

UNIVERSIDADE FEDERAL DO ACRE
CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGICAS
BACHARELADO EM SISTEMAS DE INFORMAÇÃO

ENGENHARIA DE SOFTWARE I
Daricélio Moreira Soares

Capítulo 3
Processos de Software



PROCESSOS DE SOFTWARE

- Conjuntos de atividades coerentes para especificar, projetar, implementar e testar sistemas de software



O PROCESSO DE SOFTWARE

- Um conjunto estruturado de atividades requeridas para desenvolver um sistema de software
 - Especificação
 - Projeto
 - Validação
 - Evolução
- Um modelo de processo de software é uma representação abstrata de um processo. Apresenta uma descrição de um processo de alguma perspectiva particular

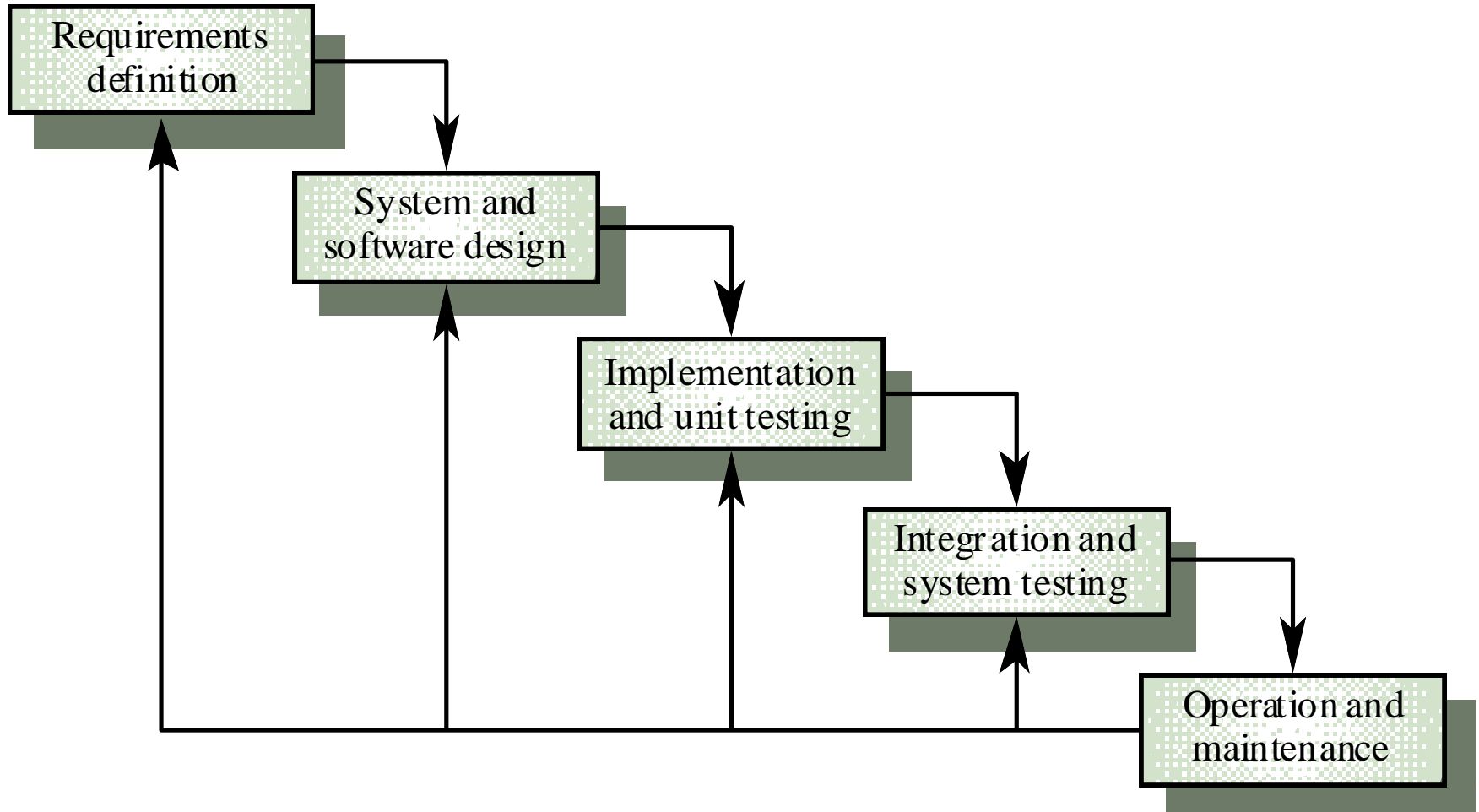


MODELOS GENÉRICOS DE PROCESSO DE SOFTWARE

- O modelo cascata
 - Separa e distingue fases de especificação e desenvolvimento
- Desenvolvimento evolucionário
 - Especificação e desenvolvimento são entrelaçados
- Desenvolvimento Formal de sistemas
 - Um modelo de sistema matemático é formalmente transformado para uma implementação
- Desenvolvimento baseado na reutilização
 - O sistema é montado a partir de componentes existentes



MODELO CASCATA



FASES DO MODELO CASCATA

- Análise e definição de requisitos
- Projeto do sistema e do software
- Implementação e teste da unidade
- Integração e teste do sistema
- Operação e manutenção
- A desvantagem do modelo cascata é a dificuldade de acomodar mudanças depois que o processo está em andamento



PROBLEMAS DO MODELO CASCATA

- Partição inflexível do projeto em diferentes estágios
- Isto faz com que seja difícil responder aos requisitos mutáveis dos clientes
- Portanto, este modelo só é apropriado quando os requisitos são bem entendidos



