

# **A Gerência do Processo de Desenvolvimento**

## Ciclo de Vida

- **Qual o propósito de estabelecer um Ciclo de Vida para o Software?**
  - Definir que **atividades** devem ser executadas em um projeto de desenvolvimento de sistemas
  - Introduzir **consistência** entre vários projetos de desenvolvimento de sistemas de uma mesma organização
  - Prover **pontos de controle** para prever, gerenciar e controlar o desenvolvimento de sistemas

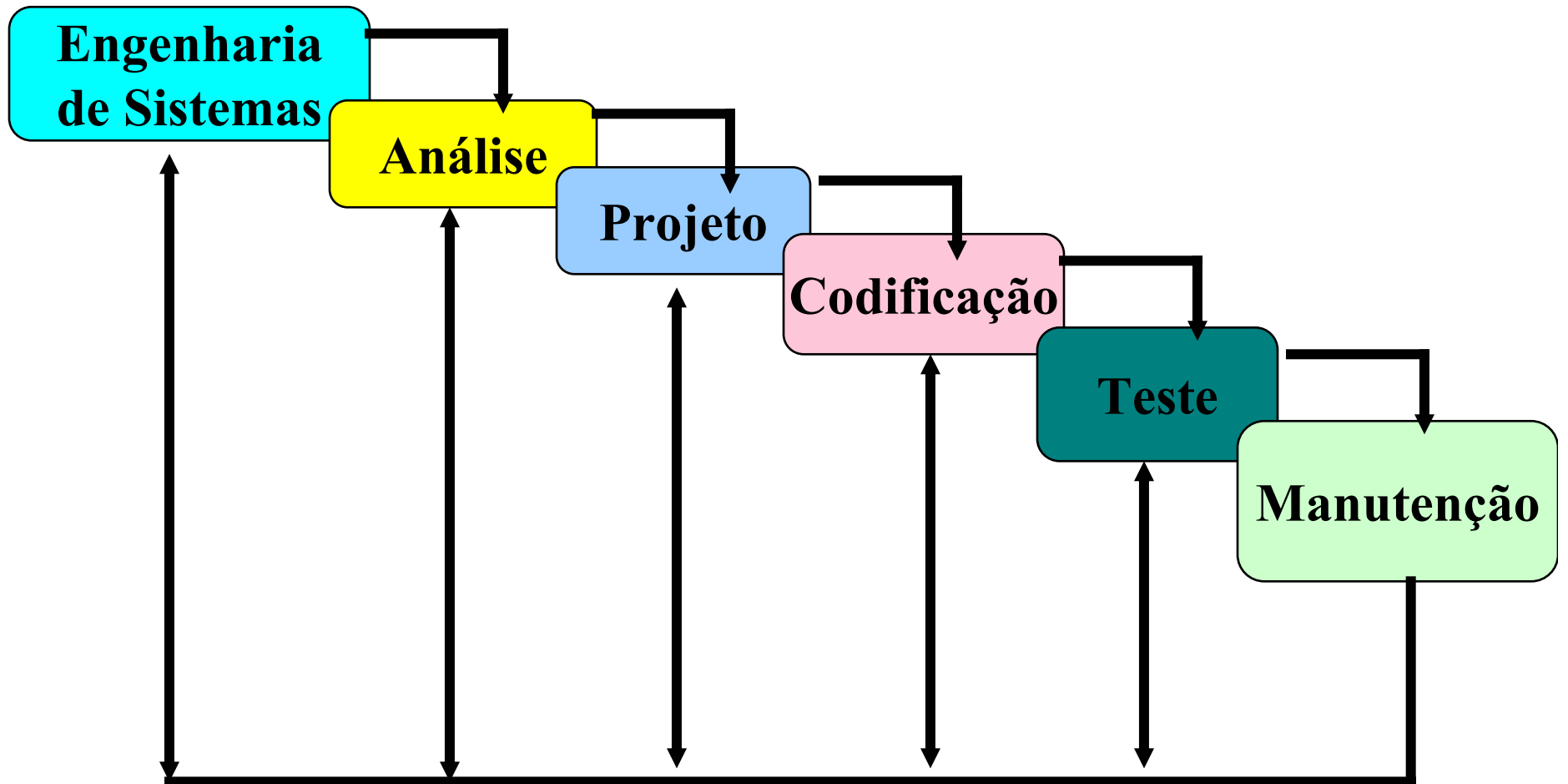
## Ciclo de Vida

- Cascata
- Evolucionário
- Formal
- Orientado a Reuso
- Espiral
- Incremental

# Cascata

- Ciclo de Vida Clássico
- Prevê um processo de desenvolvimento com etapas seqüenciais
- Base: engenharia convencional
- O resultado de cada fase envolve a elaboração de um ou mais documentos que são aprovados

