

Módulo 4: Disciplinas II

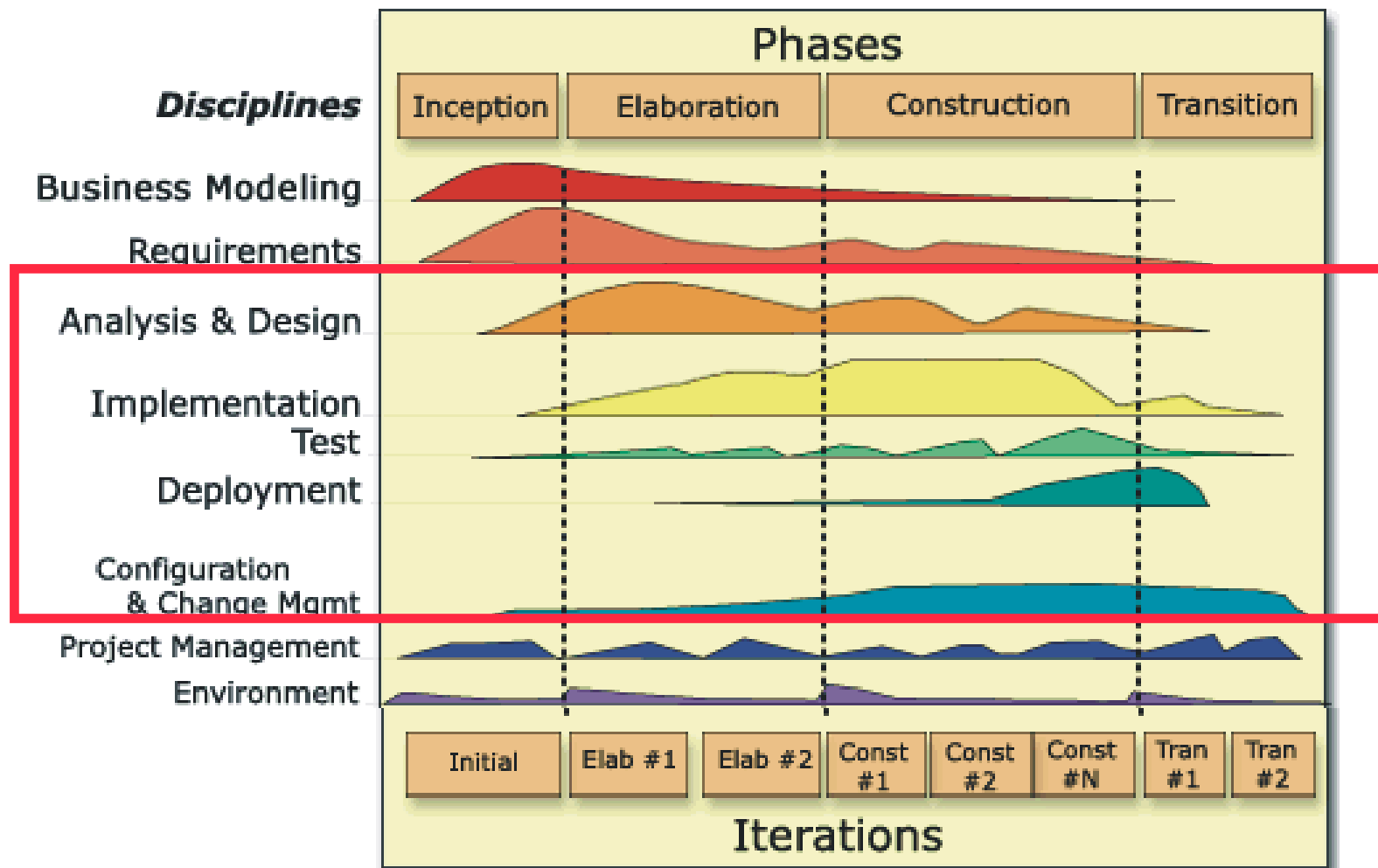
A thick, horizontal yellow brushstroke underline that spans the width of the slide, positioned below the first title.

Fundamentos do Rational Unified Process

Objetivos

- ◆ Entender os conceitos das disciplinas:
 - Análise & Desenho
 - Teste
 - Implementação
 - Implantação
 - Gerenciamento de Configuração e Mudança

Disciplinas

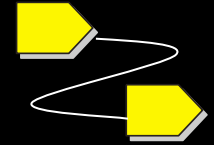


Módulo 4: Conteúdo

→ Análise & Desenho

- ◆ Teste
- ◆ Implementação
- ◆ Implantação
- ◆ Gerenciamento de Configuração e Mudança

Disciplina: Análise & Desenho



♦ Objetivo:

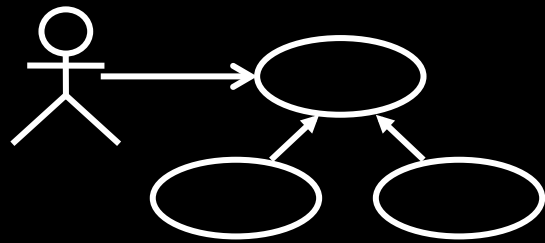
- Transformar os requisitos no Desenho do futuro sistema
- Evoluir uma arquitetura robusta do sistema
- Adaptar o Desenho para adequá-lo ao ambiente de implementação

♦ O artefato primário é o Modelo de Projeto

Um Modelo de Projeto:

- ◆ É um modelo de objeto que descreve a realização de casos de uso.
- ◆ Serve como uma abstração do modelo de implementação e seu código-fonte.
- ◆ É usado como insumo essencial nas atividades de implementação e teste.

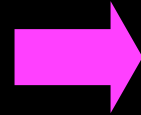
Casos de Uso Conduzem Análise & Projeto



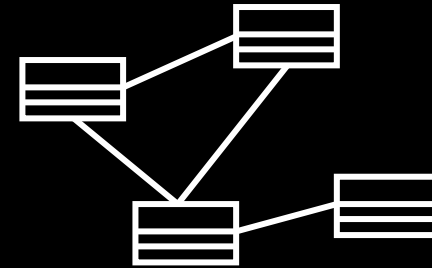
Modelo de Caso de Uso



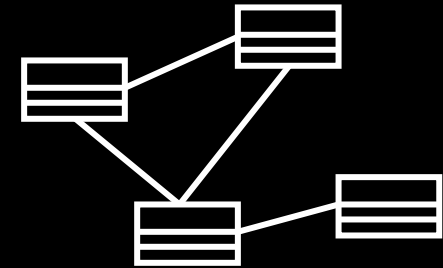
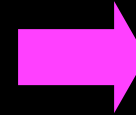
Especificação
Suplementar



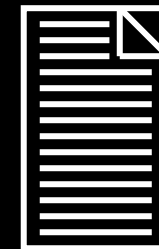
*Análise e
Desenho*



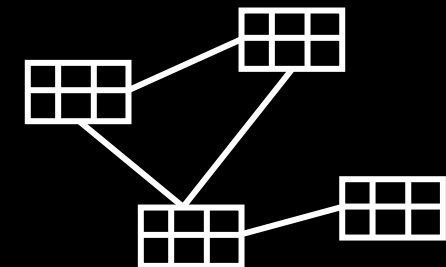
Modelo de Análise (opcional)



Modelo de Desenho



Documento
de Arquitetura

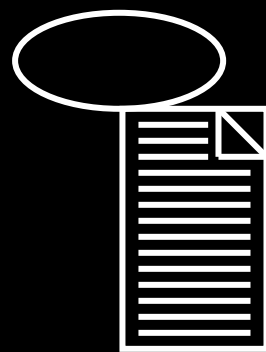


Modelo de Dados

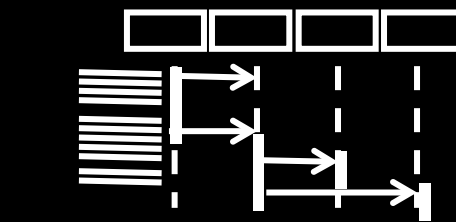
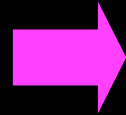
Considerações sobre Análise & Projeto

- ◆ Transforma requisitos em classes e subsistemas
- ◆ Segue as restrições de
 - Requisitos não funcionais
 - Ambiente de implementação
- ◆ Desenho de base de dados
 - Mapeia o modelo de Desenho para um modelo de dados
- ◆ Identifica componentes
 - Subsistemas e interfaces

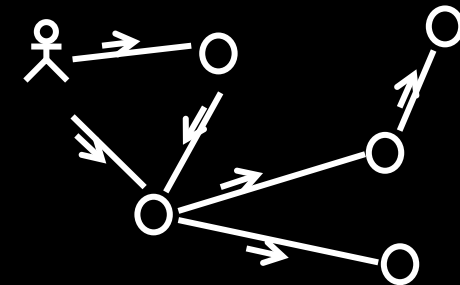
Realização de Casos de Uso na Análise & Desenho



