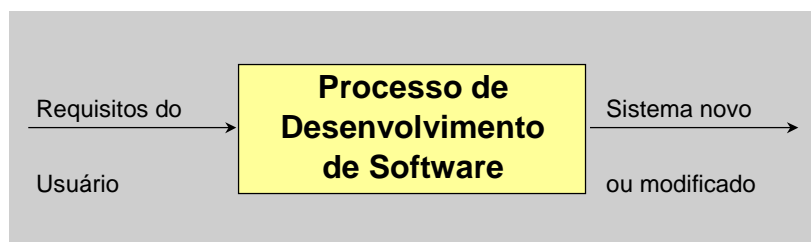


Processo Desenvolvimento de Software



- “Um processo é o conjunto total de atividades de engenharia necessárias para transformar requisitos do usuário em software”

“Managing the Process”, Humphrey, 1989

Leite, 2004

Processo de Software - definições

- “Um **processo de software** é um método para desenvolver ou produzir software [Tyrrell, 2001].
- A pesquisa em **processo de software** lida com métodos e tecnologias estimativas, suporte e melhoria das atividades de desenvolvimento de software [Fuggetta, 2000]
- Define quem faz o *que*, *quando* e *como*.

Modelo X Processo

- Um modelo é algo teórico, um conjunto de possíveis ações.
- O processo deve determinar ações práticas a serem realizadas pela equipe como prazos definidos e métricas para se avaliar como elas estão sendo realizadas

Modelo + Planejamento = Processo

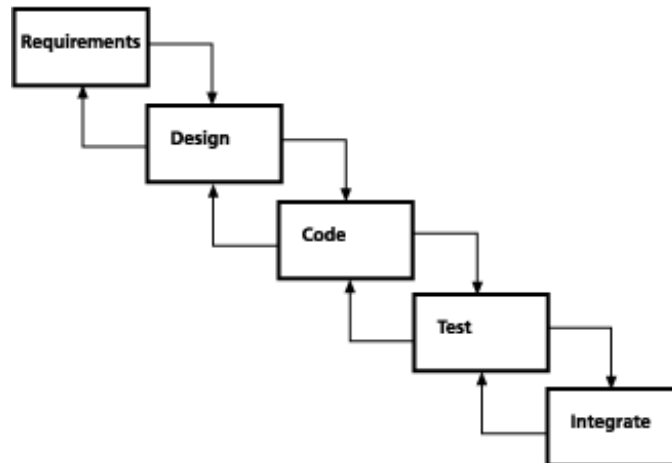
Engenharia de Software, Jair C Leite, 2004

Modelos de Processo

- Um modelo de processo ou método define um conjunto de atividades específicas.
- Principais modelos:
 - Cascata (Waterfall)
 - Espiral (Spiral)
 - Evolutivo e Incremental
 - Processo Unificado da Rational (Rational Unified Process – RUP)
 - Personal Software Process
 - Team Software Process

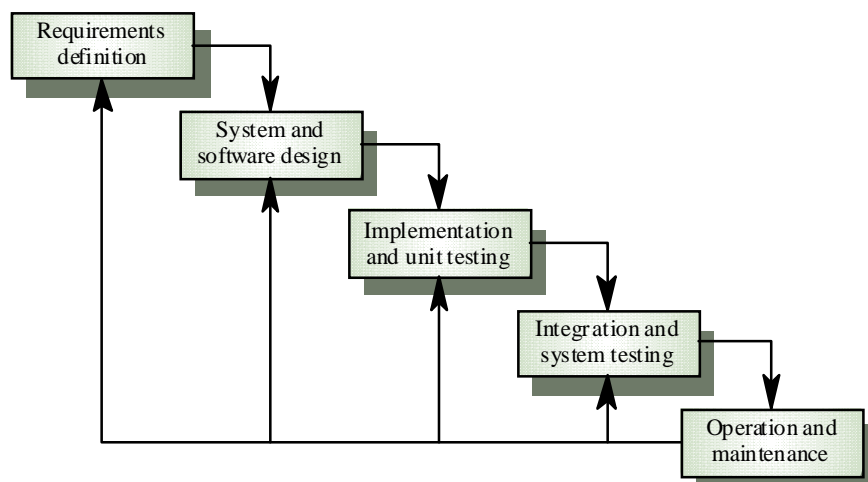
Engenharia de Software, Jair C Leite, 2004

Modelo Cascata (Waterfall)