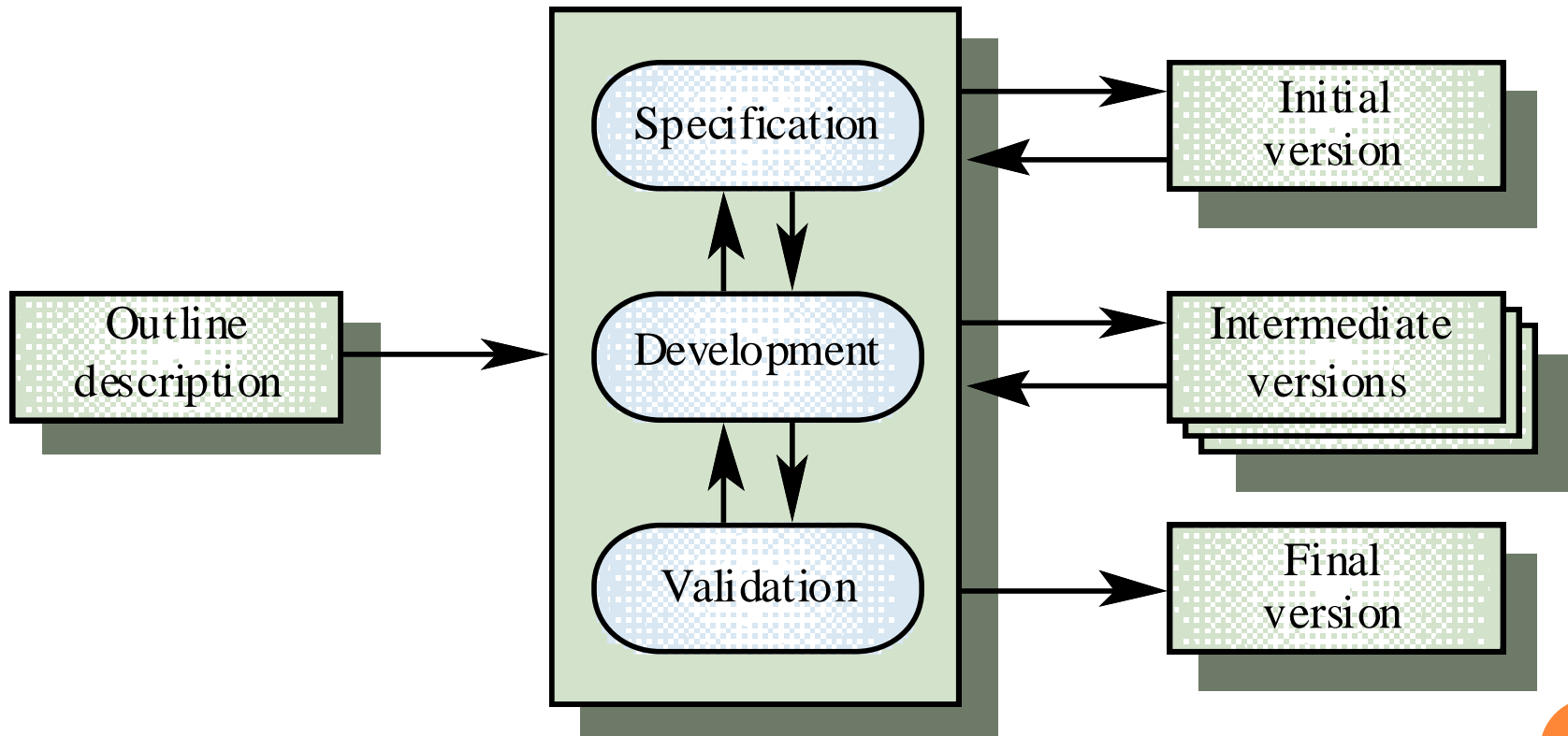
DESENVOLVIMENTO EVOLUCIONÁRIO

- Desenvolvimento exploratório
 - O objetivo é trabalhar com clientes e evoluir o sistema final de um esboço de especificação inicial. Deve começar com os requisitos que estão bem entendidos
- Preparação de protótipos descartáveis
 - Objetivo é entender os requisitos do sistema. Deve começar com requisitos pobremente entendidos



DESENVOLVIMENTO EVOLUCIONÁRIO

Concurrent
activities



DESENVOLVIMENTO EVOLUCIONÁRIO

○ Problemas

- Falta de visibilidade do processo
- Sistemas são, em geral, pobremente estruturados
- Habilidades especiais (ex. em línguas para rápida preparação de protótipos) podem ser requeridas

○ Aplicabilidade

- Para sistemas interativos pequenos ou médios
- Para partes de sistemas grandes (ex. a interface de usuário)
- Para sistemas de curto-prazo