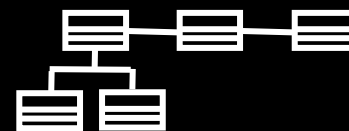
Caso de Uso



Diagramas de Seqüência



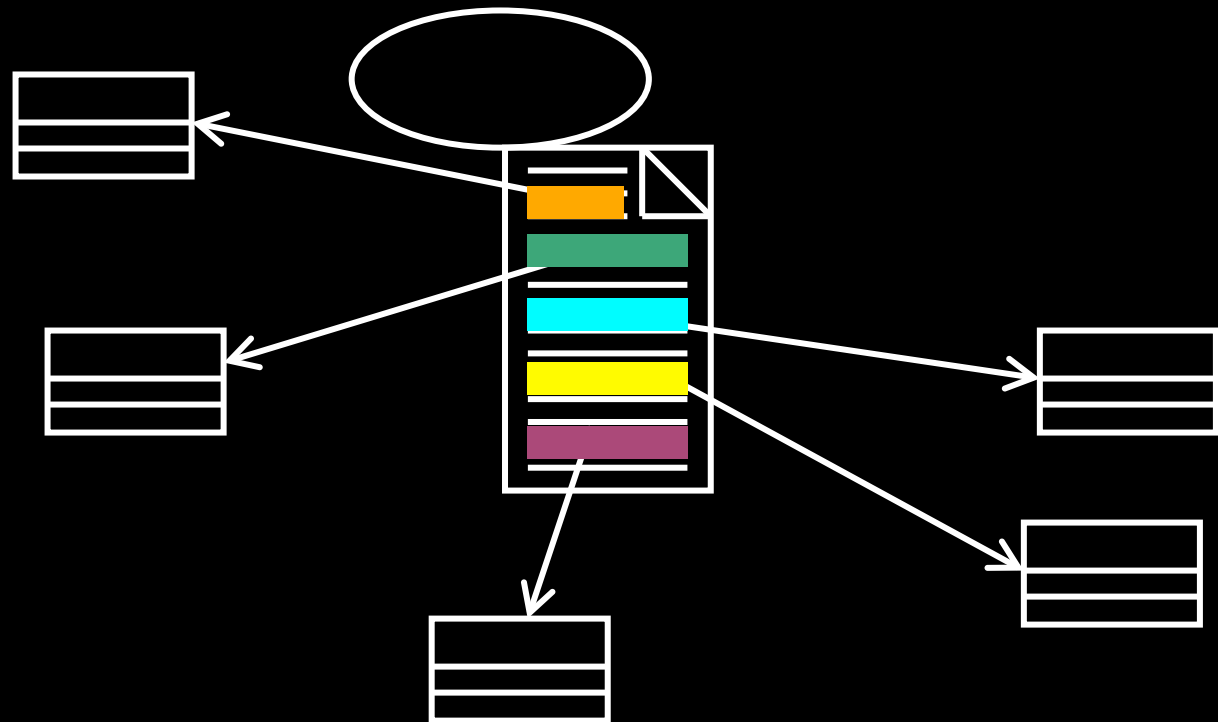
Diagramas de Colaboração