Engenharia de Software, Jair C Leite, 2004

Modelo Cascata



Fonte: Ian Sommerville

Engenharia de Software, Jair C Leite, 2004

Atividades típicas do modelo Cascata

- Análise e definição de requisitos
 - Objetivos, funções e restrições são definidos, com ajuda de clientes e usuários, e servem como uma especificação do sistema, indicando o que deve ser implementado.
- Design de sistemas e software
 - Envolve a descrição do sistema e do software em termos de unidades abstratas e de suas relações, indicando como o software deve ser implementado.
- Implementação e testes de unidade
 - As unidades do software devem ser codificadas e testadas individualmente.
- Integração e testes de sistema
 - As unidades são integradas e testadas
- Entrega, operação e manutenção
 - O sistema é instalado e colocado em operação. A manutenção envolve a correção de erros e evolução do sistema para atender a novos requisitos.

Engenharia de Software, Jair C Leite, 2004

Características do modelo Cascata

- Divisão inflexível do projeto em estágios distintos. A fase seguinte só deve iniciar quando a anterior tiver sido concluída e aprovada pelas partes envolvidas.
- Por exemplo, o design apenas deve começar quando os requisitos estiverem totalmente definidos e aprovados
- Dificuldade em realizar mudanças com o processo em andamento – requisitos sempre mudam.
- O modelo em cascata é apropriado quando se tem um entendimento claro dos requisitos.
- Oferece maior previsibilidade de prazos e custo: melhor planejamento e gerenciamento.
- A engenharia do sistema segue este tipo de modelo.

Engenharia de Software, Jair C Leite, 2004

Prototipação

- Abordagem baseada numa visão evolutiva do desenvolvimento de software, afetando o processo como um todo
- Protótipo de software é um sistema que...
 - funciona
 - não tem tempo de vida definido
 - pode servir a múltiplos propósitos
 - deve ser construído rapidamente e com baixo custo
 - é parte integrante de um design centrado no usuário, para avaliação e modificação

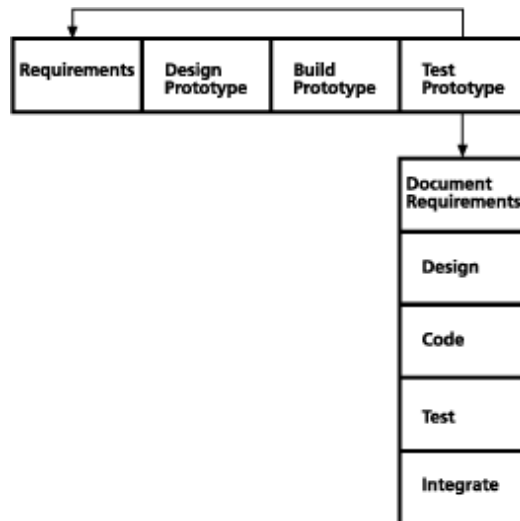
Engenharia de Software, Jair C Leite, 2004

Tipos e objetivos dos protótipos

- Tipos
 - Falso
 - Autêntico
 - Funcional
 - Sistema piloto
- Objetivos
 - Exploração
 - Experimentação
 - Evolução

Engenharia de Software, Jair C Leite, 2004

Prototipação



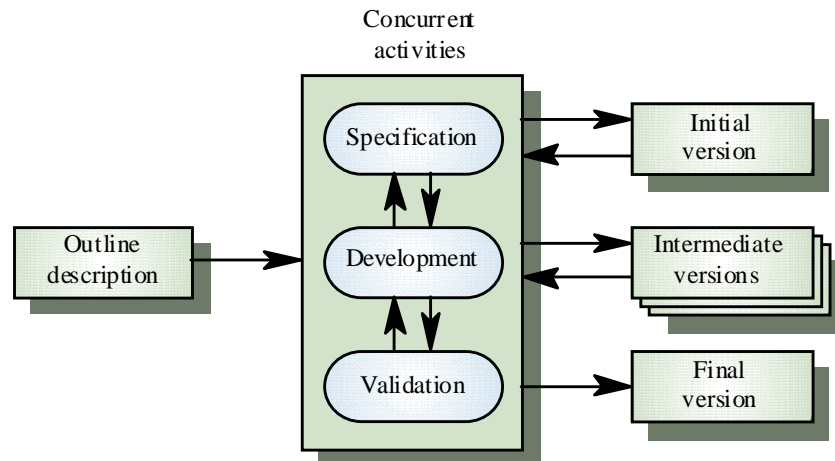
Engenharia de Software, Jair C Leite, 2004

