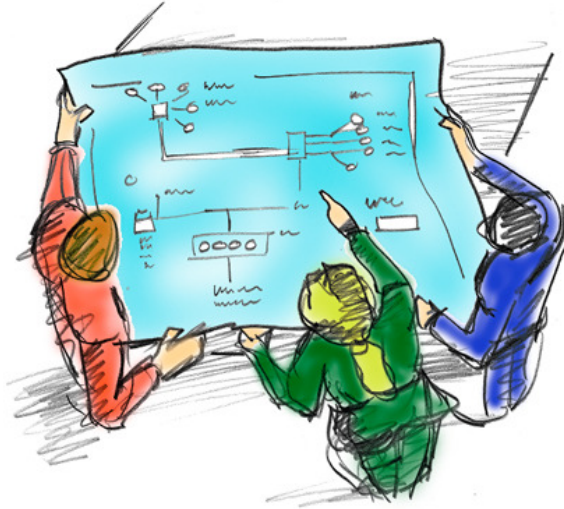
- Ausência de planejamento.
- Falta de integração do planejamento de sistemas com o planejamento estratégico da empresa.
- Não utilização de métodos e técnicas formais de desenvolvimento de sistemas.
- Adoção de metodologias não-ambientadas à realidade da empresa.
- Não utilização de ferramentas de produtividade.
- Falta de definição precisa dos objetivos e requisitos do sistema.
- Dificuldade de comunicação e/ou falta de entrosamento entre os envolvidos no processo (técnicos e usuários).
- Falta de precisão e clareza dos textos narrativos utilizados na especificação dos sistemas.

Como Reduzir os Problemas

- Aplicação de teorias, metodologias, técnicas, métodos e ferramentas visando minimizar os problemas e tornar os projetos de sistemas menos artesanais, mais próximos da engenharia.



Modelagem de Sistemas



Modelagem de Sistemas

O que significa Modelar

- Criar uma linguagem para comunicar decisões que não são óbvias ou que não podem ser inferidas.
- Prover uma semântica rica o suficiente para capturar o que é importante a nível tático e estratégico.
- Na atividade de modelagem construímos modelos que representam um sistema.

Modelo de Software

- Na construção de sistemas de software, assim como na construção de sistemas habitacionais, há uma gradação de complexidade.
 - A construção desses sistemas necessita de um planejamento inicial.
- Um modelo pode ser visto como uma representação idealizada de um sistema que se planeja construir.
- Maquetes de edifícios e de aviões e plantas de circuitos eletrônicos são apenas alguns exemplos de modelos.



Modelo de Software

- E. Yourdon apresenta três motivos para construção de modelos:
 - para focalizar características importantes de sistemas deixando de lado as menos importantes;
 - para discutir alterações e correções nos requisitos do usuário a baixo custo e mínimo risco;
 - para confirmar que entendemos o ambiente do usuário e o documentamos de uma tal forma que os projetistas de sistemas e programadores podem construir o sistema.

Ferramentas de Modelagem

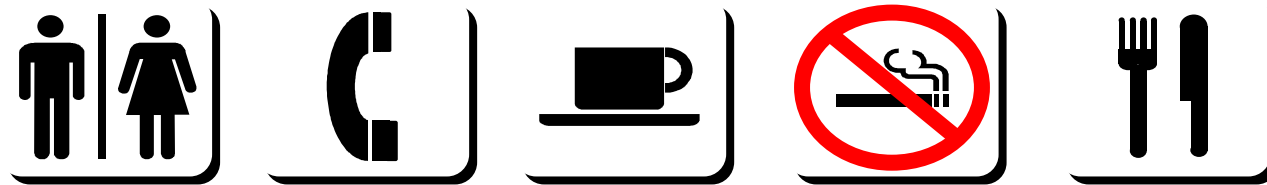
- Para construirmos modelos utilizamos de ferramentas (diagramas).
- Para Yourdon uma ferramenta deve ter as seguintes características:
 - ela deve ser gráfica, com adequado detalhamento textual de apoio;
 - ela deve permitir que o sistema possa ser visualizado de forma subdividida, na modalidade *top-down*;
 - ela deve ter mínima redundância;
 - ela deve ajudar o leitor a prognosticar o comportamento do sistema;
 - ela deve ser transparente para o leitor.

Diagramas e Documentação

- No contexto de desenvolvimento de software, diagramas correspondem a desenhos gráficos que seguem algum padrão lógico.
- Podemos também dizer que um diagrama é uma apresentação de uma coleção de elementos gráficos que possuem um significado predefinido.
- Diagramas normalmente são construídos de acordo com regras de notação bem definidas.
 - Ou seja, cada forma gráfica utilizada em um diagrama de modelagem tem um significado específico.

Diagramas e Documentação

- Diagramas permitem a construção de uma representação concisa de um sistema a ser construído.
 - “uma figura vale por mil palavras”



- No entanto, modelos também são compostos de informações textuais.
- Dado um modelo de uma das perspectivas de um sistema, diz-se que o seu diagrama, juntamente com a informação textual associada, formam a *documentação* deste modelo.

Modelagem de Software

- A modelagem de sistemas de software consiste na utilização de notações gráficas e textuais com o objetivo de construir modelos que representam as partes essenciais de um sistema, considerando-se diversas perspectivas diferentes e complementares.
(Eduardo Bezerra)

Objetivos da Modelagem de Sistemas

- Capturar os processos de negócio.
- Facilitar a comunicação entre as partes envolvidas (especialistas e analistas).
- Facilitar a gerência da complexidade do domínio permitindo exibir várias visões dos elementos do modelo.
- Definir a arquitetura lógica independente das possíveis implementações.
- Permitir a reutilização de esforços.