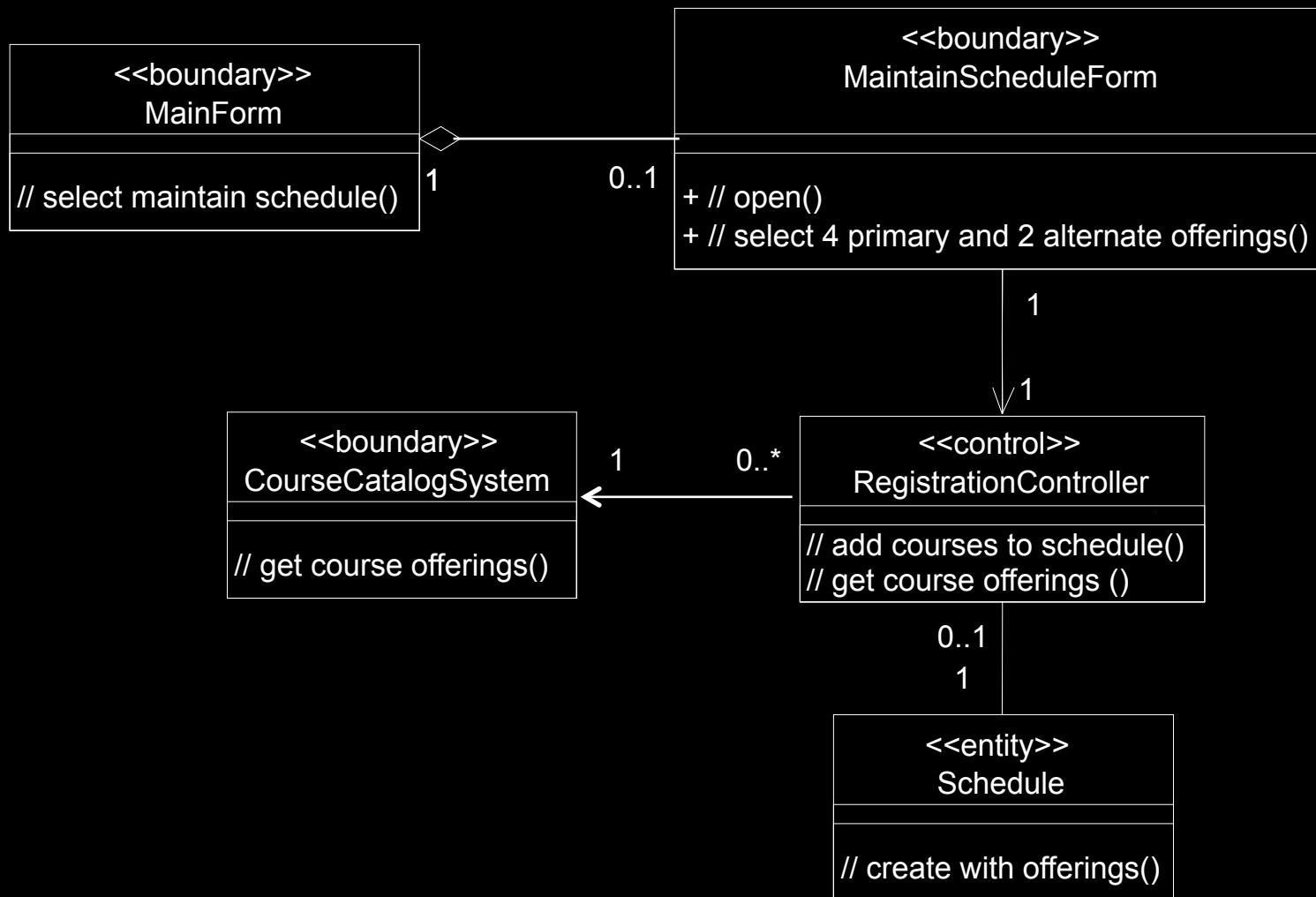
Diagramas de Classe

Análise & Desenho de Caso de Uso

- ♦ O comportamento total de um caso de uso é alocado a classes que colaboram entre si