Desenvolvimento Evolutivo

- Desenvolvimento Exploratório
 - A partir de requisitos iniciais, é elaborado um protótipo que permite, junto ao cliente, explorar novos requisitos.
- Prototipagem
 - Protótipo descartáveis
 - Protótipo evolutivos - evolui para o produto final
- Características
 - Útil quando os requisitos estão obscuros
 - Especificação é construída gradativamente
 - Possibilitam um rápido desenvolvimento da aplicação
 - Requer ferramentas específicas
 - Os sistemas são freqüentemente mal-estruturados e mal-documentados.
 - Processo não é claro, dificuldade de planejamento e gerenciamento

Engenharia de Software, Jair C Leite, 2004

Desenvolvimento Evolutivo

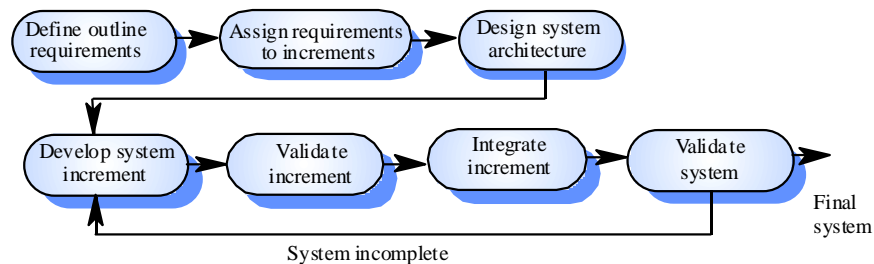


Fonte: Ian Sommerville

Engenharia de Software, Jair C Leite, 2004

Desenvolvimento incremental

- Desenvolvimento iterativo, em ciclos que permitem revisões de atividades anteriores.
- O sistema é particionado em partes independentes que podem ser entregues à medida que forem ficando prontas e avaliadas.
- A arquitetura do sistema deve possibilitar a entrega das partes independentes.



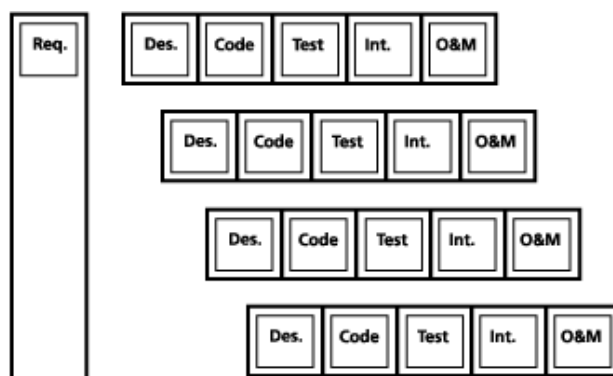
Engenharia de Software, Jair C Leite, 2004

Características do modelo incremental

- Os incrementos funcionam como protótipos do sistema.
- A avaliação do cliente pode ser feita a partir das experiências do usuário com as entregas parciais.
- Novos requisitos podem ser incorporados aos outros incrementos do sistema.
- Deve-se identificar funcionalidade prioritárias que serão desenvolvidas primeiro.
- Os processos de cada incremento podem ser independentes. Pode-se utilizar Cascata.
- Risco menor de fracasso completo do sistema.
- As funções entregues primeiro são testadas mais vezes, à medida que os incrementos são entregues.

Engenharia de Software, Jair C Leite, 2004

Incremental



Engenharia de Software, Jair C Leite, 2004

Programação eXtrema (XP)

- Abordagem baseada no desenvolvimento e entrega de pequenas partes da funcionalidade do software.
- As partes devem ser incrementadas e requer a melhoria constante do código (re-trabalho).
- Requer o envolvimento constante do cliente
- Os sistema é concebido a partir de uma metáfora e descritos em *estórias do usuário*.
- É necessário a definição de testes de aceitação.
- Não existe um processo de design tradicional e não são gerados modelos da arquitetura do software
- A programação é feita por pares de programadores. Prática do codifica-e-conserta.

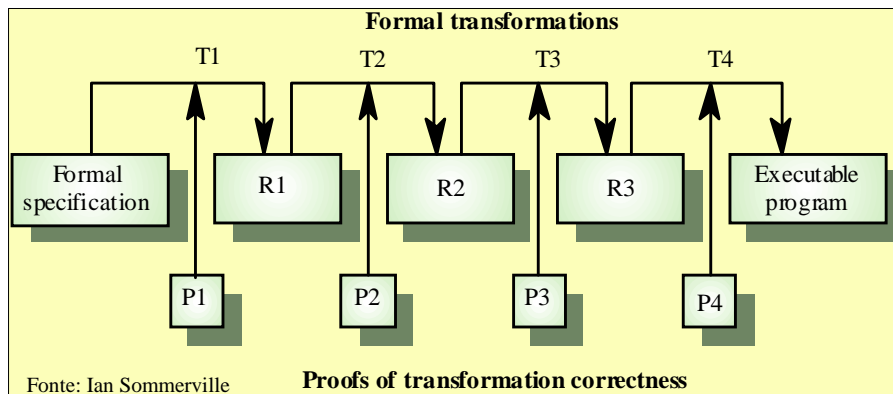
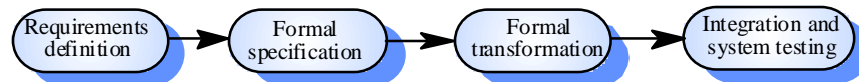
Engenharia de Software, Jair C Leite, 2004