# Cascata



# Cascata

- Problemas:
  - Para grandes projetos o tempo que decorre desde a especificação até sua implantação é grande
  - O Ambiente Externo evolui e é diferente daquele que deu origem a sua especificação
  - Na prática os estágios se sobrepõem
  - O processo de software envolve interações

# Evolucionário

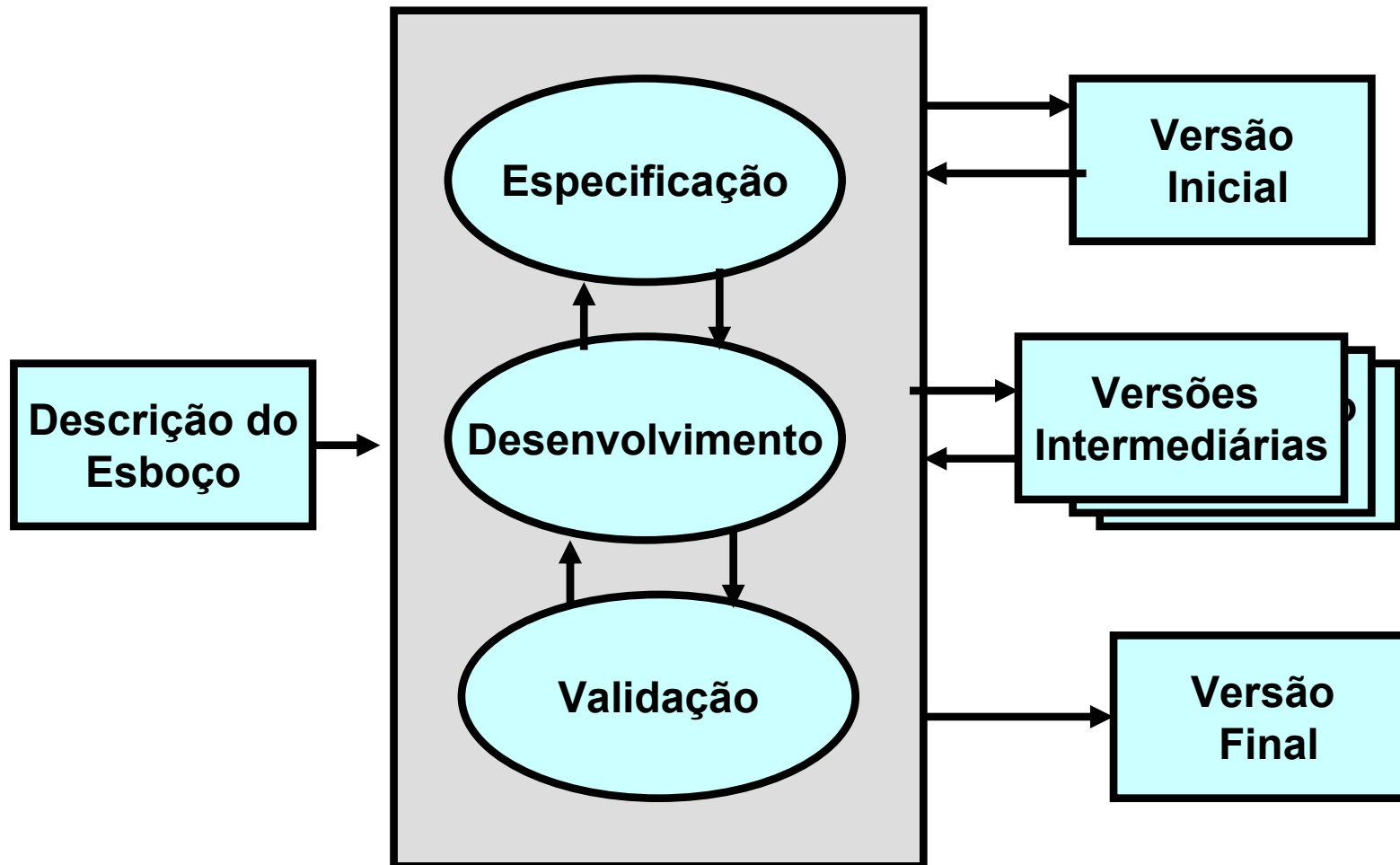
- Base
  - Desenvolver uma implementação inicial
  - Expor o resultado ao comentário do usuário
  - Aprimoramento por meio de muitas versões
  - Até que o sistema tenha sido totalmente desenvolvido
- Dois tipos:
  - Exploratório
  - Protótipos descartáveis