Importância/Vantagens da Modelagem

- Ajuda a visualizar um sistema como ele é, ou como se deseja que ele seja (predição do comportamento futuro do sistema). Auxilia no entendimento e definição do problema.
- Permite especificar o comportamento e a estrutura de um sistema, ou seja, a sua arquitetura.
- Oferece uma representação que serve como guia de construção do sistema.
- Documenta as decisões tomadas pela avaliação das alternativas propostas.
- Auxilia no gerenciamento do projeto e na avaliação de risco.
- Propicia a redução dos custos do desenvolvimento.

Considerações sobre a Modelagem de Sistemas

- A princípio, podemos ver a construção de modelos como uma atividade que atrasa o desenvolvimento do software propriamente dito.
- Mas essa atividade propicia alcançar as vantagens citadas anteriormente.
- Entretanto, note o fator **complexidade** como condicionante dessas vantagens.

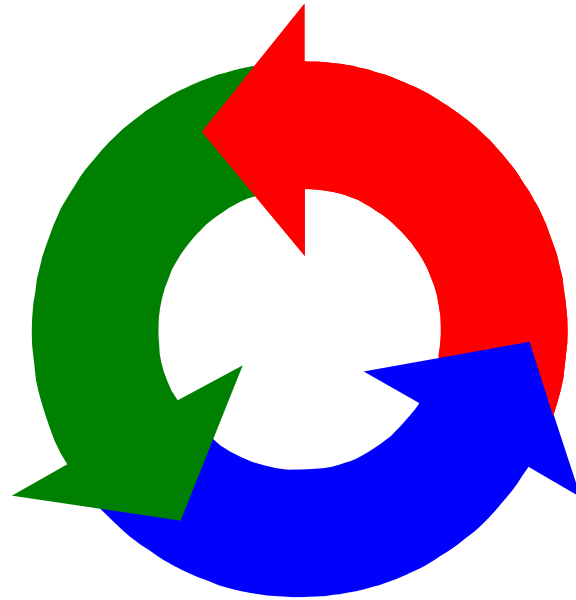
Ciclo de Vida, Processo e Metodologia

Análise e Projeto de Sistemas

Parte 03

Conteúdo

- Modelos de Ciclo de Vida
- Processo/Metodologia de Desenvolvimento de Sistemas
- Histórico das Técnicas de Desenvolvimento de Sistemas



Modelos de Ciclo de Vida

Ciclo de Vida do Desenvolvimento de Sistema

- É o encadeamento das fases para a construção de um sistema.
- Exemplo de modelos de ciclo de vida:
 - *modelo em cascata*
 - *modelo iterativo e incremental.*

Fases Típicas

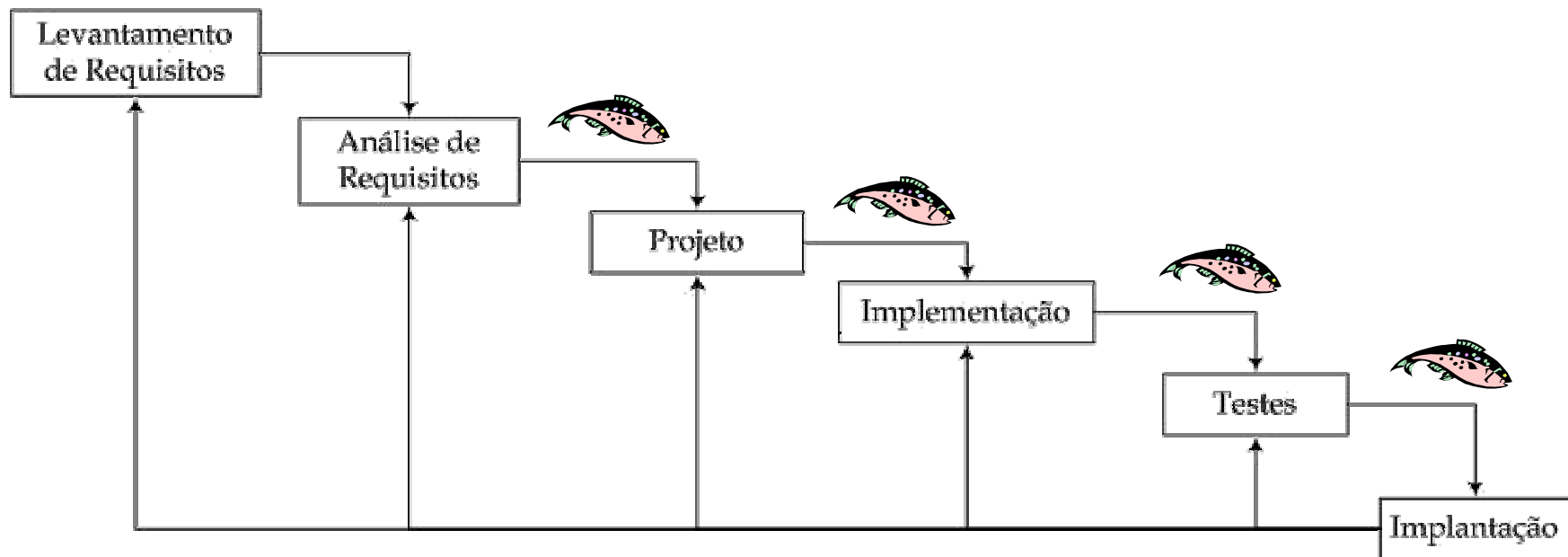
- **Análise**: fase na qual se determinarão quais os requisitos do sistema, “o que” o sistema deve fazer, em termos essenciais. Tem por objetivo interpretar e definir uma estrutura para um problema ainda não estruturado. Diz respeito a eficácia do sistema, ainda sem preocupações de performance.
- **Projeto**: fase na qual se determinará “como” o sistema funcionará para atender aos requisitos especificados na fase de análise. Diz respeito à eficiência do sistemas, já com preocupações de performance. O objetivo é modelar o sistema de modo a implementar a solução idealizada na fase de análise, mas de acordo com os recursos tecnológicos disponíveis.