Desenvolvimento formal de sistemas

- Baseado na transformação de uma série representações matemáticas da especificação em um programa executável.
- A especificação de requisitos, informal ou semi-formal, é redefinida em uma especificação formal detalhada, utilizando uma notação matemática:
 - VDM, Z, B, Larch e outras
- As atividades de design, implementação e testes são substituídas por um processo transformacional.
- As transformações devem ser matematicamente corretas, baseadas em provas-de-correção.
- As provas são longas e impraticáveis para sistemas de grande porte.
- É adequado a sistemas com exigências rigorosas de segurança, confiança e garantia.

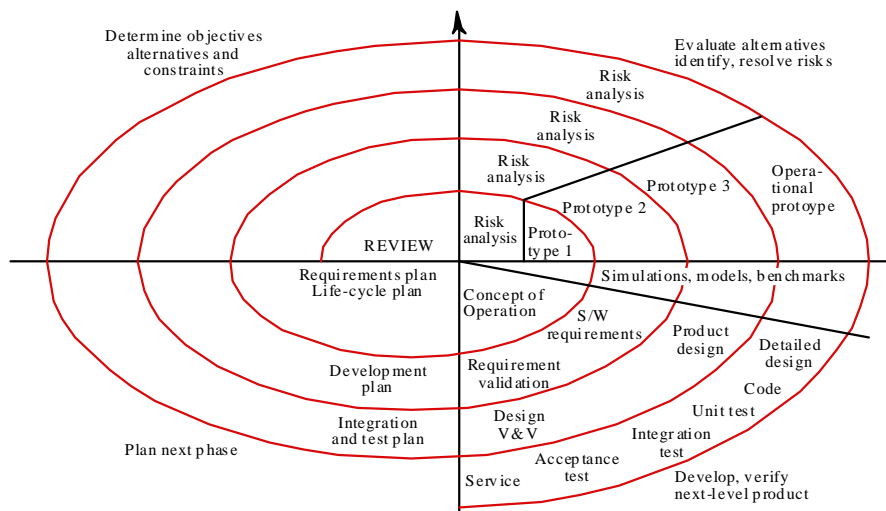
Engenharia de Software, Jair C Leite, 2004

Transformações formais



Engenharia de Software, Jair C Leite, 2004

Modelo Espiral (Spiral)



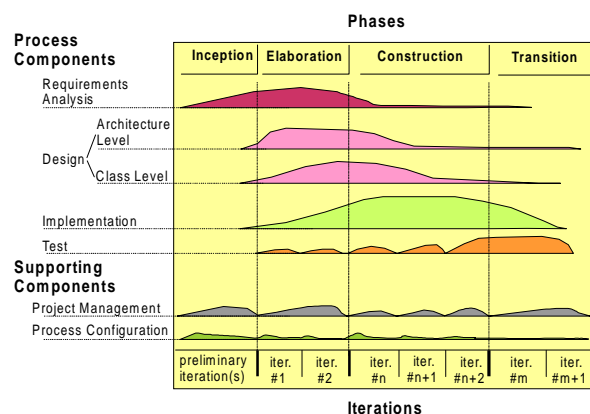
Engenharia de Software, Jair C Leite, 2004

Modelo Espiral

- O processo é representado como uma espiral. Cada ciclo do espiral é uma fase do processo
- Cada ciclo determina quatro etapas fundamentais:
 - Definição de objetivos, alternativas e restrições
 - Análise e redução de riscos
 - Desenvolvimento e validação
 - Planejamento do próximo ciclo
- Protótipos são construídos em cada ciclo.
- Não há fases fixas pré-definidas. Elas são definidas de acordo com os objetivos.
- É um meta-modelo: qualquer modelo pode ser derivado a partir do modelo espiral

Engenharia de Software, Jair C Leite, 2004

Rational Unified Process



Engenharia de Software, Jair C Leite, 2004