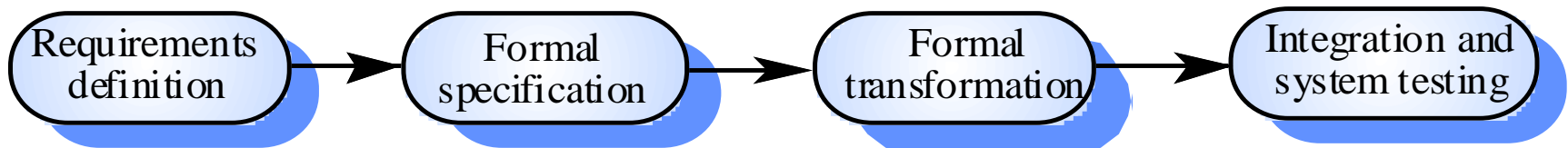


DESENVOLVIMENTO DE SISTEMAS FORMAIS

- Baseado na transformação de uma especificação matemática através de diferentes representações para um programa executável
- Transformações são 'preservadoras de exatidão', portanto, são diretas para mostrar que o programa está de acordo com sua especificação
- Contido na abordagem 'Cleanroom' para desenvolvimento de software



DESENVOLVIMENTO DE SISTEMAS FORMAIS



DESENVOLVIMENTO DE SISTEMAS FORMAIS

○ Problemas

- Necessidade de habilidades especializadas e treinamento para aplicar a técnica
- Difícil de especificar formalmente alguns aspectos do sistema como a interface de usuário

○ Aplicabilidade

- Sistemas críticos, especialmente aqueles no qual um case de segurança deve ser feito antes do sistema ser posto em operação

