

Introdução ao RUP

Livar Correia de O. C. Cunha
livarcocc@gmail.com

Rational Unified Process (RUP)

- É um framework de processo de desenvolvimento de software
- Uma metodologia é uma instanciação dos processos definidos nesse framework
- Envolve atividades, artefatos e trabalhadores

Principais Características

- **Iterativo e Incremental**
- **Dirigido por Casos de Uso**
- **Centrado na Arquitetura**
- **Orientado a Objetos**
- **Tratamento de Riscos**

Casos de Uso

- Representam uma funcionalidade do sistema
- Ajudam na comunicação com os clientes.
 - Pode-se usar também protótipos de interfaces gráficas.
- Mostram apenas o que o sistema faz, e não como.

Casos de uso servem para que?

- Criação da Arquitetura
- Testes
- Definição das Iterações
- Documentação do usuário

Arquitetura de Software

- Componentes
- Relacionamentos entre os componentes
- Propriedades destes componentes e relacionamentos

Arquitetura de Software

“Se o projeto ainda não definiu a arquitetura do sistema, incluindo sua justificativa, ele não deve prosseguir com o desenvolvimento em larga escala.”

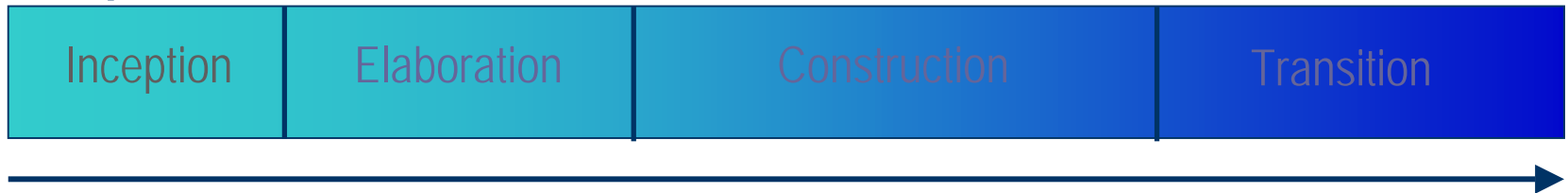
Barry Boehm

Arquitetura de Software (no RUP)

- A definição da arquitetura é imprescindível para o sucesso do projeto
- Os riscos com a arquitetura são críticos
 - Possuem tratamento padrão no RUP
 - Existe uma fase para a estabilização da arquitetura

Fases, Iterações

- O ciclo de vida de um sistema consiste de quatro fases:

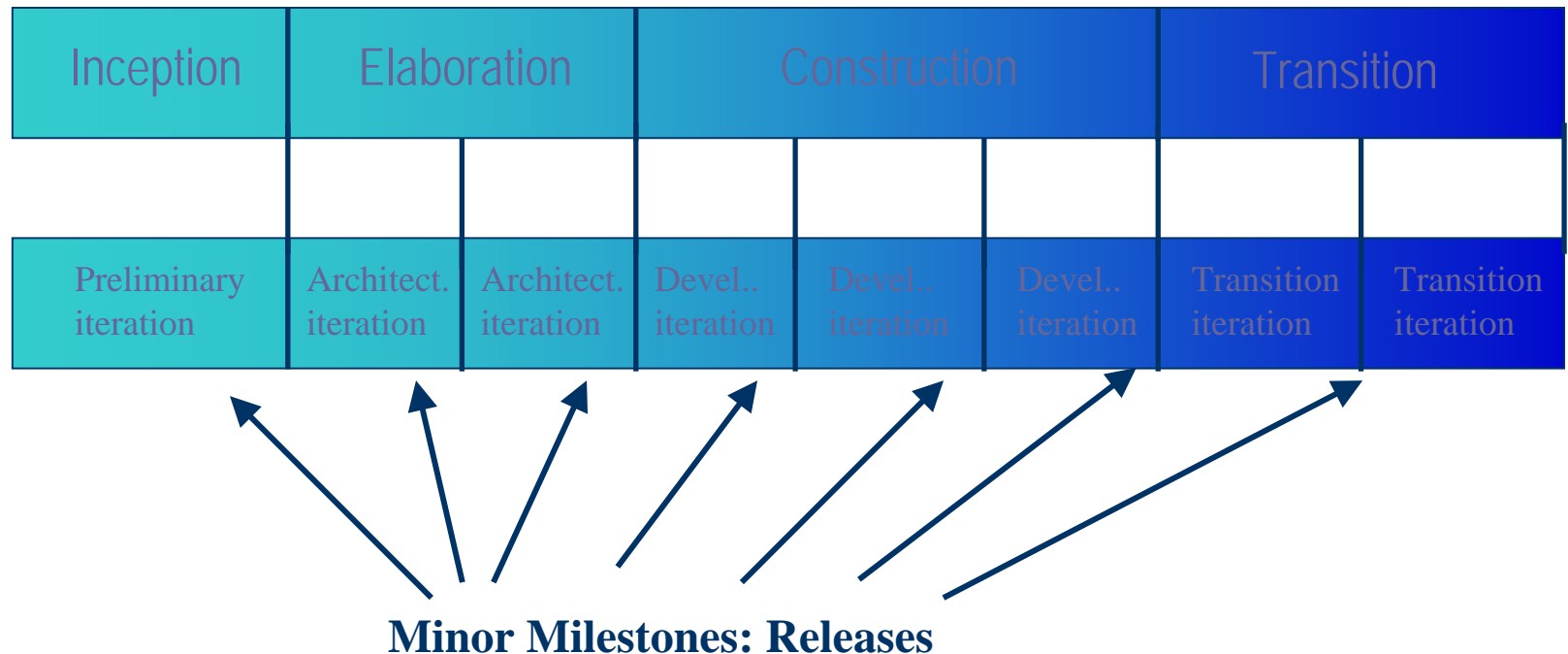


time

- Concepção (define o escopo do projeto)
- Elaboração (define os requisitos e a arquitetura)
- Construção (desenvolve o sistema)
- Transição (implanta o sistema)

Fases e Iterações

- Cada fase é dividida em iterações:



Ciclo de vida Iterativo

- Aplicação do modelo cascata em várias iterações
- As iterações iniciais atacam os riscos mais críticos

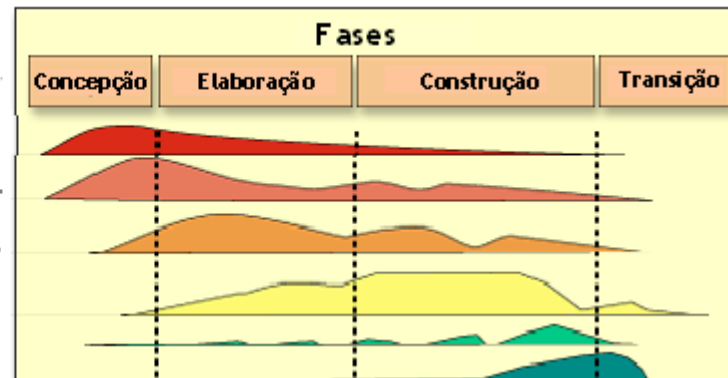
Ciclo de vida Iterativo

- Divide o projeto em partes menores
 - Mais fáceis de gerenciar
 - Mais fácil de medir o progresso
- Todos começam a trabalhar mais cedo
 - Testes e integração são realizados desde o início
 - Riscos mais críticos são resolvidos mais cedo
 - Maior feedback dos usuários

Gráfico das Baleias

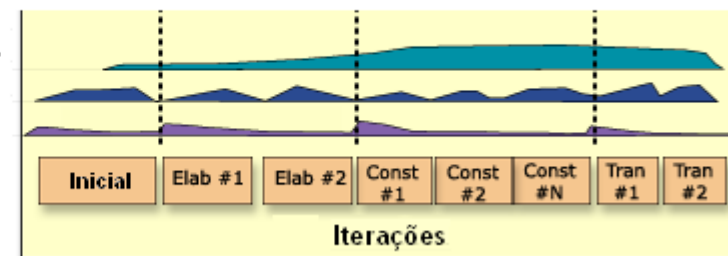
Fluxos de Processo

Modelagem de Negócio.....
 Requisitos.....
 Análise e Projeto.....
 Implementação.....
 Testes.....
 Implantação.....