Exemplo de Diagrama de Classe UML



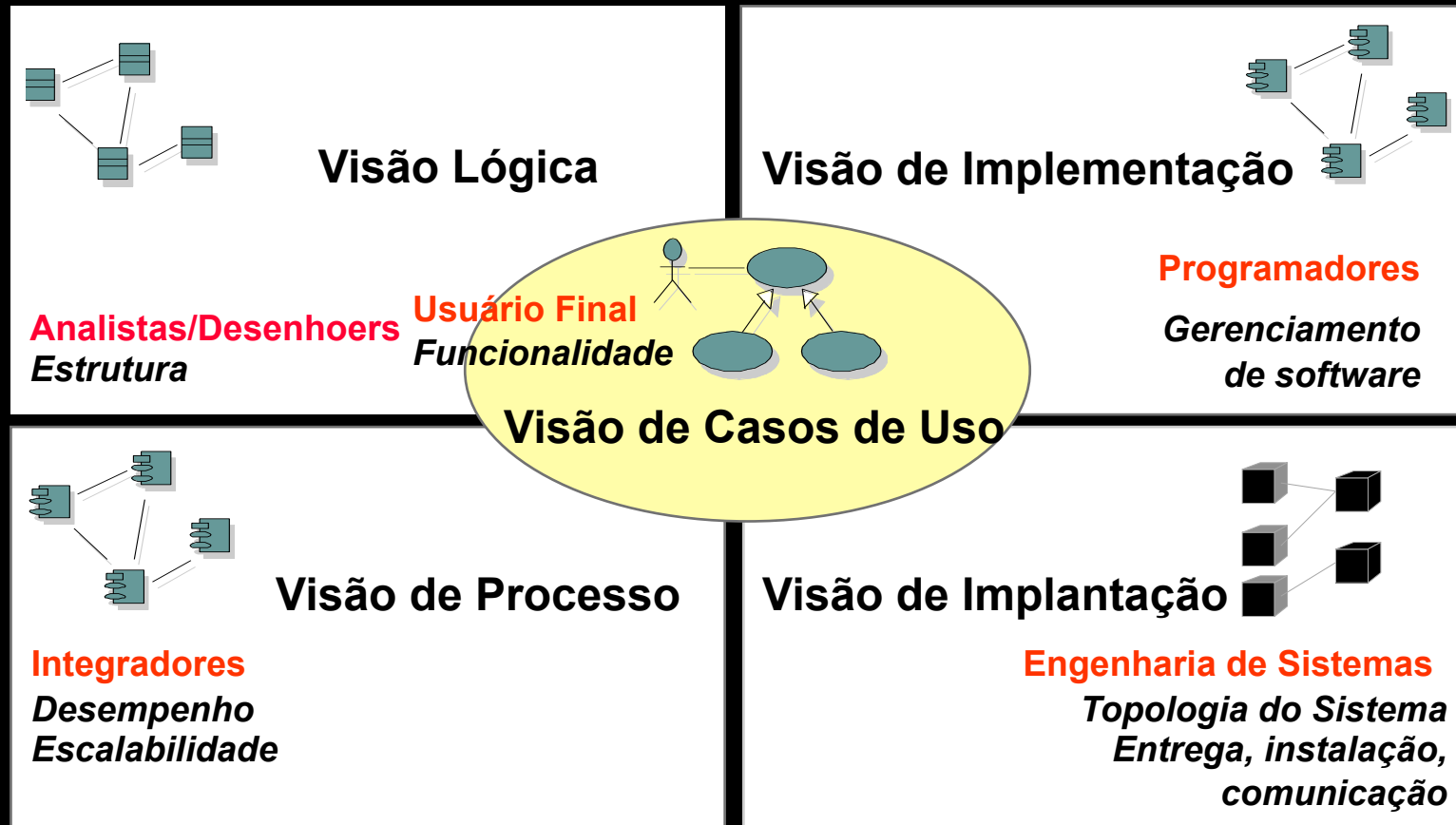
Objetivos da Arquitetura

- ◆ Controle Intelectual
- ◆ Base para reutilização
- ◆ Base para o gerenciamento de projetos

Arquitetura: Controle Intelectual

- ♦ Arquitetura significa muitas coisas para as diferentes partes interessadas.
 - Usuário final
 - Cliente
 - Gerente de projeto
 - Analista de sistema
 - Desenvolvedor
 - Arquiteto de software
- ♦ Realidade multidimensional
- ♦ Vários envolvidos
 - Visões múltiplas, múltiplas plantas

Controle Intelectual: Visões de Arquitetura



Conceitual

Físico

Arquitetura: Base para Reutilização

- ◆ Os **elementos estruturais** e as **interfaces** que compõem o sistema
- ◆ O **comportamento** visto na colaboração destes elementos
- ◆ A **composição** destes elementos em subsistemas progressivamente maiores

Elementos Arquiteturalmente Significativos

- ◆ Nem todo projeto é arquitetura
- ◆ Principais classes de negócio
- ◆ Mecanismos importantes
- ◆ Processos e processadores
- ◆ Camadas e subsistemas
- ◆ Visões de Arquitetura = dividem os modelos

Arquitetura: Base para o Gerenciamento de Projetos

- ◆ Marco de Arquitetura na fase de Elaboração é o marco da Arquitetura do Ciclo de Vida
- ◆ Arquitetura resulta basicamente da Análise & Desenho
- ◆ Arquitetura em fases e iterações:
 - Conduz a redução de risco das iterações
 - Uma baseline de arquitetura é o critério de saída para Elaboração

Módulo 4: Conteúdo

- ◆ Análise & Desenho

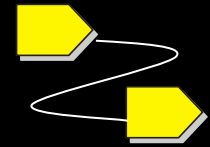
- Teste

- ◆ Implementação

- ◆ Implantação

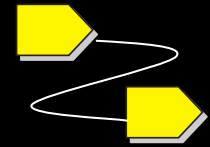
- ◆ Gerenciamento de Configuração e Mudança