Fases Típicas (cont.)

- **Implementação**: fase na qual será efetuada a construção do sistema de acordo com o modelo de funcionamento especificado na fase de projeto. Faz uso dos recursos tecnológicos disponíveis para atividades como a programação de computadores.

Modelo em Cascata

- Esse modelo apresenta uma tendência para a progressão sequencial entre uma fase e a seguinte.

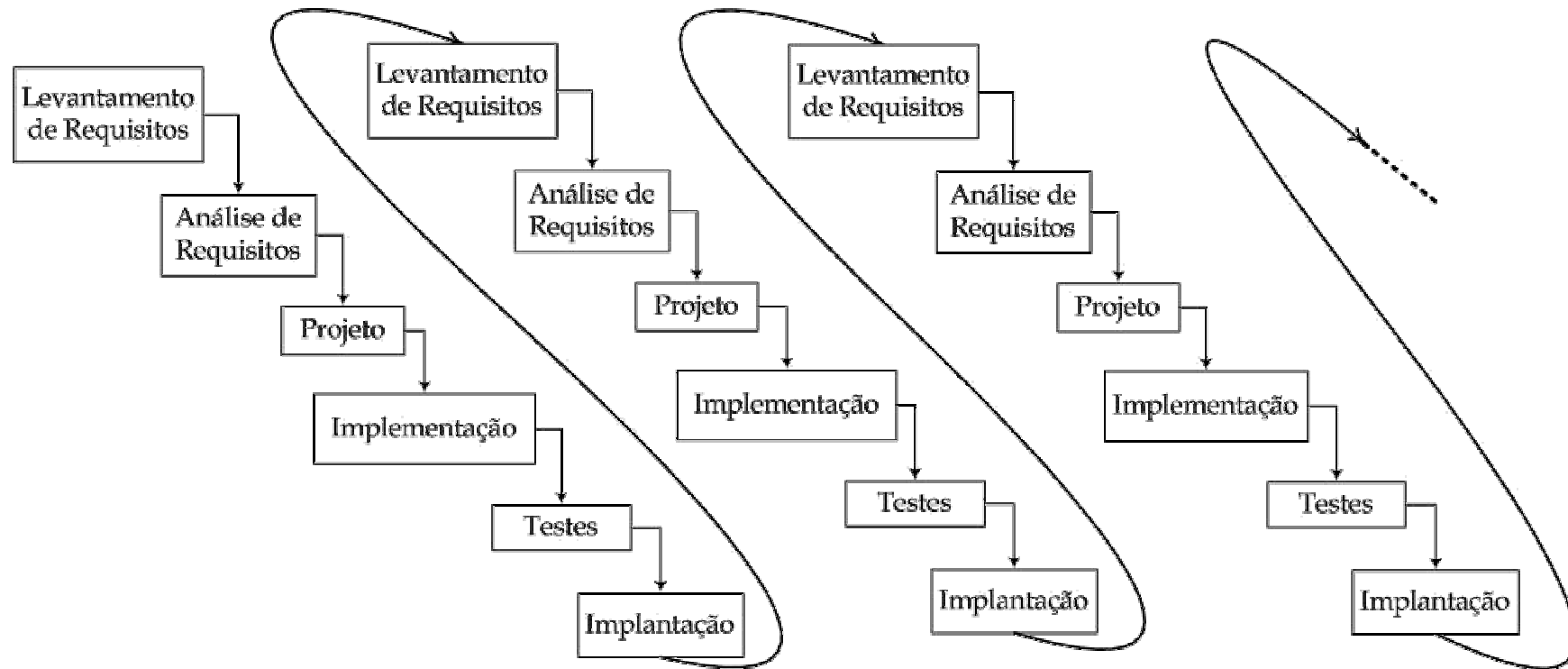


Modelo Cascata

- Projetos reais raramente seguem um fluxo seqüencial.
- Assume que é possível declarar detalhadamente todos os requisitos antes do início das demais fases do desenvolvimento.
 - propagação de erros pelas as fases do processo.
- Uma versão de produção do sistema não estará pronta até que o ciclo do projeto de desenvolvimento chegue ao final.

Modelo Iterativo e Incremental

- Desenvolvimento em mini-cascatas.

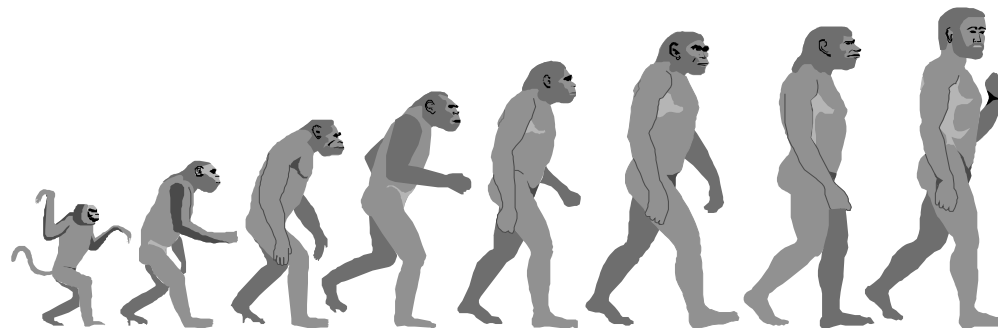


Modelo Iterativo e Incremental

- Divide o desenvolvimento de um produto de software em *ciclos*.
- Em cada ciclo de desenvolvimento, podem ser identificadas as fases de análise, projeto, implementação e testes.
- Cada ciclo considera um subconjunto de requisitos.
- Esta característica contrasta com a abordagem clássica, na qual as fases são realizadas uma única vez.