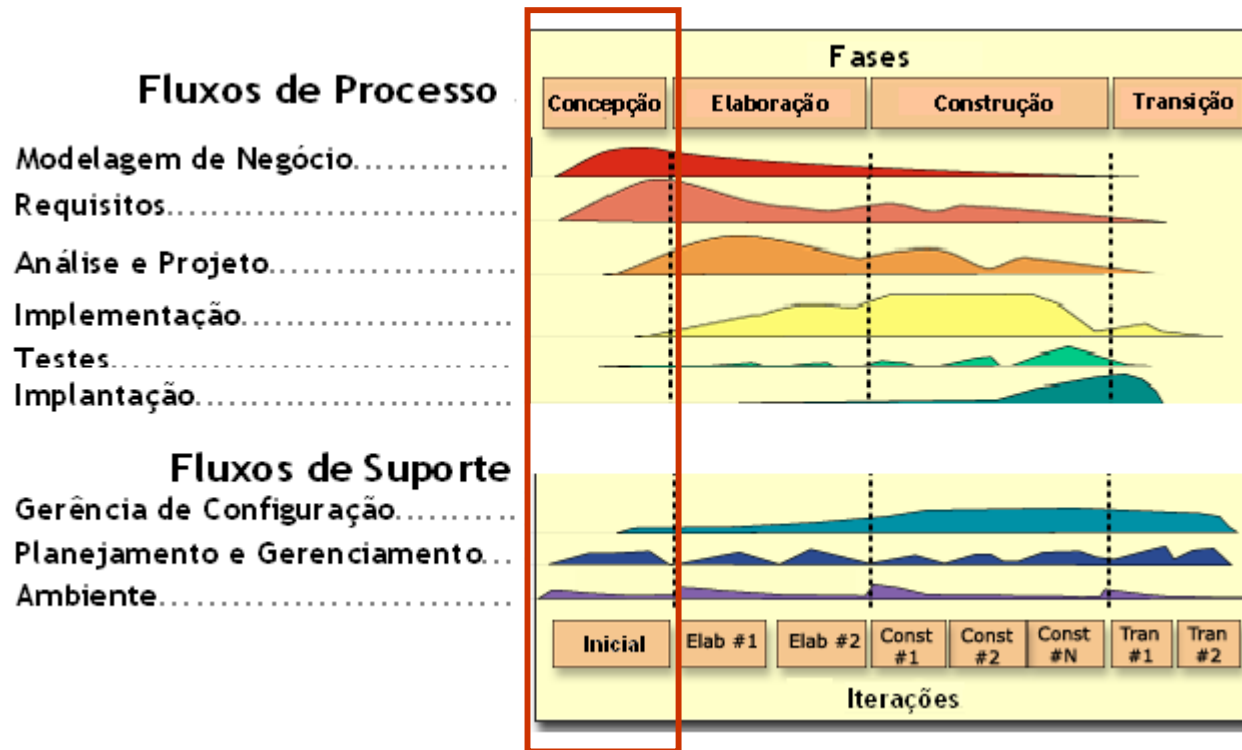


Fluxos de Suporte

Gerência de Configuração.....
 Planejamento e Gerenciamento...
 Ambiente.....