Disciplina: Teste



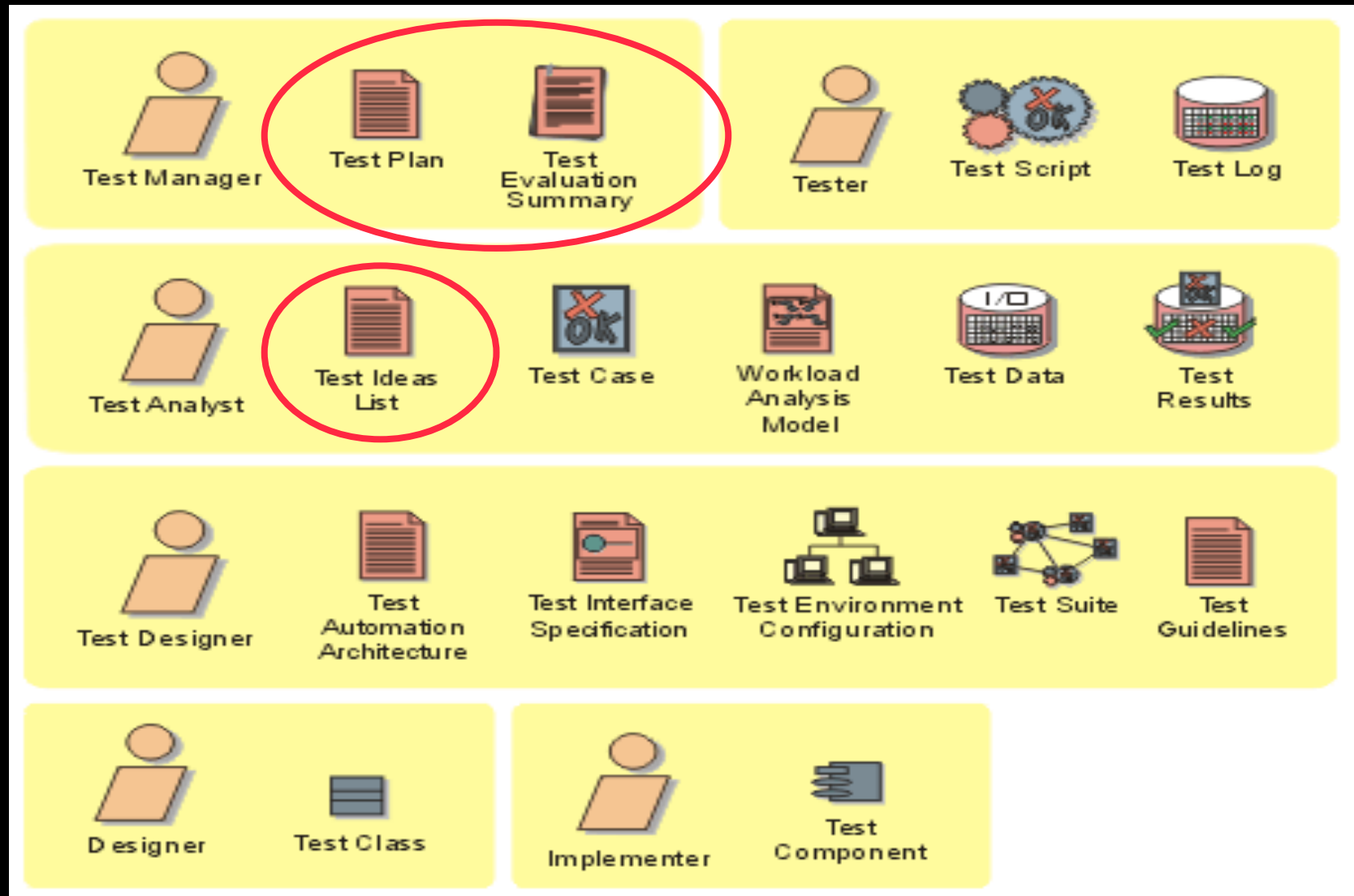
- ♦ Objetivo: Os testes se focam basicamente na avaliação ou diagnóstico de qualidade, conseguida através de um certo número básico de práticas:
 - Encontrar e documentar os defeitos em qualidade do software.
 - Aconselhar em linhas gerais sobre a qualidade percebida do software.

Disciplina: Teste



- Fornecer a validação das suposições feitas no desenho e especificações de requisitos através da demonstração concreta.
- Validar as funções do produto de software como projetadas.
- Validar que os requisitos foram implementados de forma adequada.
- ◆ A disciplina teste age em muitos aspectos, como fornecedor de serviços para outras disciplinas.

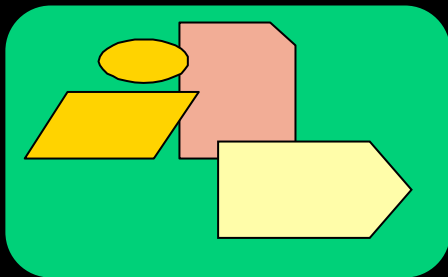
Artefatos da Disciplina Teste



Detalhamento de Fluxo de Trabalho: Definir a Missão de Avaliação

Papéis responsáveis por atividades correlatas:

- Gerenciador de Testes (principalmente)
- Analista de Testes
- Desenhoeiro de Testes



Definir a Missão de Avaliação

Para cada Iteração:

- Identificar os objetivos para o esforço de teste e para os produtos intermediários
- Identificar uma boa estratégia de utilização para os recursos de teste
- Definir o escopo e os limites dos esforços de teste
- Delinear a abordagem que será usada
- Definir como o progresso será monitorado e avaliado

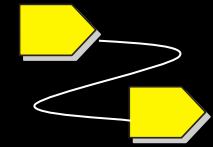
Conceito: Automação de Testes e Ferramentas

- ◆ Ferramentas de aquisição de dados
- ◆ Ferramentas de mensuração estática
- ◆ Ferramentas de mensuração dinâmica
- ◆ Simuladores ou Drivers
- ◆ Ferramentas de Gerenciamento de Testes

Módulo 4: Conteúdo

- ◆ Análise & Desenho
- ◆ Teste
- ➔ Implementação
- ◆ Implantação
- ◆ Gerenciamento de Configuração e Mudança

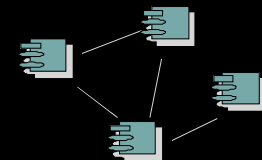
Disciplina: Implementação



♦ Os objetivos da Implementação são:

- Implementar classes e objetos em termos de componentes
- Definir a organização de componentes em termos da implementação de subsistemas
- Testar os componentes desenvolvidos como unidades
- Criar um sistema executável

