

Engenharia de Software

Natália Schots

Agenda

- Processos
 - Definição
 - Notações
- Processo Unificado
- Desenvolvimento ágil
- Exemplos

Na aula passada...

Tópicos Abordados (1/2)

– Processos do MR-MPS

- Nível G

- Gerência de Projetos / Gerência de Requisitos

- Nível F

- Aquisição / Gerência de Configuração / Garantia da Qualidade / Gerência de Portfólio / Medição

- Nível E

- Definição do Processo Organizacional / Avaliação e Melhoria do Processo Organizacional / Gerência de Recursos Humanos / Gerência de Reutilização

Tópicos Abordados (2/2)

– Processos do MR-MPS (cont.)

- Nível D

- Desenvolvimento de Requisitos / Projeto e Construção do Produto / Integração do Produto / Verificação / Validação

- Nível C

- Desenvolvimento para Reutilização / Gerência de Decisões / Gerência de Riscos

- Nível B e Nível A

- Análise de Desempenho de Processos

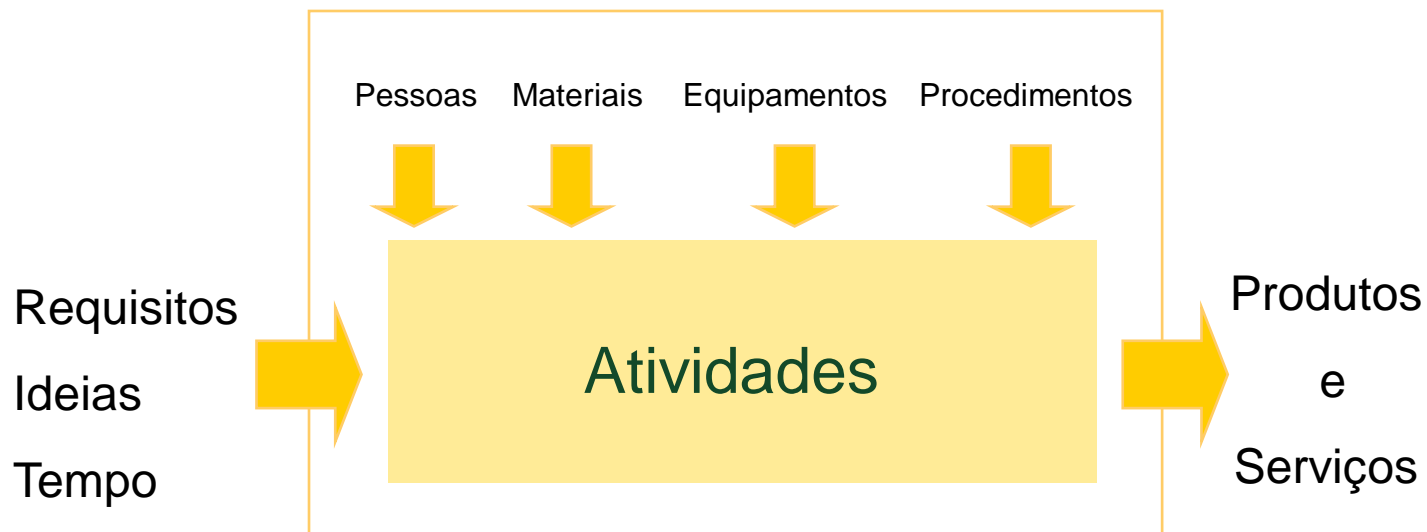
- Gerência Quantitativa de Projetos

- Análise de causas / Melhoria contínua e Inovação

Definições

Processo de Software

- Um conjunto de atividades inter-relacionadas ou interativas, que transforma insumos (entradas) em produtos (saídas) (ISO 9000, 2000)



Por que definir processo? (1/2)

- “O uso de um processo de software bem definido (automatizado ou não) leva à **redução dos custos** de produção, bem como à **melhoria da qualidade e integridade do software**”