Modelo Iterativo e Incremental

- ***Iterativo***: o sistema de software é desenvolvido em vários passos similares.
- ***Incremental***: Em cada passo, o sistema é estendido com mais funcionalidades.



Modelo Iterativo e Incremental

- Vantagens e Desvantagens -

- 👍 Incentiva a participação do usuário.
- 👍 *Riscos* do desenvolvimento podem ser mais bem gerenciados.
 - Um *risco de projeto* é a possibilidade de ocorrência de algum evento que cause prejuízo ao processo de desenvolvimento, juntamente com as consequências desse prejuízo.
 - Influências: custos do projeto, cronograma, qualidade do produto, satisfação do cliente, etc.
- ☹ Mais difícil de gerenciar

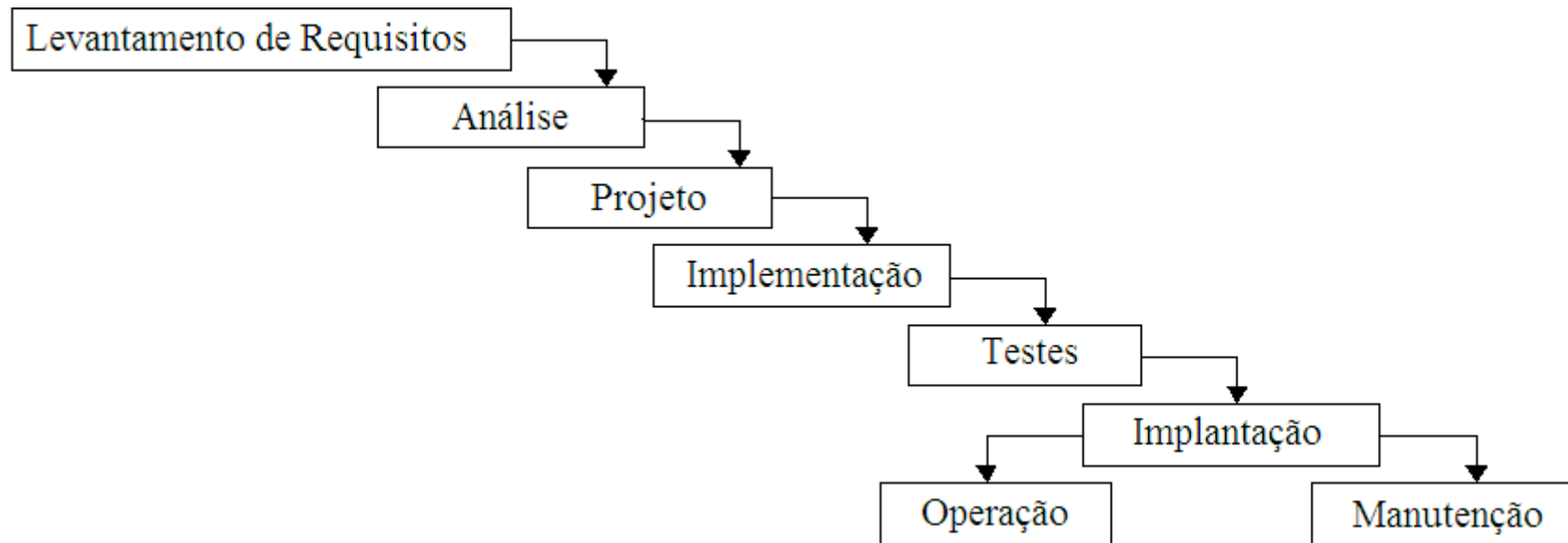
Ataque os riscos

- “*Se você não atacar os riscos [do projeto] ativamente, então estes irão ativamente atacar você.*” (Tom Gilb).
 - A maioria dos Processos de Desenvolvimento de Software que seguem o modelo iterativo e incremental aconselha que as partes mais arriscadas sejam consideradas inicialmente.