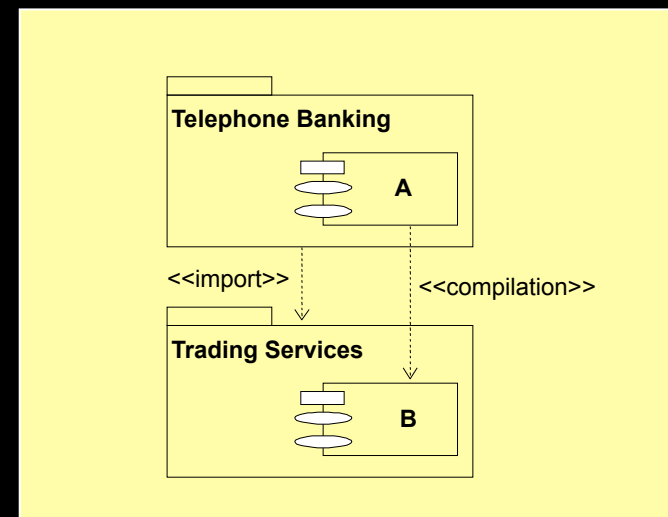
♦ Implementação resulta em um Modelo de Implementação.



Modelo de
Implementação

O que é um Modelo de Implementação?

- ◆ Um Modelo de Implementação consiste de:
 - Componentes
 - Subsistemas de Implementação
- ◆ Componentes incluem:
 - Componentes que podem ser entregues, tais como executáveis
 - Componentes dos quais os executáveis são produzidos, tais como os arquivos de códigos-fonte



Componentes e subsistemas de implementação em um Sistema de Telefonia Bancária.

Conceito: Build

- ◆ A versão operacional de um sistema ou parte de um sistema
- ◆ Demonstra um subconjunto de capacidades fornecidas no produto final
- ◆ Parte integral do ciclo de vida iterativo
- ◆ Fornece pontos de revisão
- ◆ Ajuda a descobrir problemas de integração assim que eles surgem

Módulo 4: Conteúdo

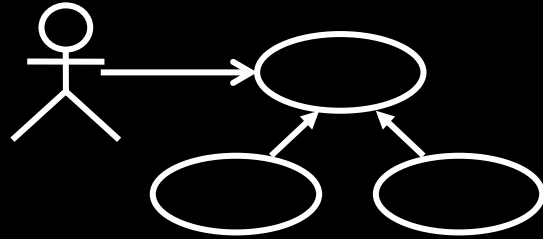
- ◆ Análise & Projeto
- ◆ Teste
- ◆ Implementação
- ➔ Implantação
- ◆ Gerenciamento de Configuração e Mudança

Disciplina: Implantação



- ♦ Objetivo: Gerenciar as atividades associadas com a garantia de que o produto de software esteja disponível para seus usuários finais, tais como:
 - Implantação do Produto
 - Teste na instalação e nos sites alvo
 - Testes Beta
 - Criação de material de suporte para o usuário final
 - Criação de material de treinamento para o usuário
 - Entregar ao cliente (em pacotes fechados, sites para download, etc.)

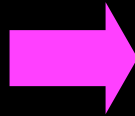
Documentação de Casos de Uso e Usuário Final



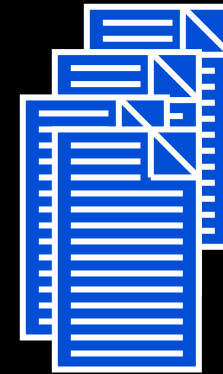
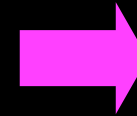
Modelo de Caso de Uso



Especificação
Suplementar



Implantação



Material de Suporte para o Usuário Final

- Guia do Usuário
- Ajuda Online
- Demos
- Tutoriais
- Material de Treinamento

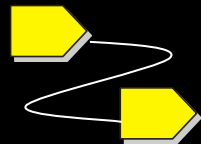
Módulo 4: Conteúdo

- ◆ Análise & Desenho
- ◆ Teste
- ◆ Implementação
- ◆ Implantação
- ➔ Gerenciamento de Configuração e Mudança