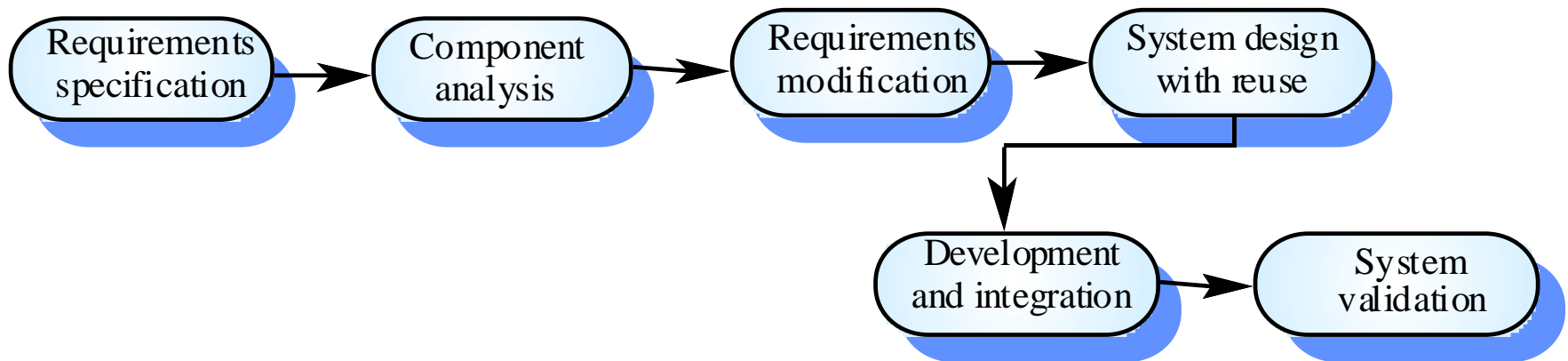


DESENVOLVIMENTO ORIENTADO AO REUSO

- Baseado no reuso sistemático, onde os sistemas são integrados de componentes existentes ou sistemas padronizados
- Estágios do Processo
 - Análise do componente
 - Modificação dos requisitos
 - Projeto do sistema com reuso
 - Desenvolvimento e integração
- Esta abordagem está se tornando mais importante, mas a experiência ainda é limitada com ela



DESENVOLVIMENTO ORIENTADO AO REUSO



ITERAÇÃO DO PROCESSO

- Requisitos do sistema SEMPRE evoluem no decorrer de um projeto, então a iteração do processo, onde estágios anteriores são re-trabalhados, é sempre parte de um processo para sistemas maiores
- Iteração pode ser aplicada para qualquer modelo de processo genérico
- Duas abordagens (relacionadas)
 - Desenvolvimento incremental
 - Desenvolvimento espiral

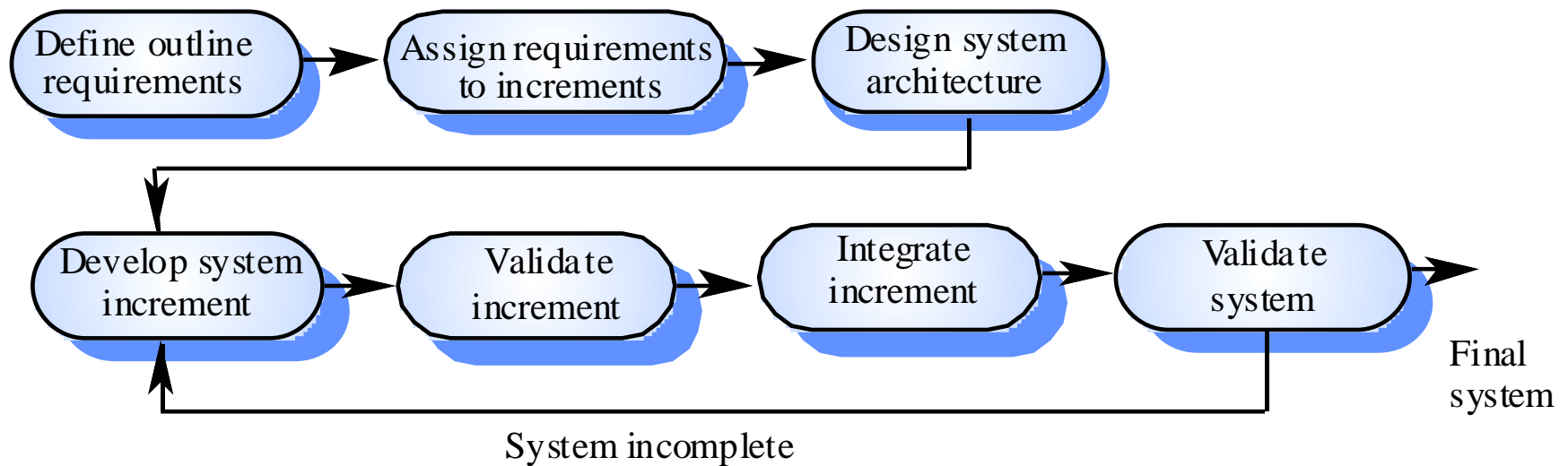


DESENVOLVIMENTO INCREMENTAL

- Ao invés de entregar o sistema de uma única vez, o desenvolvimento e a entrega é dividida em incrementos com cada incremento entregando parte da funcionalidade requerida
- Os requisitos dos usuários são priorizados e os requisitos de maior prioridade são incluídos em incrementos iniciais
- Uma vez que o desenvolvimento de um incremento é iniciado, os requisitos são congelados embora requisitos para incrementos posteriores possam continuar a evoluir