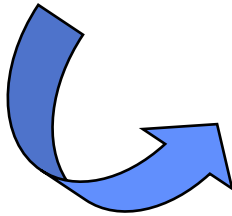
Por que definir processo? (2/2)

- Razões para Modelar o Processo de Software
 - Formar um entendimento comum
 - Encontrar inconsistências, redundâncias e omissões
 - Encontrar e avaliar atividades propostas mais adequadas aos objetivos
 - Fazer um processo geral para uma situação particular na qual ele será utilizado
 - Benefícios da Execução e outras ferramentas

Processo imaturo

❑ Características

- *Ad hoc* - Improvisado
- Fortemente dependente dos profissionais
- Indisciplinado



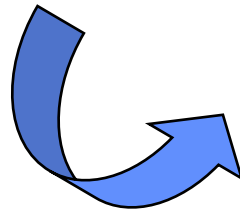
❑ Consequências

- Pouca produtividade
- Qualidade de difícil previsão
- Alto custo de manutenção
- Risco na adoção de novas tecnologias

Processo maduro

❑ Características

- Processo conhecido por todos
- Apoio visível da alta administração
- Auditoria da fidelidade ao processo
- Medidas do produto e do processo
- Adoção disciplinada de tecnologias



❑ Consequências

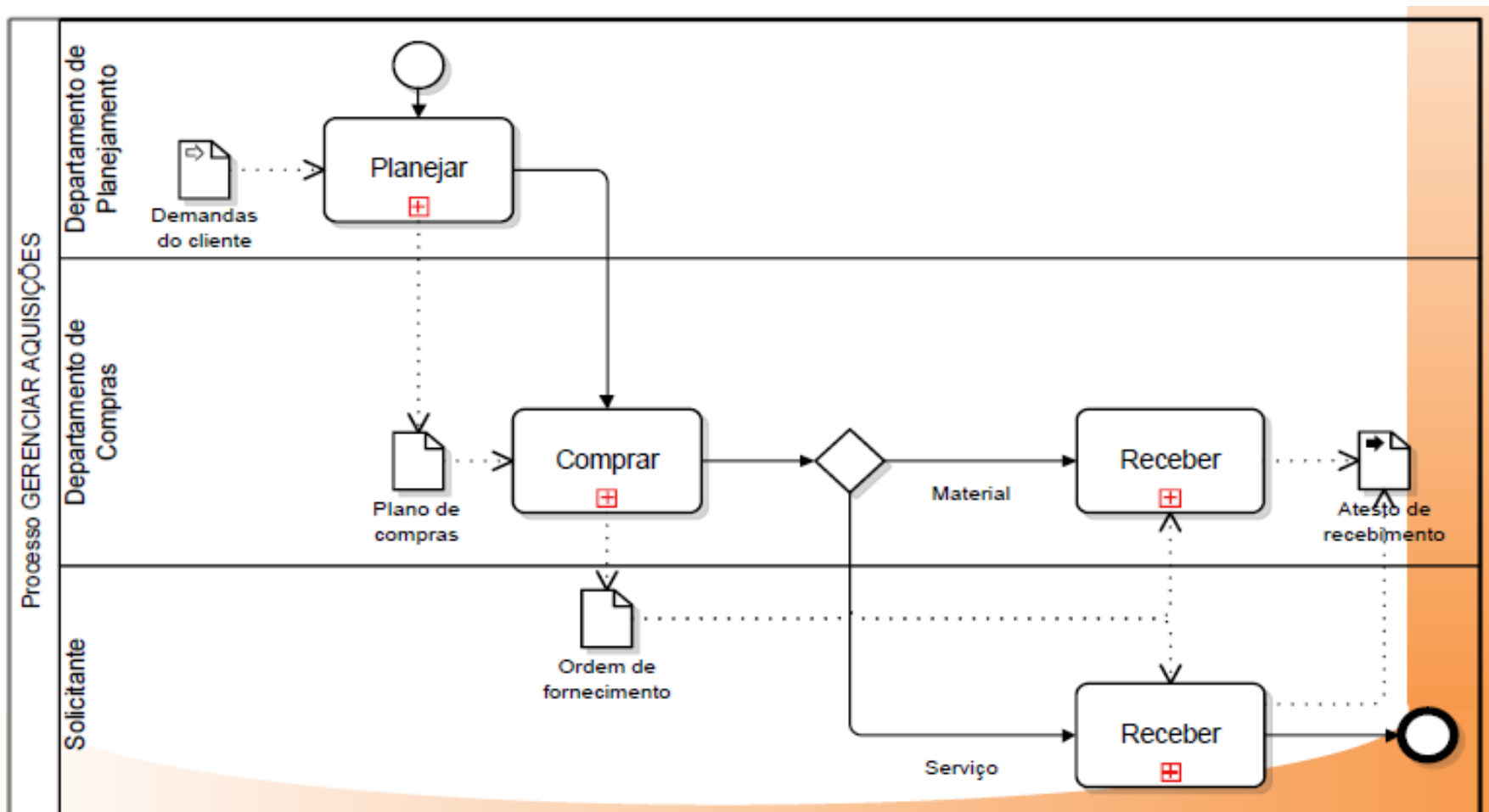
- Papéis e responsabilidades claramente definidos
- Acompanhamento da qualidade do produto e da satisfação do cliente
- Expectativas para custos, cronograma, funcionalidades e qualidade do produto é usualmente alcançada