# Evolucionário

- Exploratório
  - Trabalhar com o cliente
  - O desenvolvimento inicia com as partes do sistema que são compreendidas
  - O sistema evolui com as novas características propostas pelo cliente
- Protótipos descartáveis
  - O protótipo experimenta os requisitos não compreendidos
  - Neste caso o objetivo é a Especificação de Requisitos
  - Falaremos de protótipos mais adiante



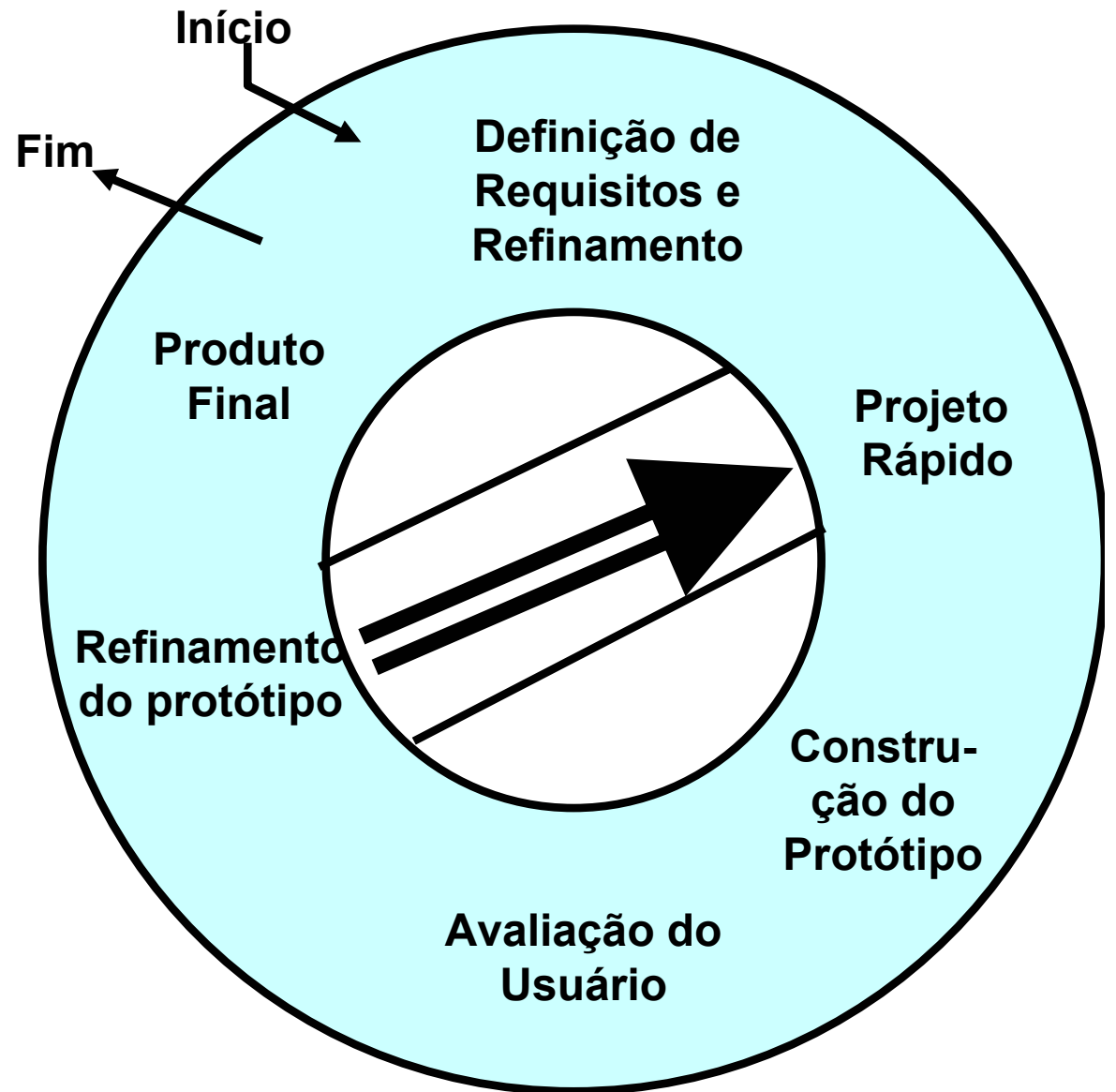
# Evolucionário



# Evolucionário

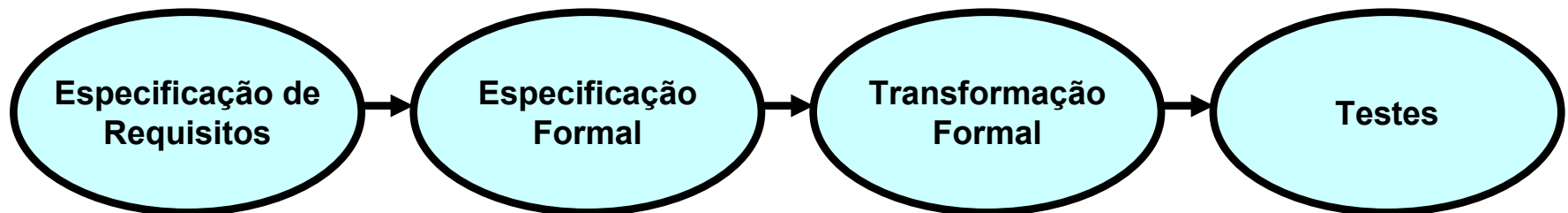
- Produz sistemas que atendem as necessidades imediatas do cliente
- Problemas
  - O processo não é visível
    - Não se produzem documentos que reflitam as versões do sistema
  - Frequentemente são mal estruturados
    - Software sem estrutura
    - Modificações cada vez mais custosas
  - Ferramentas e técnicas especiais
    - Possibilitam rápido desenvolvimento
    - Poucas pessoas podem ter a habilitação necessária para usá-las

# Evolucionário



# Formal

- Base: a transformação matemática formal de uma especificação de sistema em um programa executável
  - A especificação de requisitos é redefinida com uma linguagem formal



# Formal

- Transformação formal
  - Várias etapas
  - Representação mais detalhada, matematicamente correta
- Adequada a sistemas críticos
  - Permite verificação automática de consistência
  - *Model checking*
    - utiliza máquinas de estado para verificar onde uma determinada propriedade é satisfeita sob todas as condições
  - Prova de teoremas
    - axiomas do comportamento do sistema são empregados para derivar uma prova de que o sistema vai se comportar de uma determinada forma

## Orientado a Reuso

- Ampla base de componentes reusáveis
- Infra-estrutura de integração
- Etapas:
  - De posse da Especificação de Requisitos, buscam-se componentes para implementá-la
  - Negociação – requisitos são modificados para que se possa usar os componentes
- Redução do esforço de desenvolvimento