DESENVOLVIMENTO INCREMENTAL



VANTAGENS DO DESENVOLVIMENTO INCREMENTAL

- O valor agregado ao Cliente está na entrega em cada incremento de modo que a funcionalidade do sistema estará disponível mais cedo
- Incrementos iniciais funcionam como protótipos para ajudar a evocar requisitos para incrementos posteriores
- Menores riscos de falha no projeto em geral
- Os serviços do sistema de alta prioridade tendem a receber a maioria dos testes



PROGRAMAÇÃO EXTREMA

- Nova abordagem para o desenvolvimento de software baseado no desenvolvimento e entrega de incrementos de funcionalidade bem pequenos
- Conta com melhoramento constante do código, envolvimento do usuário no time de desenvolvimento e programação em pares

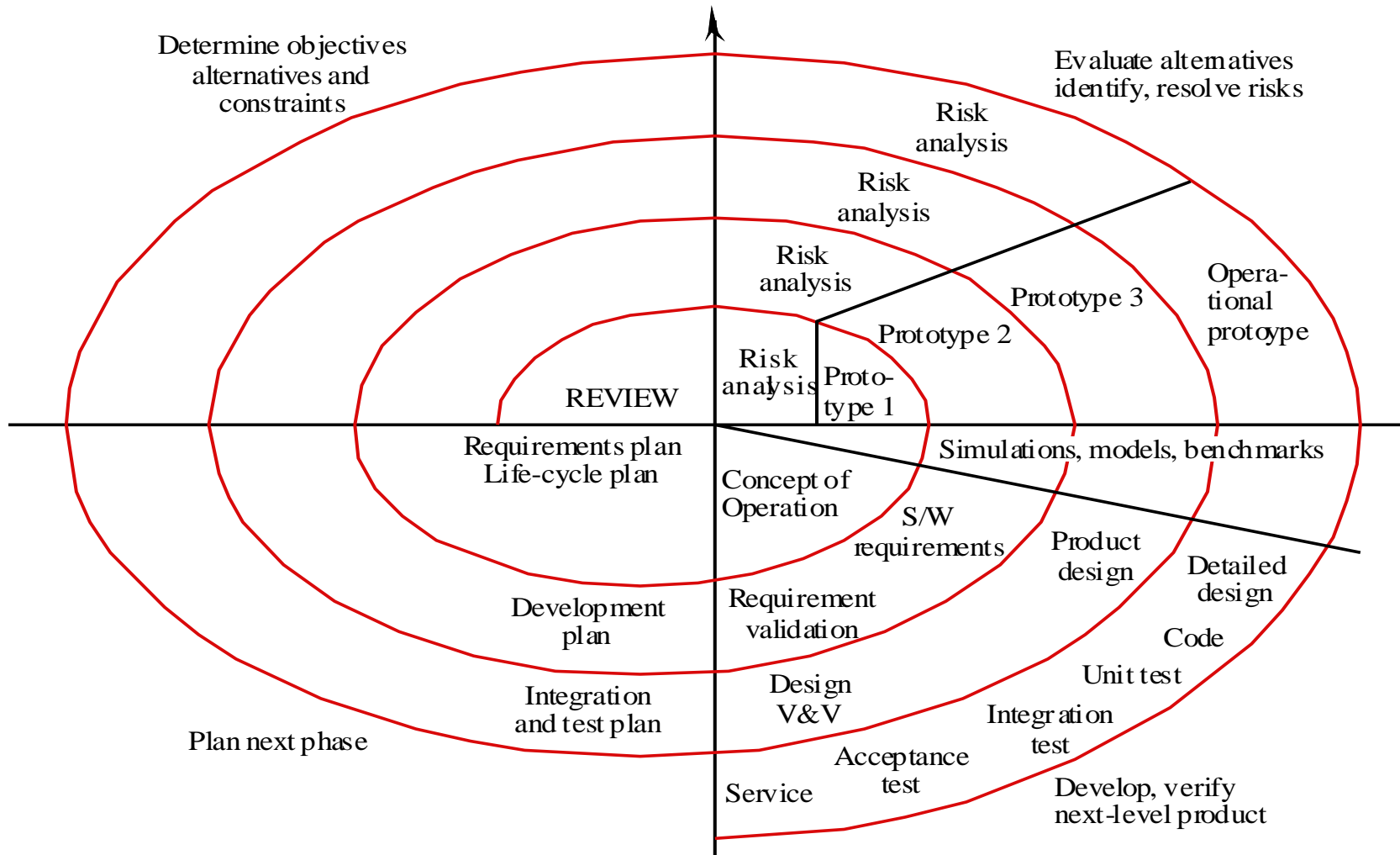


DESENVOLVIMENTO ESPIRAL

- Processo é representado como uma espiral ao invés de uma seqüência de atividades com retorno
- Cada volta na espiral representa uma fase no processo.
- Não existem fases fixas como especificação ou projeto – as voltas na espiral são escolhidas de acordo com o que é requerido
- Os riscos são explicitamente cotados e resolvidos durante todo o processo



MODELO ESPIRAL DO PROCESSO DE SOFTWARE



SETORES DO MODELO ESPIRAL

- Estabelecimento de objetivos
 - Objetivos específicos para a fase são identificados
- Avaliação e redução de riscos
 - Os riscos são avaliados e atividades postas em prática para reduzir os riscos principais
- Desenvolvimento e validação
 - Um modelo de desenvolvimento para o sistema é escolhido, podendo ser qualquer um dos modelos genéricos
- Planejamento
 - O projeto é revisado e a fase seguinte da espiral é planejada