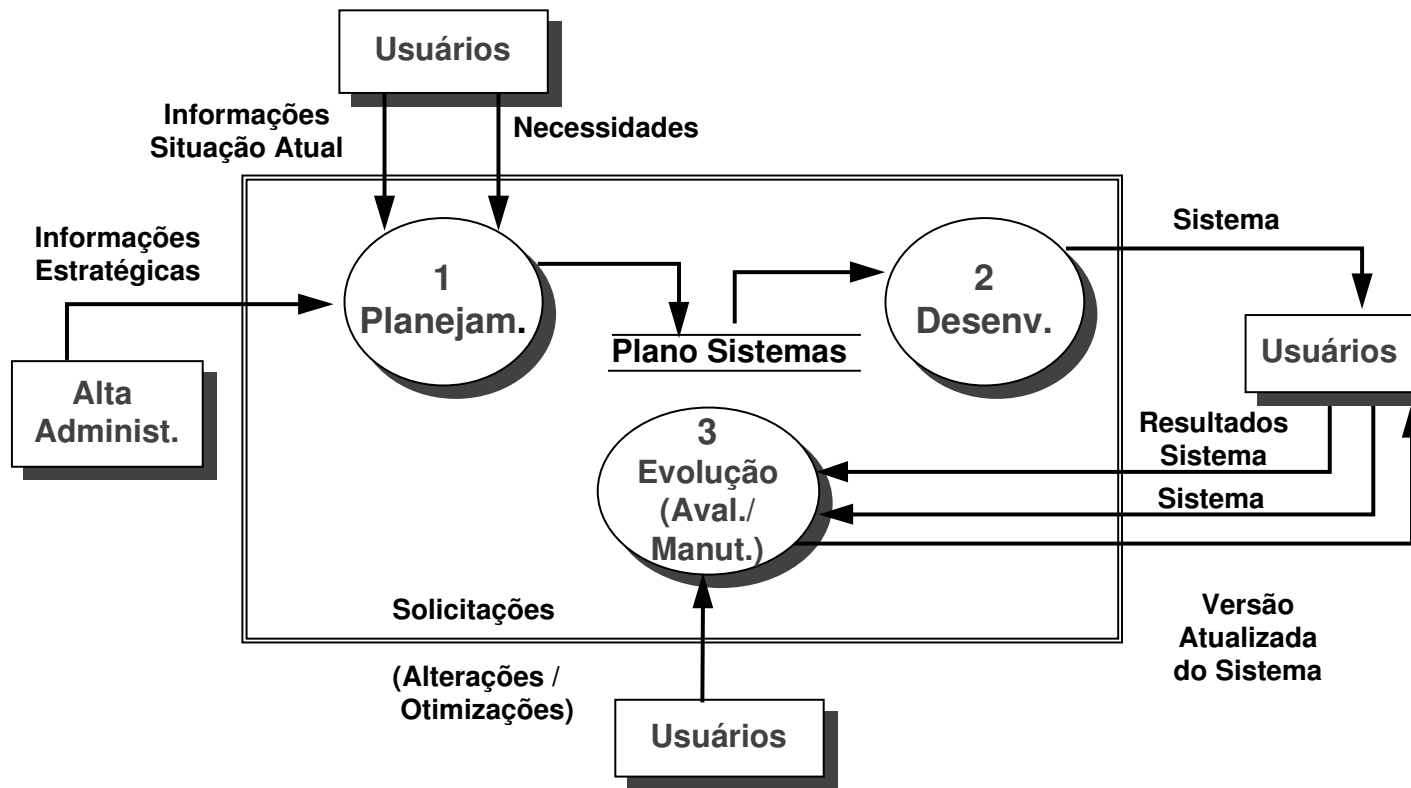


Ciclo de Vida do Sistema

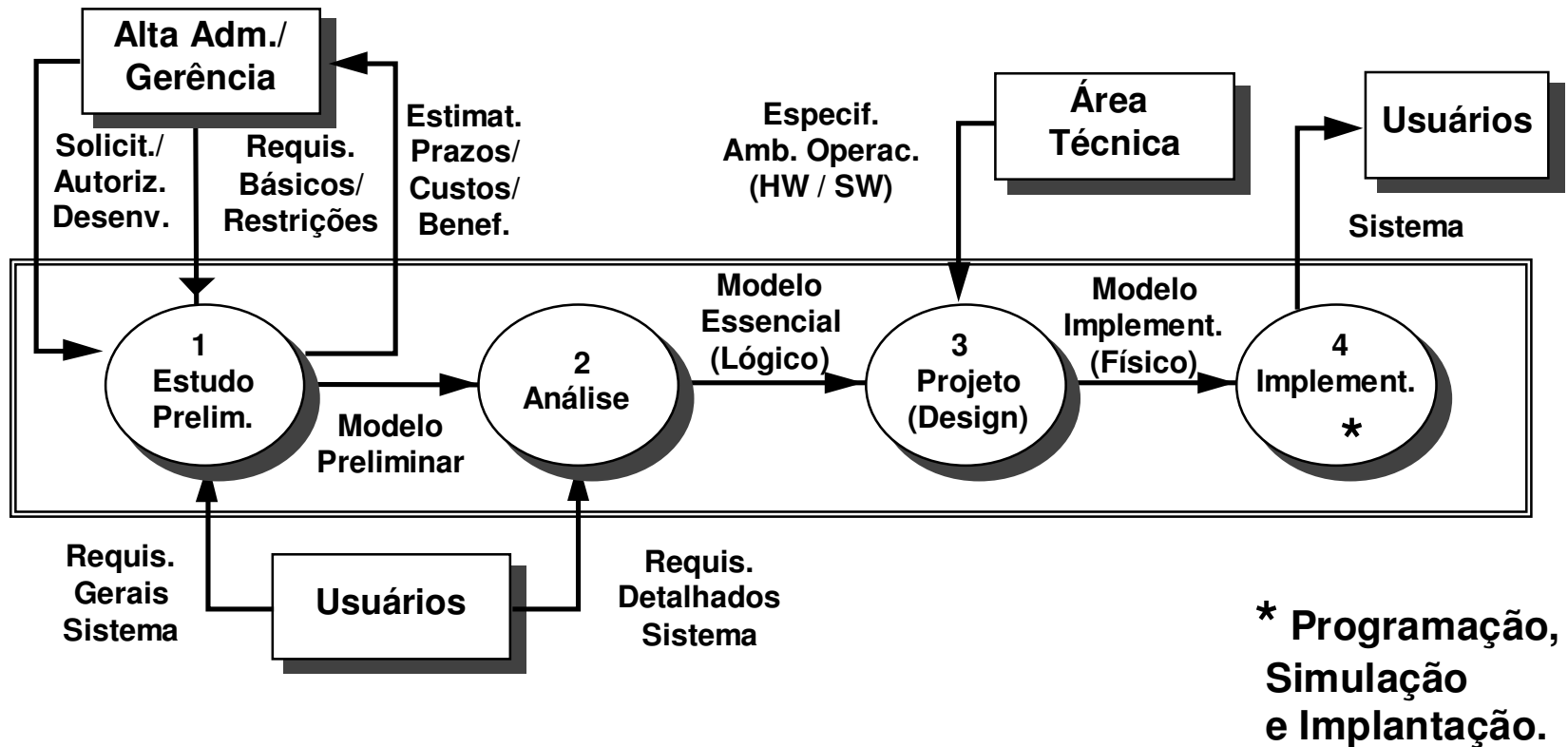
- O ciclo de atividades que inclui, além das fases de desenvolvimento, a operação e a manutenção de um sistema.