## Iteração de processo

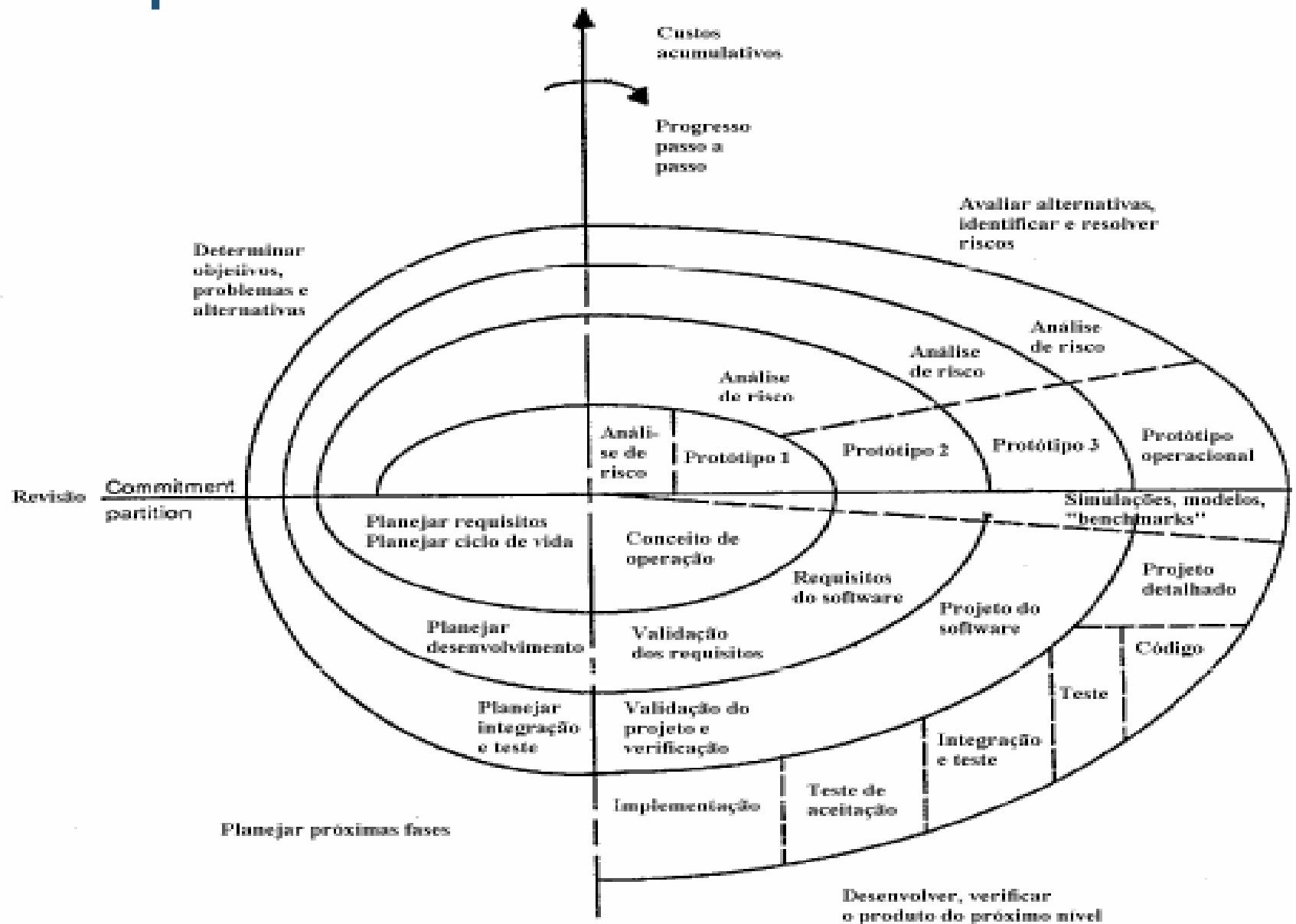
- Existe a necessidade de utilizar diferentes abordagens para diferentes partes do sistema
- Partes do processo são repetidas enquanto os requisitos evoluem
- Modelos Híbridos
  - Apóiam a iteração do processo
  - **Desenvolvimento Espiral**
  - **Desenvolvimento Incremental**

## Modelo Espiral

- O processo de desenvolvimento se move sobre uma espiral evolucionária
  - Melhores características do
    - Ciclo de vida clássico
    - Evolucionário – Prototipação
    - Acrescenta Análise de Riscos
- As fases são executadas de forma iterativa
- As fases de análise e projeto não são monolíticas e distintas