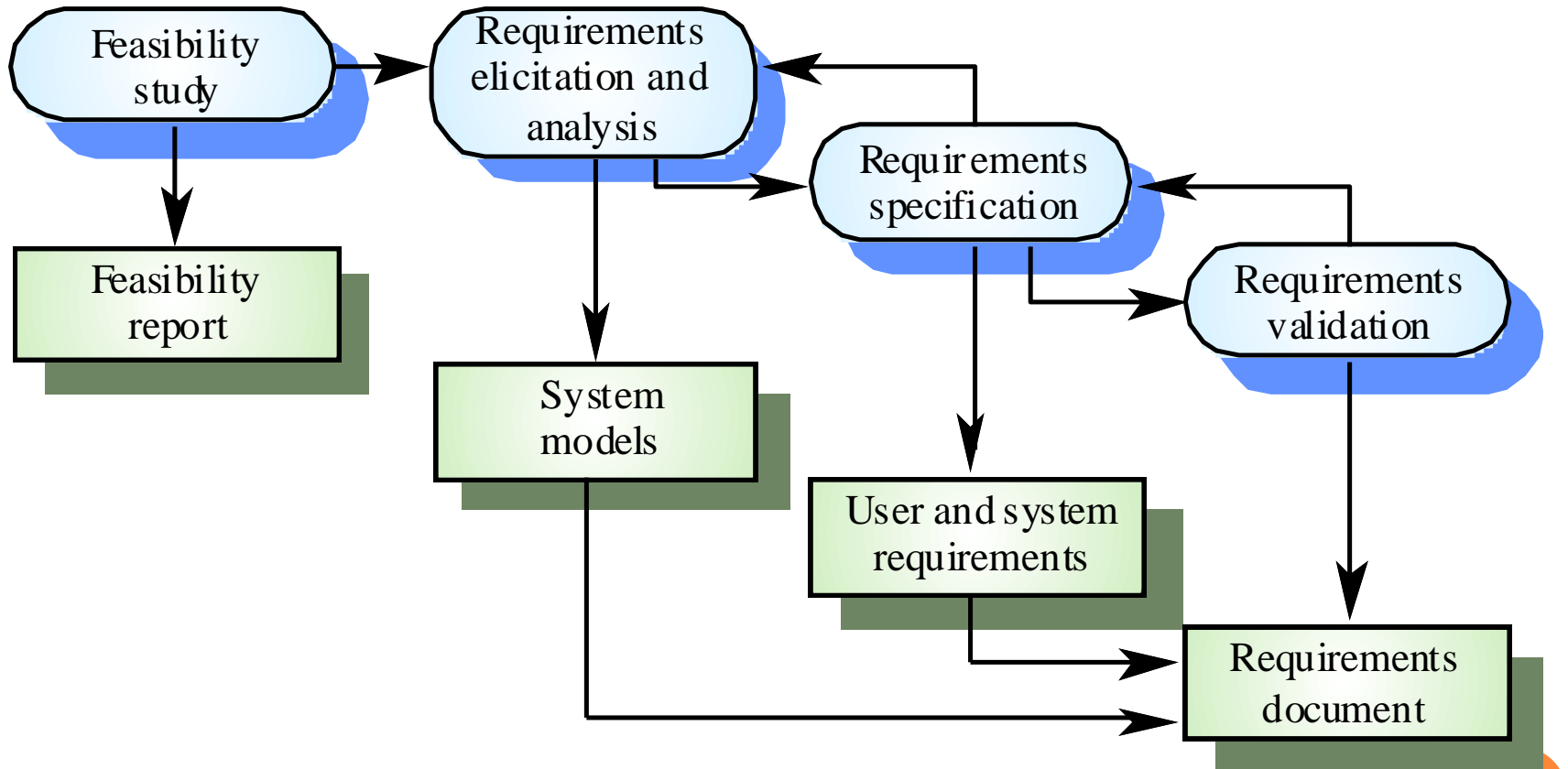


ESPECIFICAÇÃO DO SOFTWARE

- O processo de estabelecer que serviços são requisitados e quais as restrições na operação e desenvolvimento do sistema
- Processo de engenharia de requisitos
 - Estudo de viabilidade
 - Elicitação e análise dos requisitos
 - Especificação dos requisitos
 - Validação dos requisitos



O PROCESSO DE ENGENHARIA DE REQUISITOS



PROJETO E IMPLEMENTAÇÃO DE SOFTWARE

- O processo de converter a especificação do sistema em um sistema executável
- Projeto de Software
 - Projeto de uma estrutura de software que perceba a especificação
- Implementação
 - Transformar esta estrutura em um programa executável
- As atividades de projeto e implementação são intimamente relacionadas e podem ser entrelaçadas



ATIVIDADES DE PROCESSO DE PROJETO