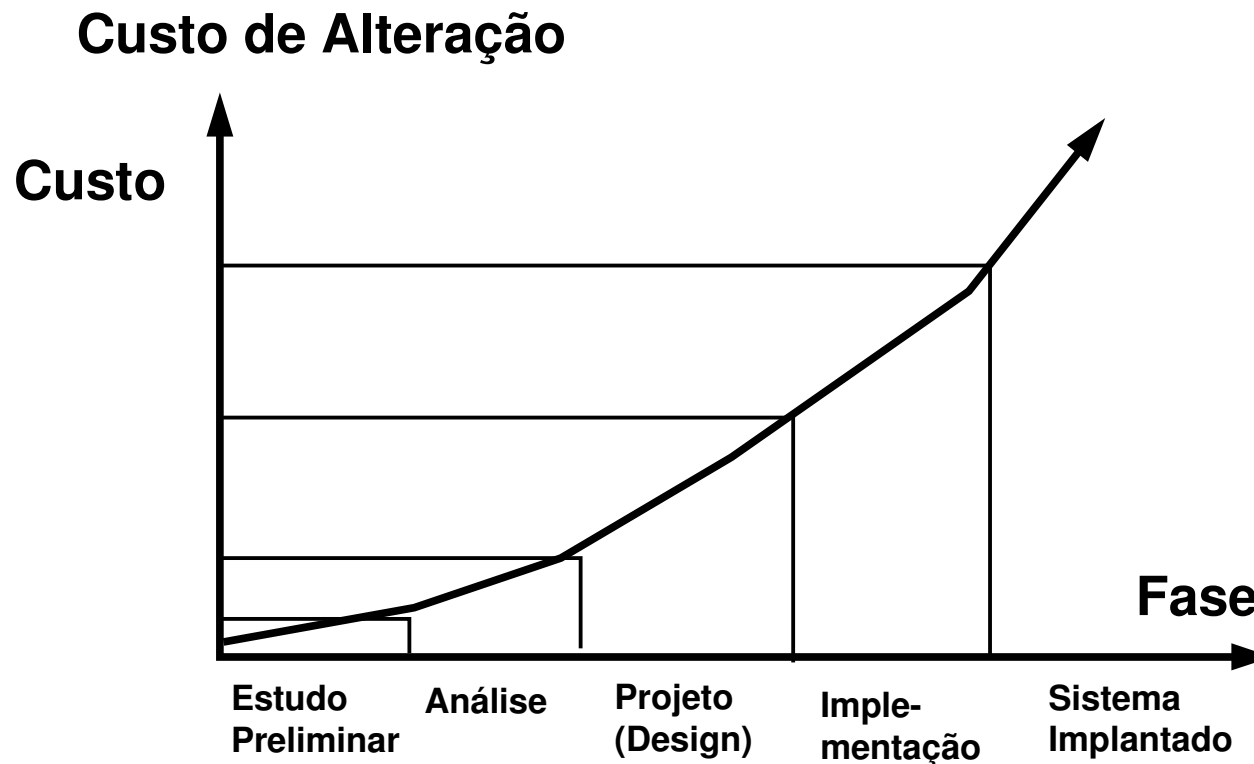
Visão Geral do Ciclo de Vida de um Sistema



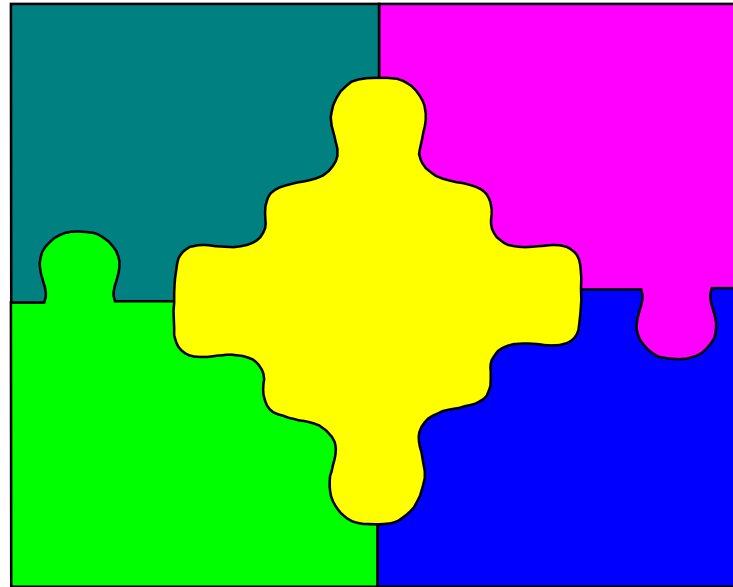
Processo Desenvolvimento



Benefício da Modelagem associado às fases do Ciclo de Vida do Sistema



Quanto mais tarde se detecta um erro ou se identifica uma necessidade de mudança, mais difícil e cara é a alteração.