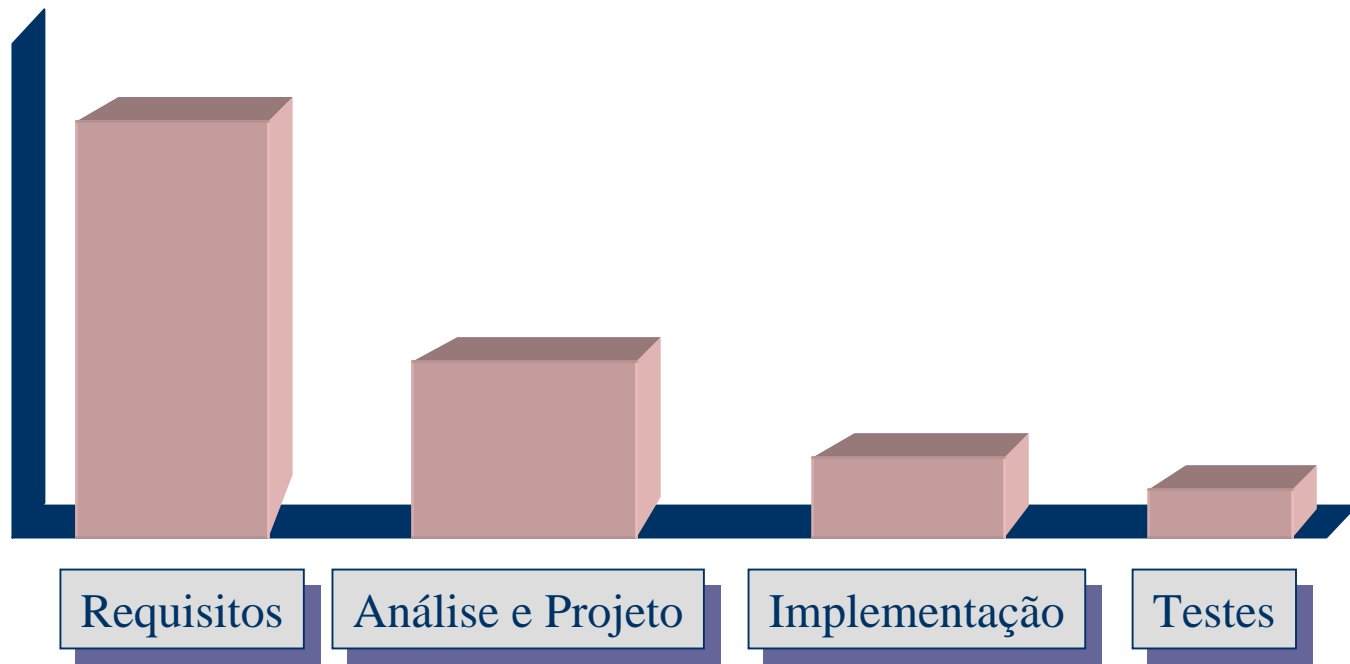


Fase de Concepção



Concepção

Recursos



Fluxos

Requisitos

Análise e Projeto

Implementação

Testes

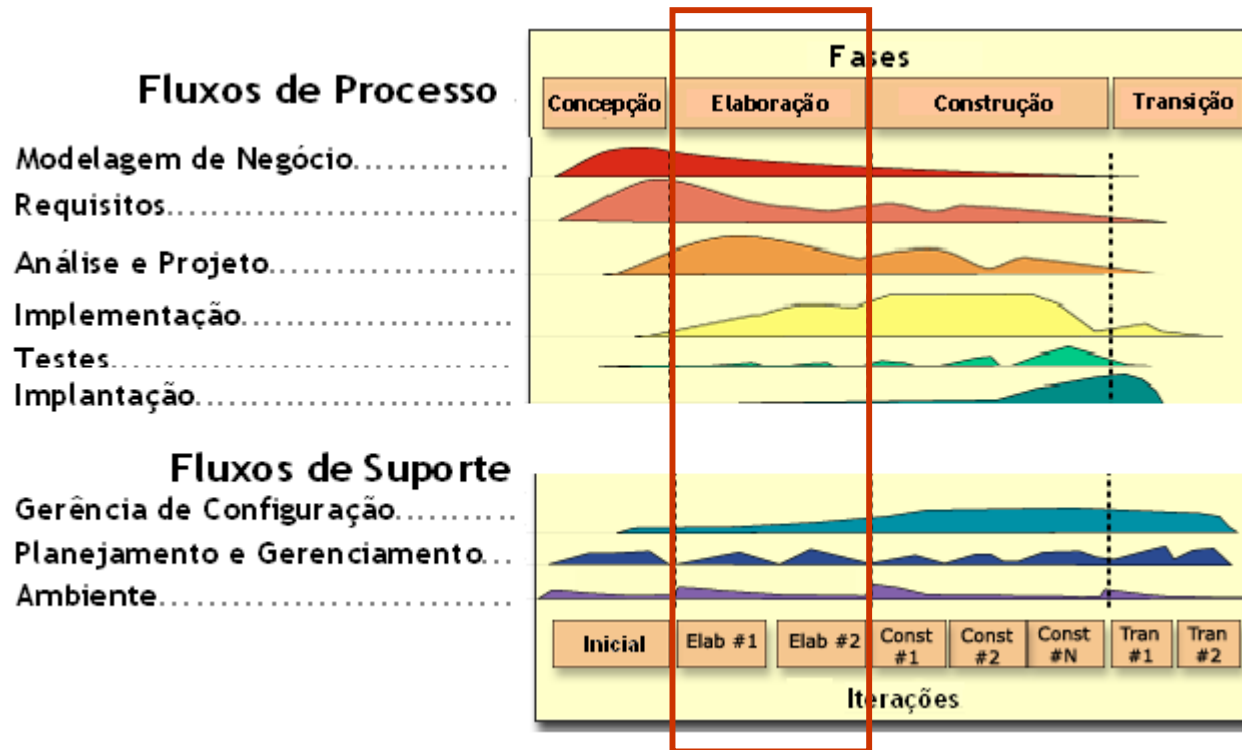
Objetivos da Fase de Concepção

- Definir o escopo do software
 - Visão do Projeto
 - Definir os critérios de aceitação do produto final
 - O que faz parte e o que não faz parte do produto
- Descobrir os casos de uso críticos ou cenários críticos de caso de uso
- Estimar por alto o custo e o cronograma de todo o projeto
 - Estimar em detalhes os custos e cronograma da fase seguinte de Elaboração