Processo de negócio x Processo de desenvolvimento (1/3)

- Processo de Negócio:
 - Um grupo de atividades relacionadas de forma lógica que usa os recursos da organização para fornecer resultados definidos em apoio aos objetivos da organização
 - Foco nos objetivos estratégicos da organização

Processo de negócio x Processo de desenvolvimento (2/3)

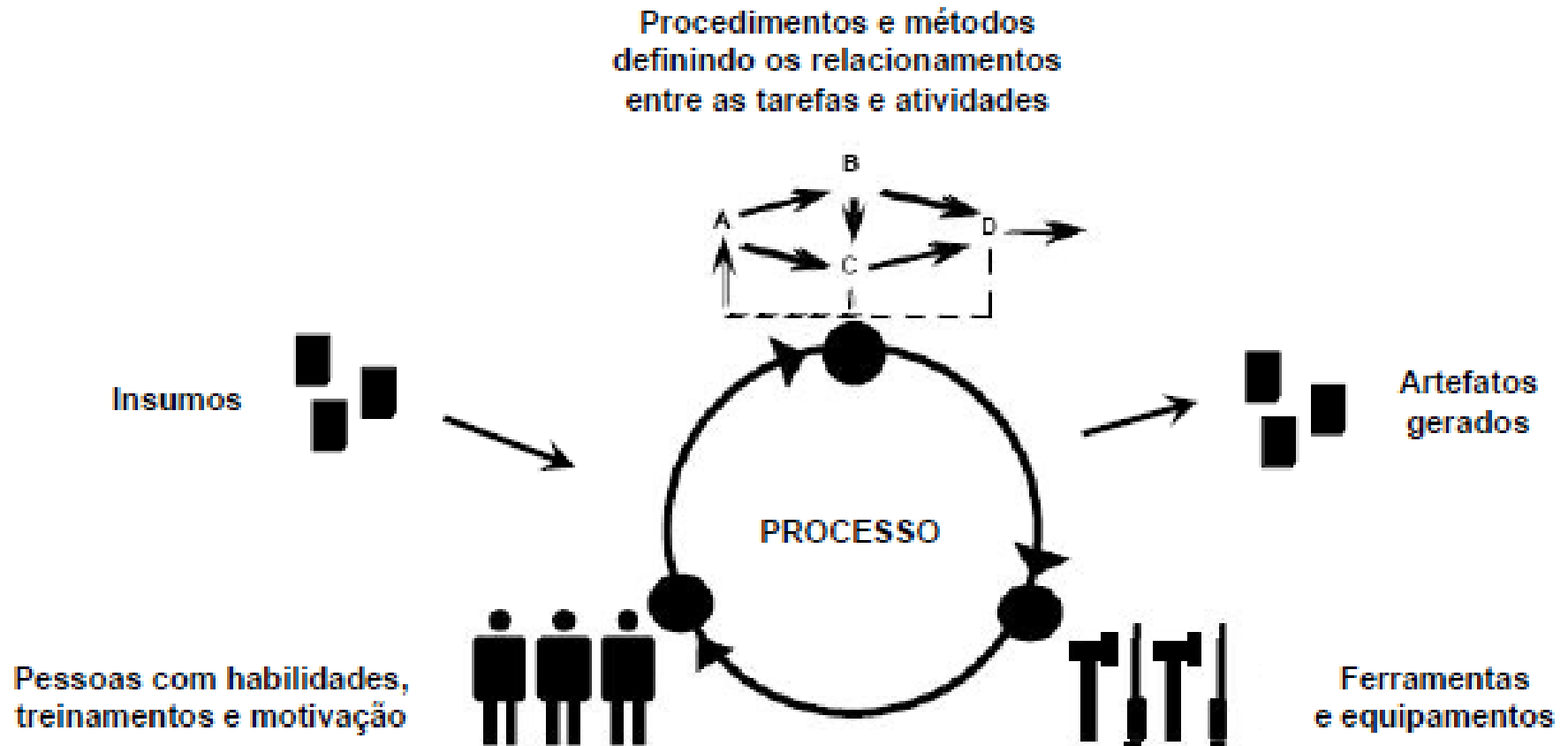
- Exemplo de processo de Negócio:



Processo de negócio x Processo de desenvolvimento (3/3)

- Processo de desenvolvimento
 - Um conjunto de passos ordenados e executados com o objetivo de alcançar uma meta
 - No caso de desenvolvimento de software, a meta consiste em criar um software ou desenvolver um já existente
 - Pode ser visto como um processo de negócio aplicado a objetivos específicos

Componentes do processo (1/5)



Componentes do processo (2/5)

- Insumo
 - Elemento necessário para a realização de uma tarefa ou atividade
 - Pode ser um elemento de saída de outras atividades ou tarefas
- Artefato
 - É um resultado de uma atividade
 - Ex.: Documento revisto e aceito, módulo implementado, testado e aceito
 - Quando entregue ao usuário (cliente) o artefato é um produto

Componentes do processo (3/5)

- Tarefa
 - É uma ação desempenhada por alguma pessoa visando a realização ou monitoramento do projeto
 - Consome recursos - consumo real
 - Esforço (tempo de pessoa)
 - Equipamento
 - Financeiro

Componentes do processo (4/5)

- Atividade
 - Conjunto de tarefas que levam a um ou mais artefatos de qualidade controlada
 - Representa uma evidência de progresso no desenvolvimento
 - O esforço é medido através das tarefas constituintes

Componentes do processo (5/5)

- Ferramentas e equipamentos
 - Auxiliam a execução das atividades e tarefas dos processos
 - Podem automatizar partes da execução das atividades e tarefas
- Papel
 - Descreve como as pessoas se comportam no processo e quais são as responsabilidades que elas têm
 - Requer habilidades específicas necessárias
 - **Papéis não são pessoas**
 - pessoas executam papéis

Notações

Formatos para definir processos

- Textual

Nome	Identificação e registro de requisitos
Descrição	<p>Nesta atividade deve-se realizar o levantamento de requisitos com o cliente (ou fornecedor de requisitos sugerido por ele), preferencialmente por meio de entrevistas e análise de documentação fornecida sobre o produto. A partir desse levantamento, deve-se identificar os requisitos de cliente, os requisitos funcionais e não-funcionais de software e registrar as dependências e relacionamentos entre eles.</p> <p>Uma vez que os requisitos tenham sido formalizados pelo analista de sistemas, deve-se marcar uma reunião com a equipe do projeto para apresentação do documento de Visão e a posterior análise e aprovação dos demais participantes do projeto. Caso ajustes sejam sugeridos, devem ser resolvidos para que se registre o comprometimento da equipe, atestando que compreenderam os requisitos a serem desenvolvidos.</p> <p>Após a conclusão da atividade os documentos produzidos devem ser submetidos a aprovação da garantia da qualidade e colocados sob gerência de configuração, conforme pertinente.</p>
Critérios de Entrada	<ul style="list-style-type: none">▪ Projeto iniciado▪ Cliente (ou fornecedor de requisitos) identificado
Critérios de Saída	<ul style="list-style-type: none">▪ Levantamento dos requisitos concluído▪ Conclusão da primeira versão do documento de requisitos▪ Rastreabilidade registrada▪ Aprovação da equipe do projeto para os requisitos▪ Ajustes identificados solucionados, conforme pertinente
Responsáveis	Analista de Sistemas