- Projeto arquitetural
- Especificação abstrata
- Projeto de interface
- Projeto de componente
- Projeto de estrutura de dados
- Projeto de algoritmo



MÉTODOS DO PROJETO

- Abordagens sistemáticas para desenvolver um projeto de software
- O projeto é geralmente documentado como uma série de modelos gráficos
- Modelos possíveis
 - Modelo de fluxo de dados
 - Modelo de atributos relacionados à entidade
 - Modelo Estrutural
 - Modelos de objetos

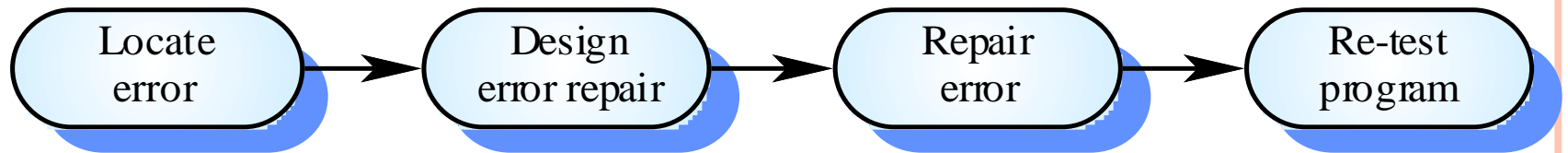


PROGRAMANDO E DEPURANDO

- Transformar um projeto em um programa e remover erros do programa
- Programação é uma atividade pessoal – não existe processo de programação genérico
- Programadores realizam alguns testes de programa para detectar falhas no programa e remover tais falhas no processo de depuração