Objetivos da Fase de Concepção

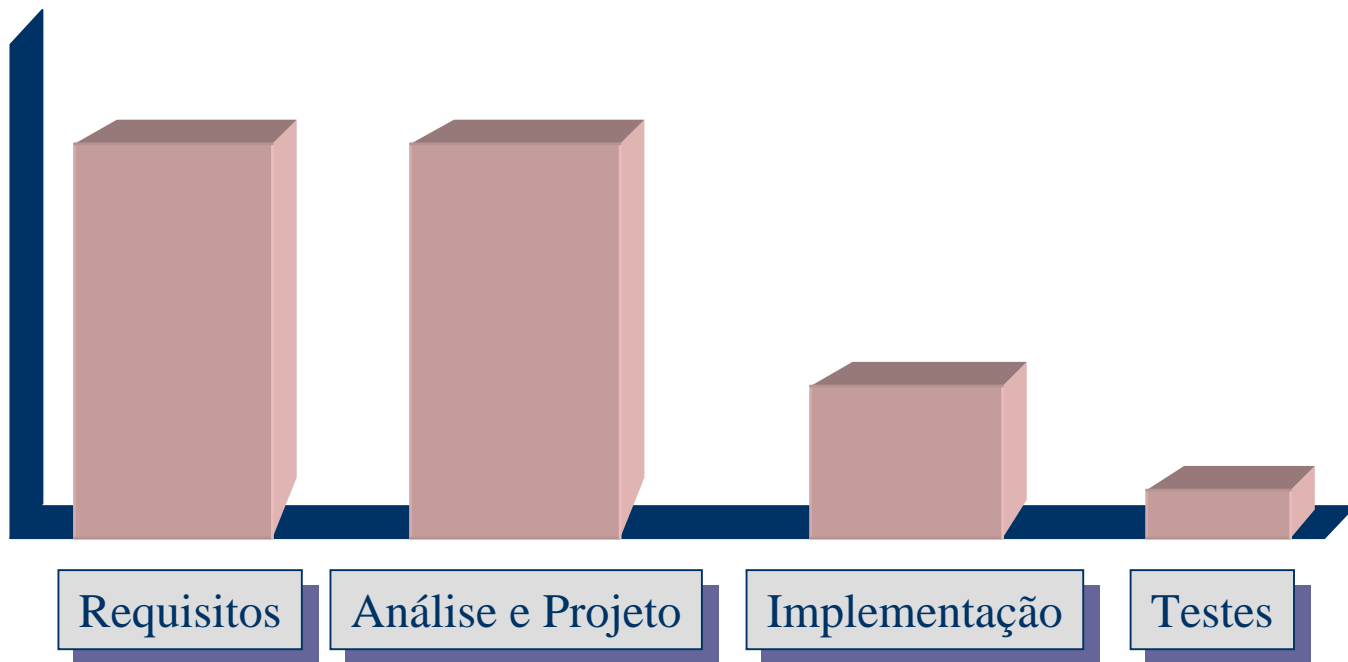
- Levantar os potenciais riscos
- Preparar o ambiente de suporte do projeto
 - Definir e preparar os processos e ferramentas a serem utilizados
- Definir e, eventualmente, demonstrar com protótipos ao menos um candidato de arquitetura
 - Avaliar alternativas de projeto
 - Que componentes fazer, comprar ou reusar?