- Fluxos



Textual (1/2)

- Detalha o processo; provê entendimento sem ambiguidades
- Para cada atividade do processo definir:
 - Nome
 - Descrição
 - Pré-atividade e Pós-atividade
 - Critérios de Entrada e de Saída
 - Responsável e Participantes
 - Artefatos de Entrada e de Saída
 - Ferramentas

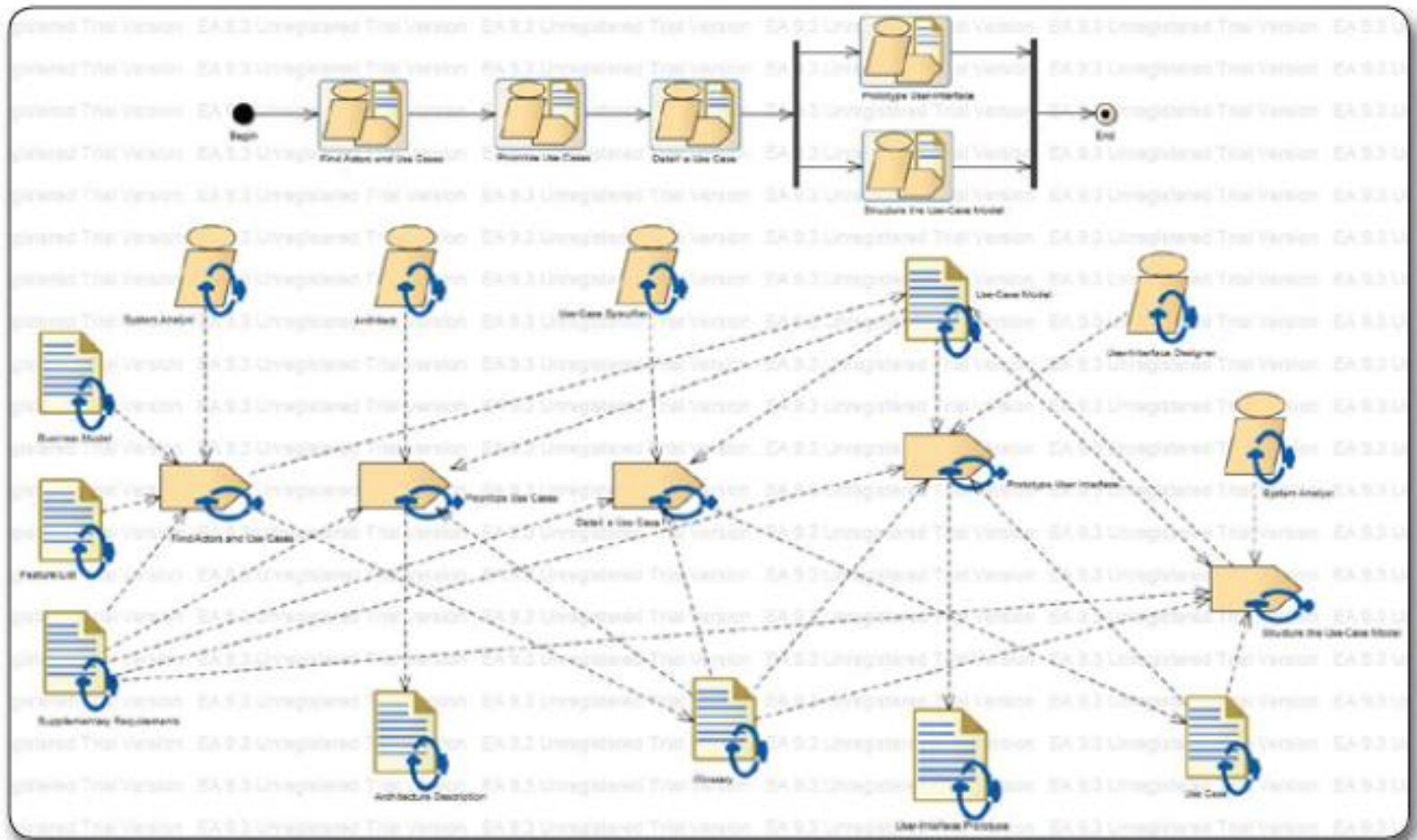
Textual (2/2)