Processo/Metodologia de Desenvolvimento de Sistemas

Introdução

- Tentativas de lidar com a complexidade e de minimizar os problemas envolvidos no desenvolvimento de software envolvem a definição de *processos (ou metodologias) de desenvolvimento de software*.
- “O uso de metodologias que proponham a modelagem dos sistemas é a melhor maneira de se resolver os problemas da atividade de desenvolvimento.” (S.Pompilho)
- **Obs.:** Edward. Yourdon utiliza o termo ciclo de vida do projeto.

Conceitos

- Um processo de desenvolvimento de software (PDS) compreende todas as atividades necessárias para definir, desenvolver, testar e manter um produto de software. (Eduardo Bezerra)

processo de desenvolvimento de software

- “... o ciclo de vida do projeto documentado oferece um modo simples para qualquer pessoa da organização de desenvolvimento de sistemas entrosar-se com a atividade de desenvolvimento de um sistema de processamento.” (Edward Yourdon)

Objetivos de um PDS

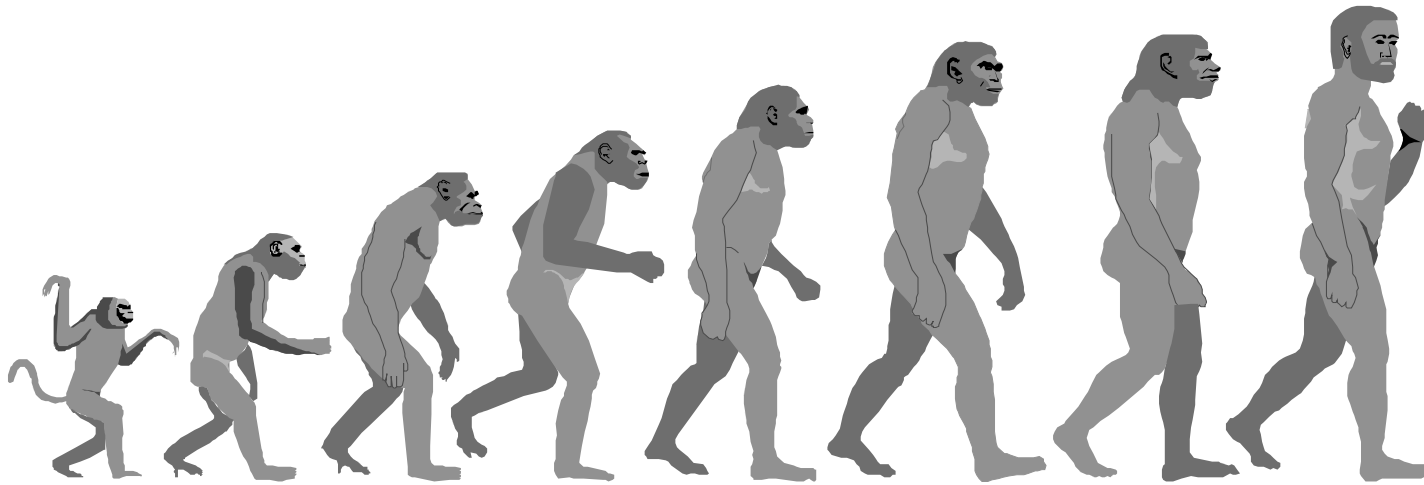
- Definir *quais* as atividades a serem executadas ao longo do desenvolvimento:
 - definir quais as fases de trabalho previstas no desenvolvimento de sistemas, ou seja, qual o modelo de ciclo de vida no qual se baseia;
 - para cada fase, quais as técnicas adotadas (Análise Estruturada, Análise Essencial, Projeto Estruturado etc.);
 - para cada técnica adotada definir as ferramentas a serem utilizadas (Diagrama de Fluxo de Dados, Diagrama Entidade-Relacionamento, Diagrama de Transição de Estado etc.);
 - definir quais os modelos que as ferramentas irão produzir (Modelo Funcional, Modelo Conceitual de Dados, Modelo de Controle etc.).
- Definir *quando, como e por quem* tais atividades serão executadas.
- Prover pontos de controle para verificar o andamento do desenvolvimento.
- Padronizar a forma de desenvolver software em uma organização.

Atividades Típicas de um PDS

- Estudo (Levantamento de requisitos)
- Análise
- Projeto
- Implementação
- Testes
- Implantação

Processo X Metodologia de DS

- Como vimos até aqui, os termos “metodologia” e “processo” de desenvolvimento de sistemas, muitas vezes, são utilizados indistintamente na literatura.
- Contudo, há quem diferencie tais termos com as seguintes definições:
 - Processo de Desenvolvimento de Sistemas corresponde aos conceitos que verificamos nas transparências anteriores. Porém, apresenta-se como uma referência, em um nível quase conceitual.
 - Metodologia de Desenvolvimento de Sistemas é a adequação (um filtro, um subconjunto) de um determinado PDS à realidade de uma determinada equipe ou empresa de desenvolvimento de sistemas.



Histórico das Técnicas de Desenvolvimento de Sistemas

Evolução do Software

- O rápido crescimento da capacidade computacional das máquinas resultou na demanda por sistemas de software cada vez mais complexos.
- O surgimento de sistemas de software mais complexos resultou na necessidade de reavaliação da forma de se desenvolver sistemas.
- Consequentemente as técnicas utilizadas para a construção de sistemas computacionais têm evoluído de forma impressionante, notavelmente no que tange à modelagem de sistemas.