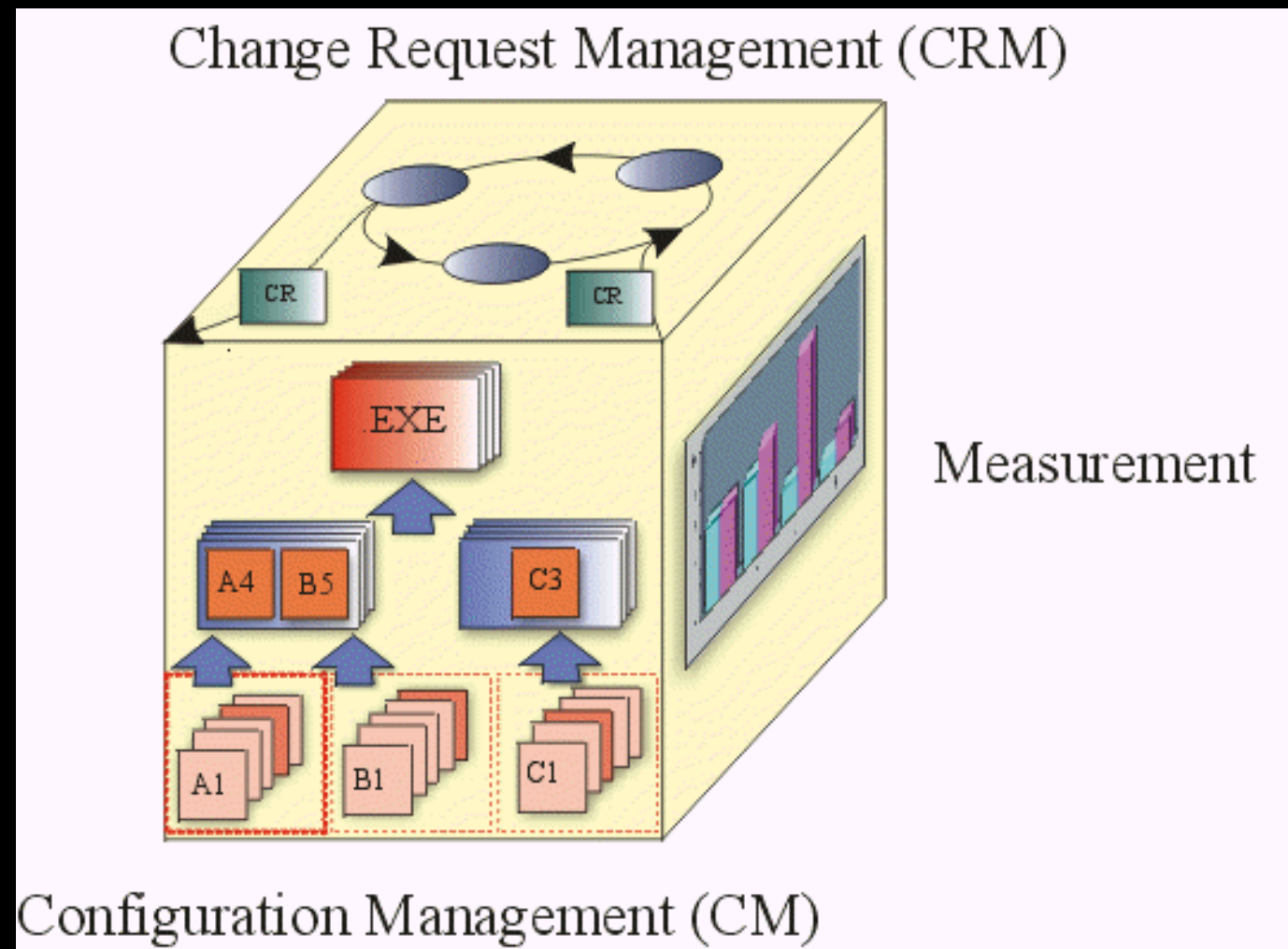
Disciplina: Gerenciamento de Configuração e Mudança

♦ Objetivo: Rastrear e manter a integridade dos artefatos de projeto

- Controle de mudanças
- Identificação e gerenciamento de configurações
- Contabilização do status de configuração
- Rastreamento de mudanças
- Escolha da versão
- Produção de software
- Gerenciamento de áreas de trabalho



Gerenciamento de Configuração e Mudança (CCM)



Gerenciamento de Configuração (CM)

- ◆ Descreve a estrutura do produto e identifica seus itens de configuração, que são tratados como entidades de versão individuais no processo de gerenciamento de configuração
- ◆ Trata de definir configurações, building e rotular; agrupar artefatos versionados em conjuntos e manter a rastreabilidade entre estas versões

Gerenciamento de Solicitação de Mudança (CRM)

Trata:

- ♦ Da infraestrutura organizacional necessária para avaliar o impacto dos custos e dos prazos de uma mudança solicitada no projeto
- ♦ Do trabalho de uma Equipe de Revisão de Mudanças ou Comitê de Controle de Mudança (CCB)

Contabilização do Status de Configuração

- ◆ Este tipo de contabilização descreve o estado de um produto baseado no tipo, número, taxa, e severidade dos defeitos encontrados e solucionados durante o curso do desenvolvimento do produto.
- ◆ Métricas derivadas deste aspecto, ou através de auditorias ou dados rudes, são úteis em determinar o status de conclusão geral do projeto.

Revisão

- ◆ Quais são as partes de um Modelo de Implementação?
- ◆ Quais são os principais componentes de uma unidade de teste de automação?
- ◆ Como os casos de uso são utilizados na disciplina de Implantação?
- ◆ Quais são os elementos da disciplina de Gerenciamento de Configuração e Mudança?