Nome	<i>Nome da atividade</i>
Descrição	<i>Qual o propósito da atividade?</i>
Responsável(is)	<i>Quem é responsável pela execução da atividade?</i>
Participantes	<i>Quem está envolvido (mas não é o responsável direto por sua execução)?</i>
Produtos Requeridos	<i>Qual(is) as entradas requeridas para que a atividade possa ser executada?</i>
Produtos Gerados	<i>Qual(is) os produtos de trabalho criados/modificados por esta atividade?</i>
Critérios de Entrada	<i>O que deve ser garantido antes que esta atividade possa ser executada?</i>
Critérios de Saída	<i>Que itens devem ser produzidos e/ou que ações devem ser executadas para se considerar que esta atividade pode ser considerada concluída?</i>

SPEM (1/2)

- *Software Process Engineering Metamodel*
- É uma linguagem de modelagem, denominada de meta-modelo, ou seja, um conjunto de construtores e regras para definir e modelar processos de software
- Mantido pela OMG (*Object Management Group*)
 - <http://www.omg.org/spec/SPEM/2.0/>

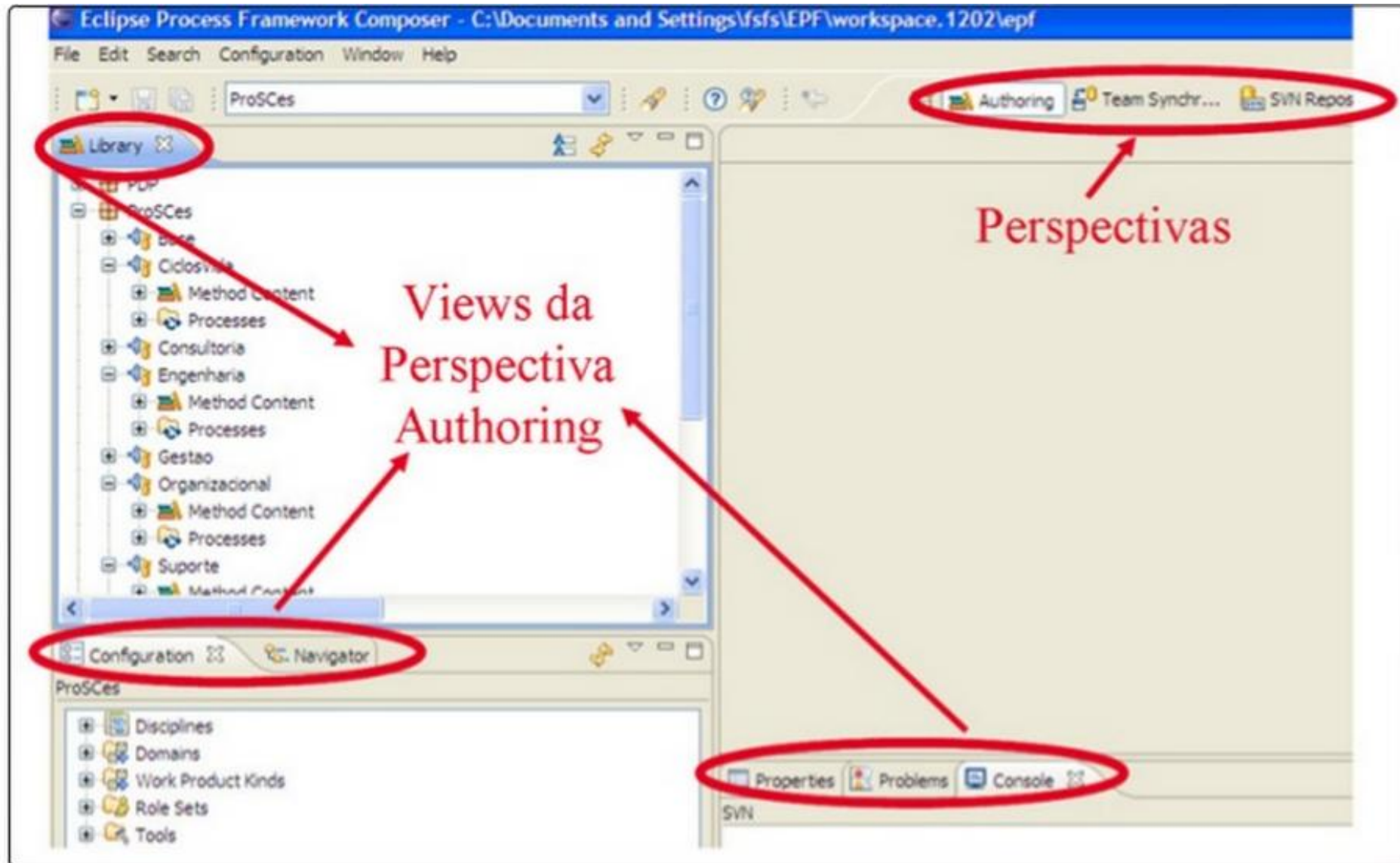
SPEM (2/2)



EPF Composer (1/3)

- *Eclipse Process Framework*
 - Plugin para o Eclipse
- É um framework para criação e descrição de processos de desenvolvimento de software customizados
- Mais informações em:
<http://www.eclipse.org/epf/>

EPF Composer (2/3)



EPF Composer (3/3)

The screenshot shows the Eclipse Process Framework Composer interface. The left sidebar contains a tree view of project components. Three items are circled in red and numbered 1, 2, and 3:

- 1. **Adaptações do Programa Ut** (selected)
- 2. **Adaptações de Análise e Projeto**
- 3. **Artefatos de saída do processo de Análise e Projeto**

The main area displays the configuration for 'Artefatos de saída do processo de Análise e Projeto'. It includes a title bar, a subtitle, and a table of artifacts.

Níveis de Controle dos Artefatos do Processo de Análise e Projeto - adaptações para o Programa