O PROCESSO DE DEPURAÇÃO

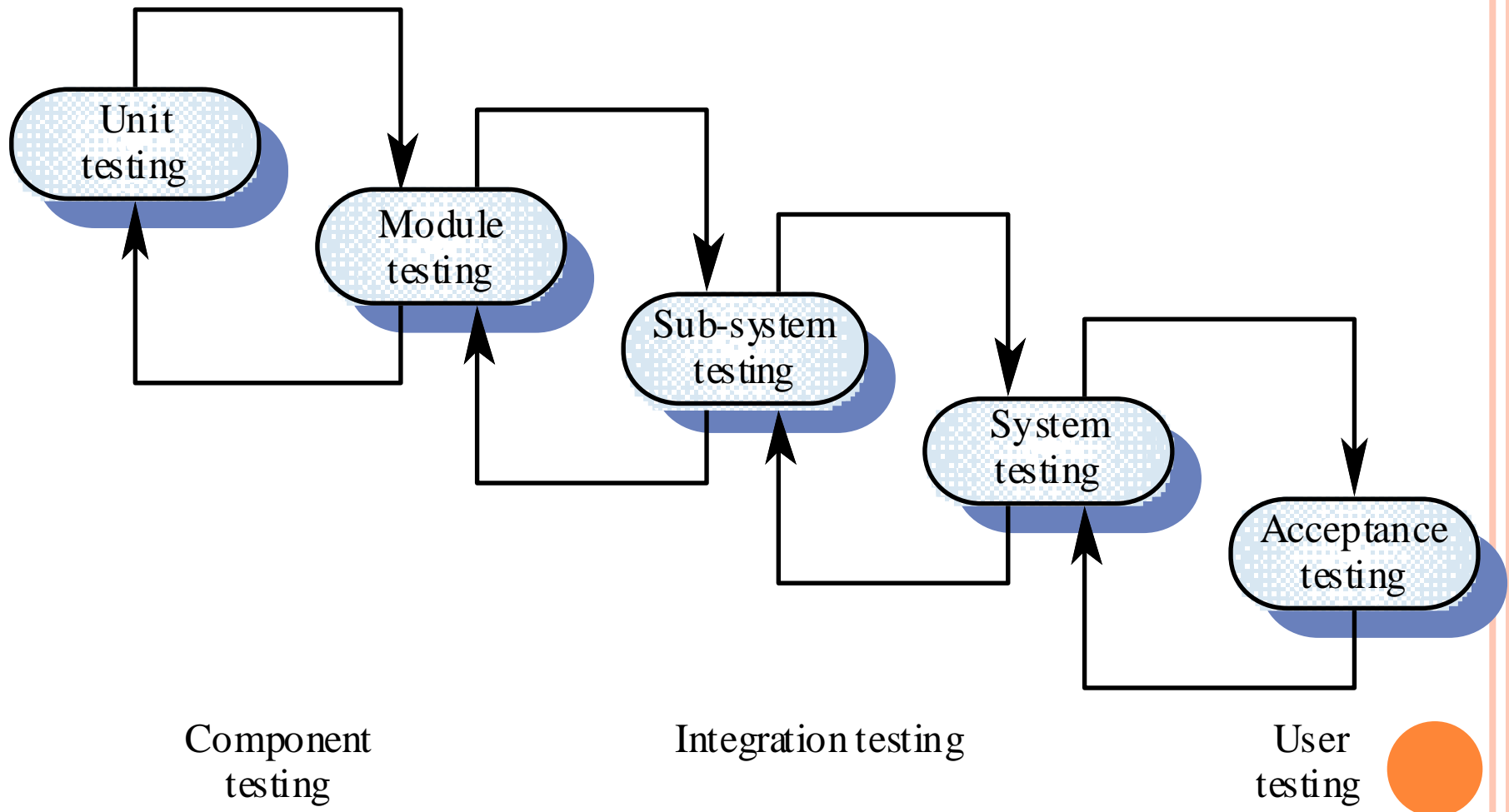


VALIDAÇÃO DO SOFTWARE

- Verificação e validação pretendem mostrar que um sistema está de acordo com sua especificação e cumpre os requisitos do cliente do sistema
- Envolve a verificação e a revisão de processos e teste do sistema
- Teste de sistema envolve a execução do sistema com cases de teste que são derivados da especificação dos dados reais a serem processados pelo sistema



O PROCESSO DE TESTE



ETAPAS DE TESTE

- Teste da Unidade
 - Os componentes individuais são testados
- Teste do Módulo
 - Conjuntos de componentes dependentes relacionados são testados
- Teste do Sub-sistema
 - Os módulos são integrados em sub-sistemas e testados. O foco aqui deve ser no teste da interface
- Teste do Sistema
 - Teste do sistema como um todo. Teste das propriedades emergentes
- Teste de Aceitação
 - Teste com dados do consumidor para verificar que é aceitável



EVOLUÇÃO DO SOFTWARE

- Software é hereditariamente flexível e pode ser mudado.
- Como os requisitos mudam ao se alterar as circunstâncias de negócios, o software que suporta o negócio também deve evoluir e mudar
- Embora tenha havido uma demarcação entre desenvolvimento e evolução (manutenção), este é cada vez mais irrelevante na medida que menos e menos sistemas são totalmente novos