Evolução das Técnicas de Modelagem de Sistemas

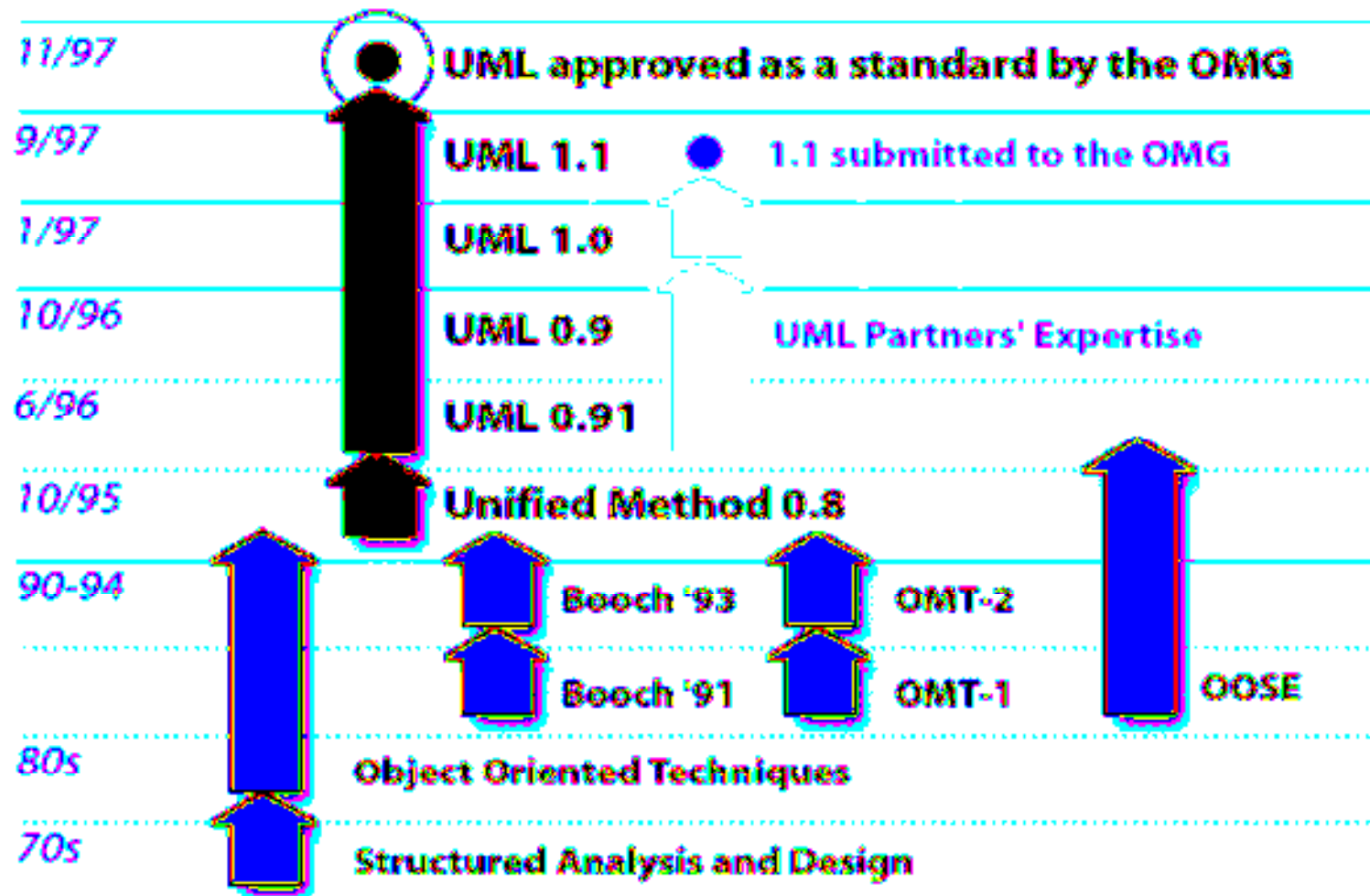
| Período | Técnica | Abordagem | Características | Ferramentas |
|------------------------------|--|--------------------------------|--|---|
| Décadas de 1950/60 | Análise tradicional / técnicas simples | Funcional | <ul style="list-style-type: none"> - Sistemas simples - Desenvolvimento <i>ad hoc</i> | Textos Fluxogramas Diagramas de Módulos |
| Década de 1970 | Análise Estruturada Projeto Estruturado | Funcional Dados | <ul style="list-style-type: none"> - Surgem sistemas mais complexos - Surge a programação estruturada | Diagrama de Fluxo de Dados Diagrama de Estrutura de Dados Miniespecificações Normalização Dicionário de Dados |
| Década de 1980 | Análise Essencial Análise Estruturada Moderna | Funcional Dados Controle | <ul style="list-style-type: none"> - Interfaces homem-máquina mais sofisticadas - Necessidades de sistemas mais complexos | Tabela de Eventos Diagrama de Fluxo de Dados Diagrama Entidade-Relacionamento Diagrama de Transição de Estado Diagrama de Estrutura de Dados Miniespecificações Normalização Dicionário de Dados |
| Início da década de 1990 | Análise Orientada a Objetos | Orientada a Objetos | <ul style="list-style-type: none"> - OO apresenta-se em setores pontuais - Pouca maturidade da abordagem OO - Várias propostas técnicas de modelagem OO | Várias iniciativas propondo ferramentas |
| Fim da década de 1990 / 2000 | Análise e Projeto Orientados a Objetos | Orientada a Objetos | - Maturidade do paradigma OO | Unified Modeling Language (UML) |

Evolução da Análise Estruturada

- **1977 a 1979: Análise Estruturada** (clássica) (Tom DeMarco) / (Chris Gane e Trish Sarson)
- **1984: Análise Essencial** (McMenamim e Palmer)
- **1989: Análise Estruturada Moderna** (Edward Yourdon)

Evolução da UML

Evolution of the Unified Modeling Language



Evolução da UML