Gráfico das Baleias - Elaboração



Elaboração

Recursos



Fluxos

Requisitos

Análise e Projeto

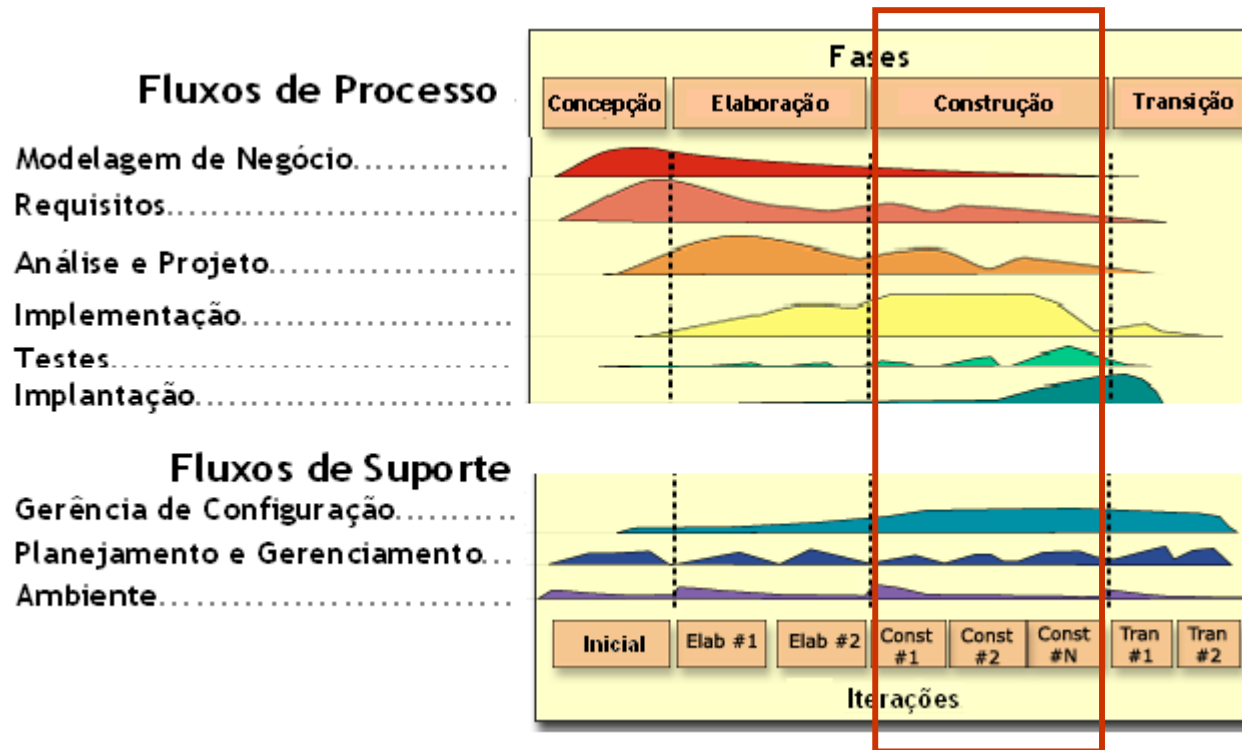
Implementação

Testes

Objetivos da Fase de Elaboração

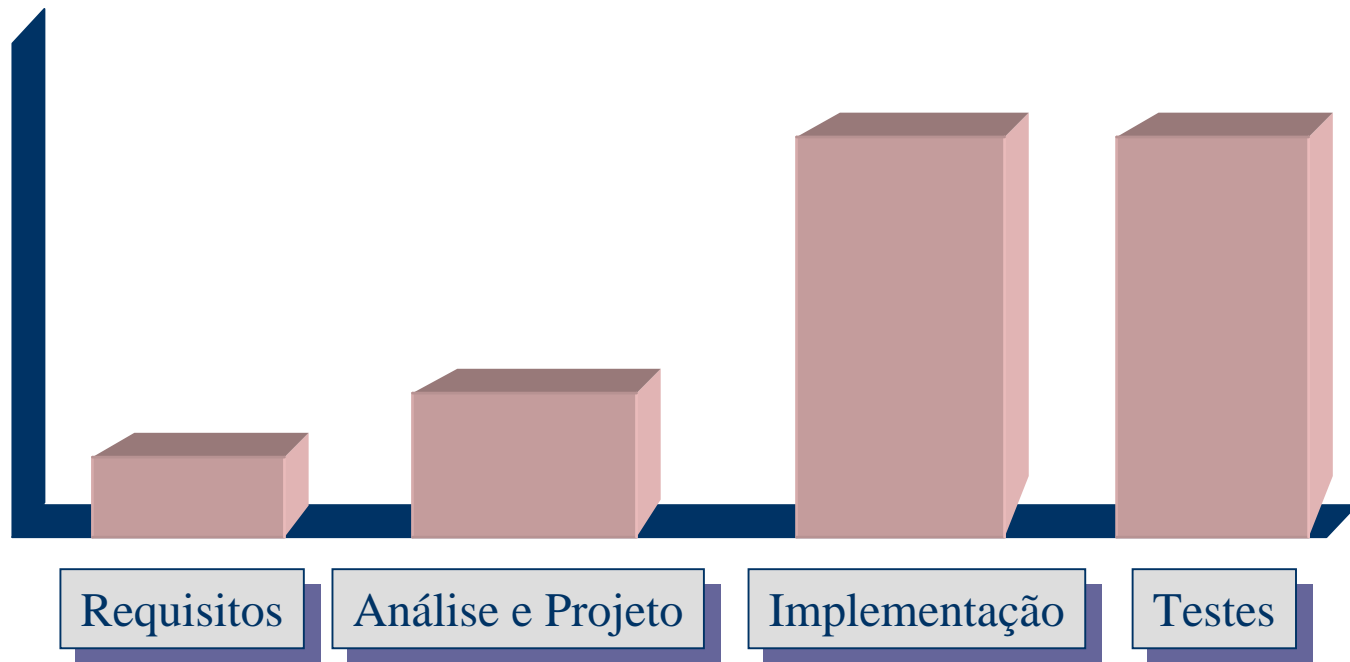
- Capturar a maioria dos requisitos
- Construir a arquitetura do sistema
 - Na forma de um executável (arquitetura executável)
 - Que demonstre a capacidade de esta arquitetura acomodar o resto do sistema
- Produzir protótipos evolucionários ou descartáveis que eliminam riscos de
 - Requisitos ou projeto
 - Reusabilidade de componentes
 - Viabilidade técnica