# Modelo Espiral



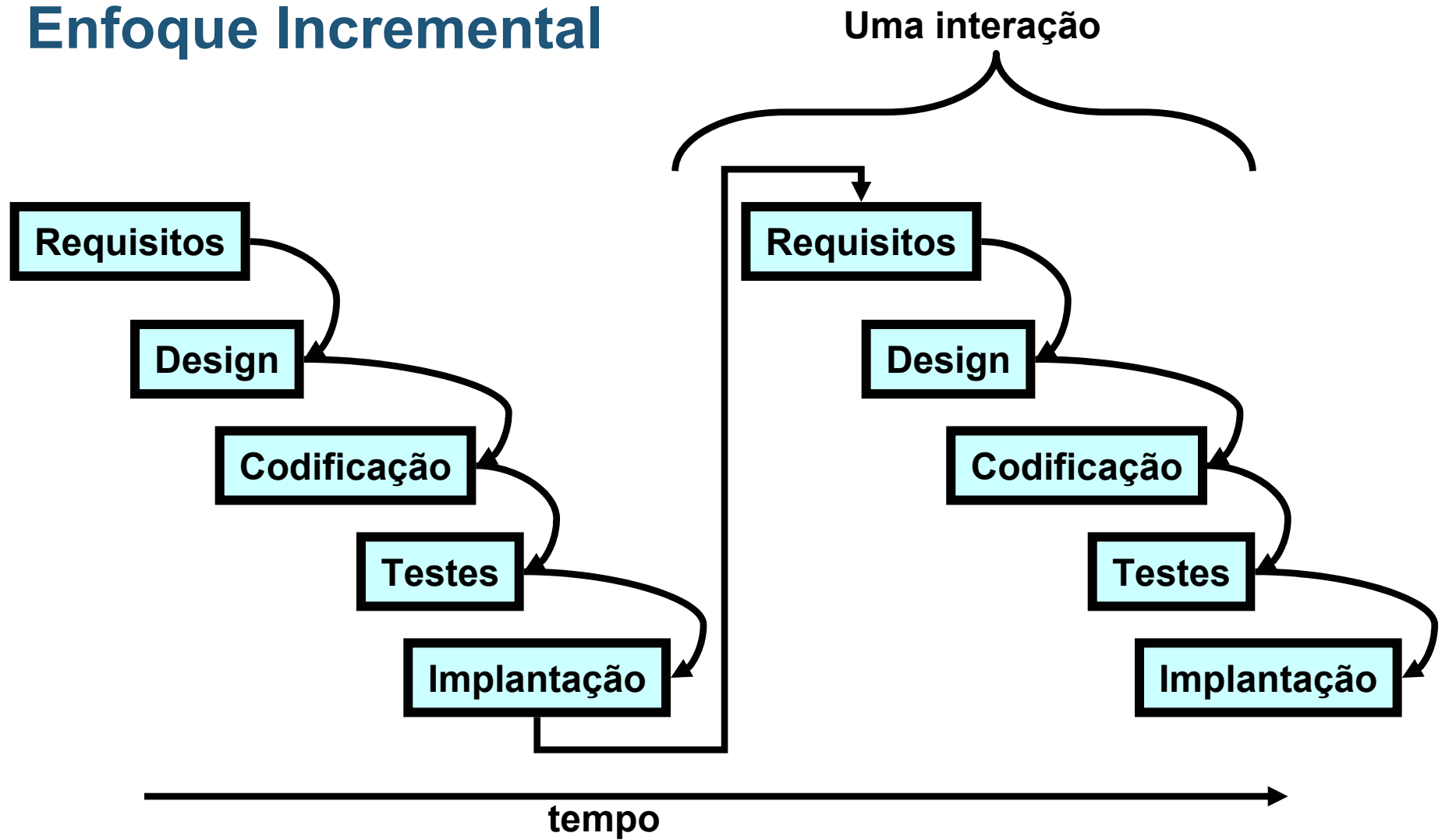
# Modelo Espiral

- Planejamento
  - objetivos, alternativas e restrições
- Análise de Riscos
  - Análise de alternativas e identificação/resolução de riscos
  - Prototipação pode ser usada
  - Simulações e outros modelos podem ser usados para definir melhor o problema
- Desenvolvimento e Validação
  - Desenvolvimento do produto no “nível seguinte”
- Avaliação feita pelo Cliente
- Volta ao Planejamento

## Enfoque Incremental

- Uma variação do modelo cascata onde a partir da fase de especificação de requisitos são feitos incrementos sucessivos.
- Estratégia para minimizar riscos, obtendo-se resultados de médio e curto prazo sem se descuidar do objetivo final

# Enfoque Incremental



## Desenvolvimento Incremental

- Em vez de entregar o sistema como um todo, o desenvolvimento e a entrega são divididos em partes, com cada incremento entregando parte da funcionalidade requerida
- Requisitos dos usuários são priorizados e os requisitos de mais alta prioridade são incluídos nas iterações iniciais
- Uma vez que o desenvolvimento de um incremento é iniciado, os requisitos são "congelados", embora possam continuar a evoluir para incrementos posteriores

# Desenvolvimento Iterativo e Incremental (RUP)