Gráfico das Baleias - Construção



Construção

Recursos



Fluxos

Requisitos

Análise e Projeto

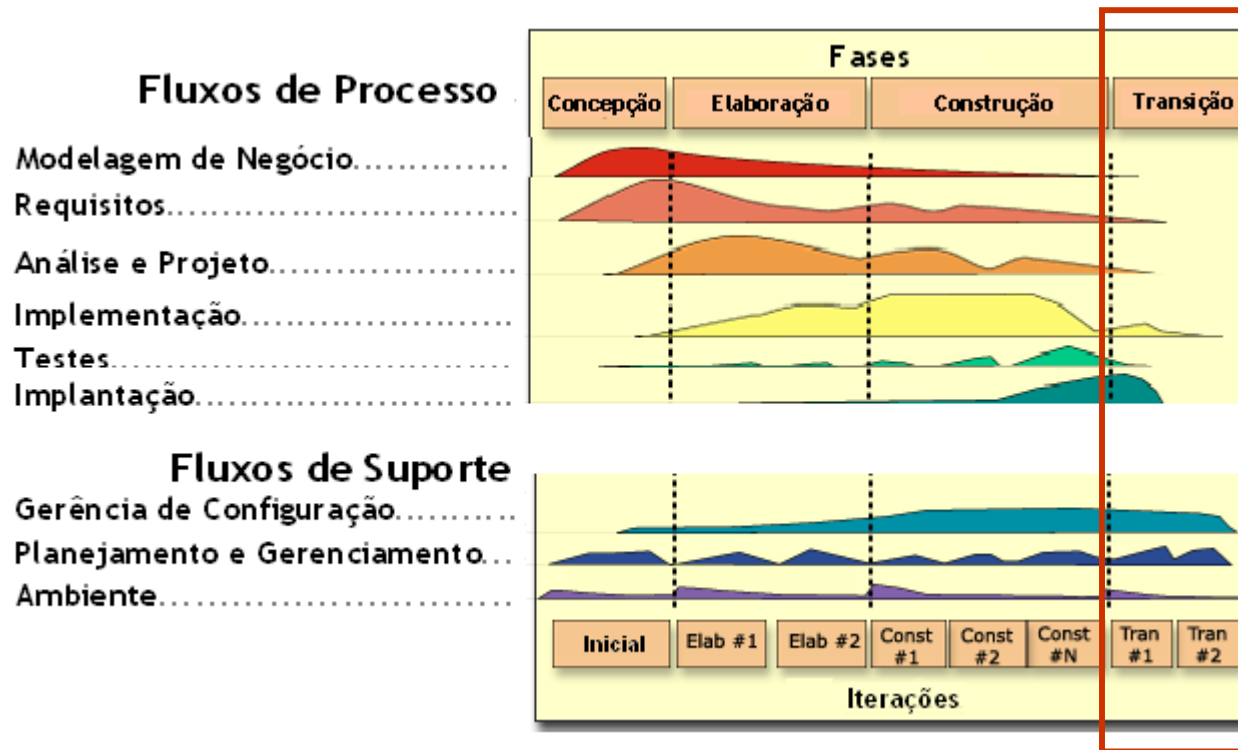
Implementação

Testes

Características da Fase de Construção

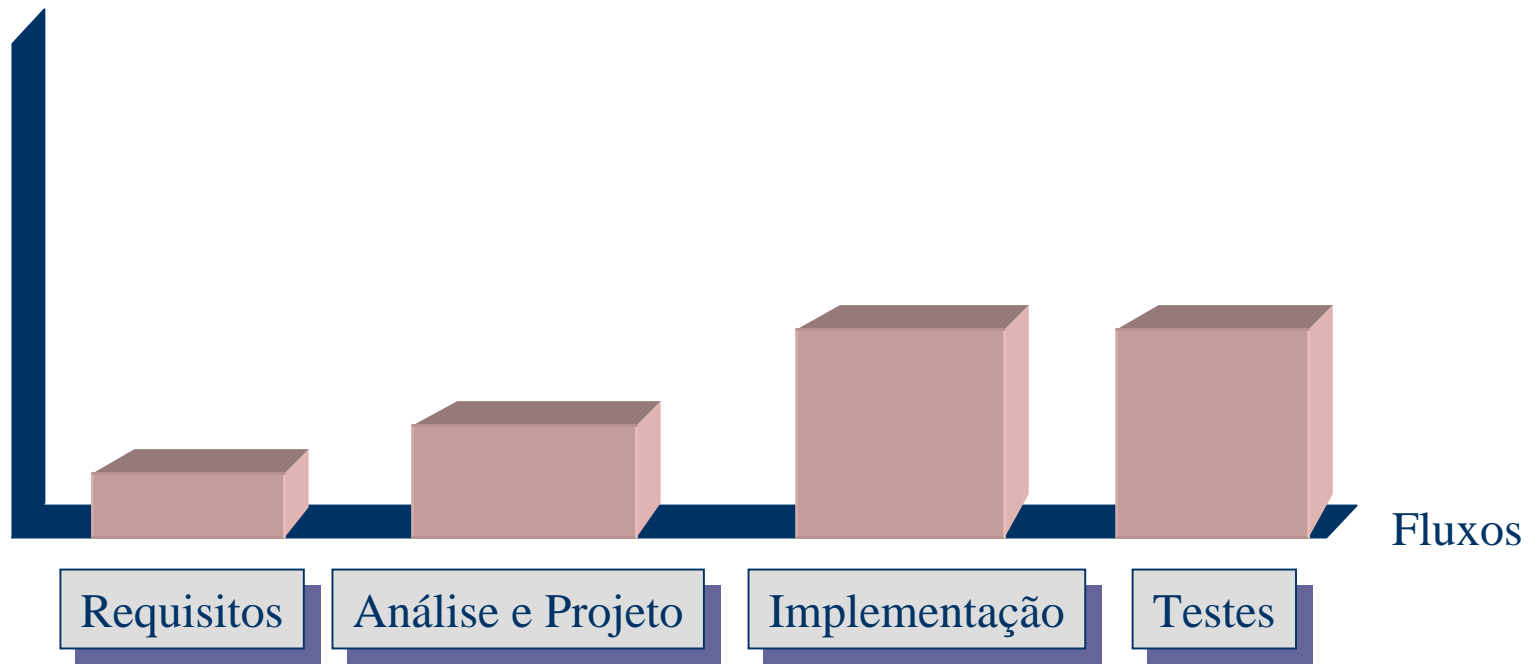
- Produzir a versão para beta-testes
- A ênfase é na produção de um software operacional
- Envolve análise, projeto e implementação dos requisitos levantados na Elaboração
 - Pelo menos 80% dos casos de uso foram levantados (identificados) e entendidos durante a Elaboração
 - Destes, apenas 10-15% são arquiteturalmente relevantes e foram especificados (detalhados), analisados, projetados e implementados
 - A Construção finaliza o sistema, atingindo 100% das funcionalidades implementadas
 - Adiciona “músculos” ao “esqueleto” (arquitetura)

Gráfico das Baleias - Transição



Transição

Recursos



Objetivos da Fase de Transição

- Validar o sistema em relação às expectativas dos usuários
- Beta testes e operacionalização do software (geralmente, em paralelo com o sistema legado)
- Conversão de dados
- Treinamento de usuários
- Correção de erros, pequenas melhorias, pequenos ajustes

Referências

- *The Rational Unified Process: An Introduction (2nd Edition)*, Philippe Kruchten.
- *The Unified Software Development Process*. Ivar Jacobson, Grady Booch e James Rumbaugh.