Descrição Principal

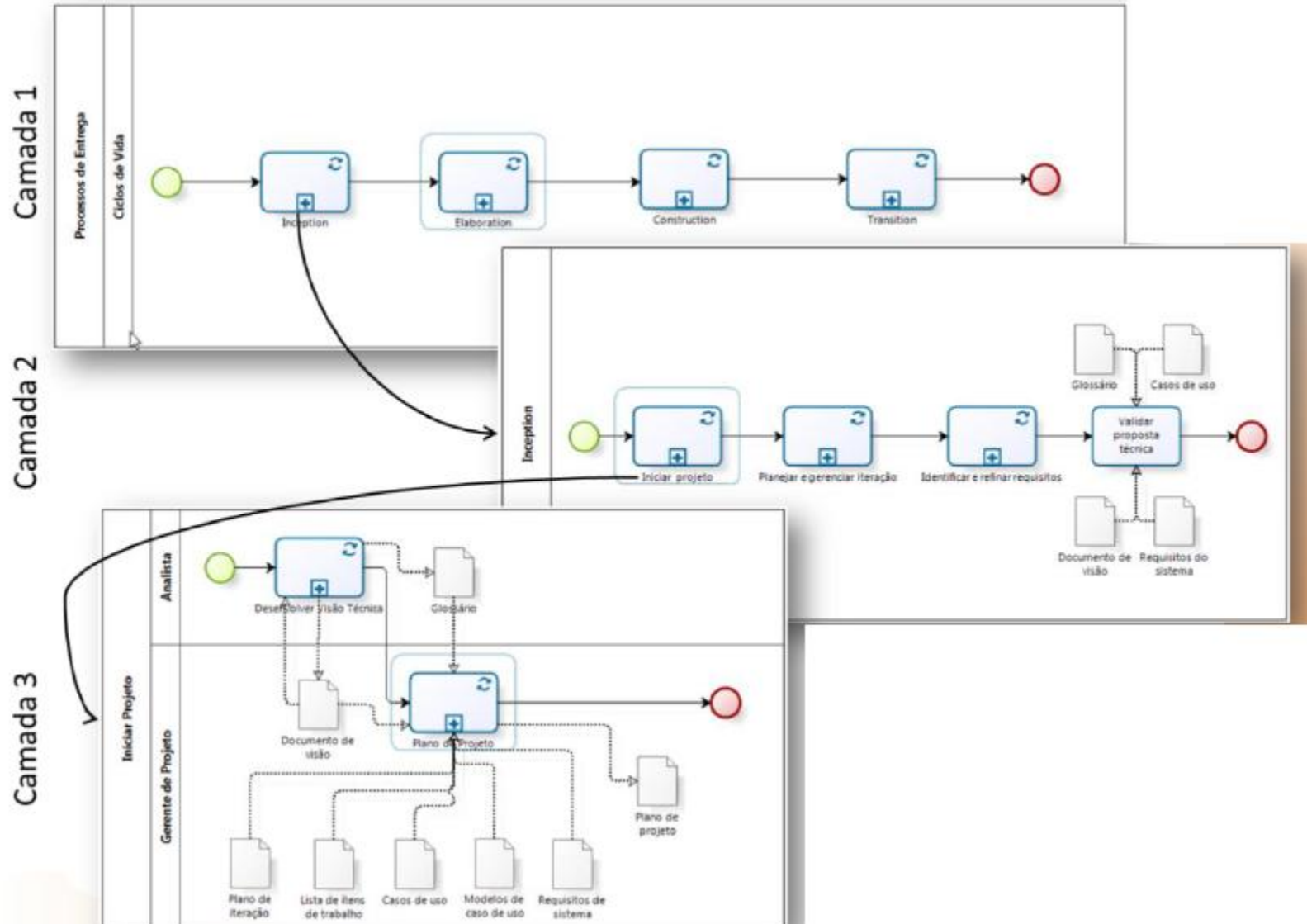
Artefatos de Análise e Projeto	Nível de Controle	Técnica para Revisão Cheia (XX.00)	Técnica para Revisão Intermediária (XX.YY)	Aprovador
Documento de Arquitetura	1	INSP	IPDC	NA
Padrões Arquiteturais	2	NA	NA	NA
Modelo de Projeto	3	INSP	IPDC	NA
Engagement Plan	2	IPDC	IPDC	Gerente de Projeto Líder Técnico
Modelo de Dados	3	NA	NA	NA
Especificação de Componentes	1	PDC	IPDC	NA
Modelo de Dados	2	PDC	IPDC	NA

Nível de Controle	Descrição
1 - Controlado e Gerenciado	É obrigatória a abertura de uma "Solicitação de Mudança" na ferramenta de controle de mudanças.
2 - Controlado	O controle é feito através de um histórico de revisões no próprio artefato.
3 - Backup	Este nível não exige controle versão das alterações.

BPMN (1/2)

- Notação para modelagem de processos de negócio, mas também utilizado para processo de software
 - Mais visual
- Mais informação em: <http://www.bpmn.org>
- BizAgi
 - Ferramenta mais utilizada

BPMN (2/2)



Processo Unificado

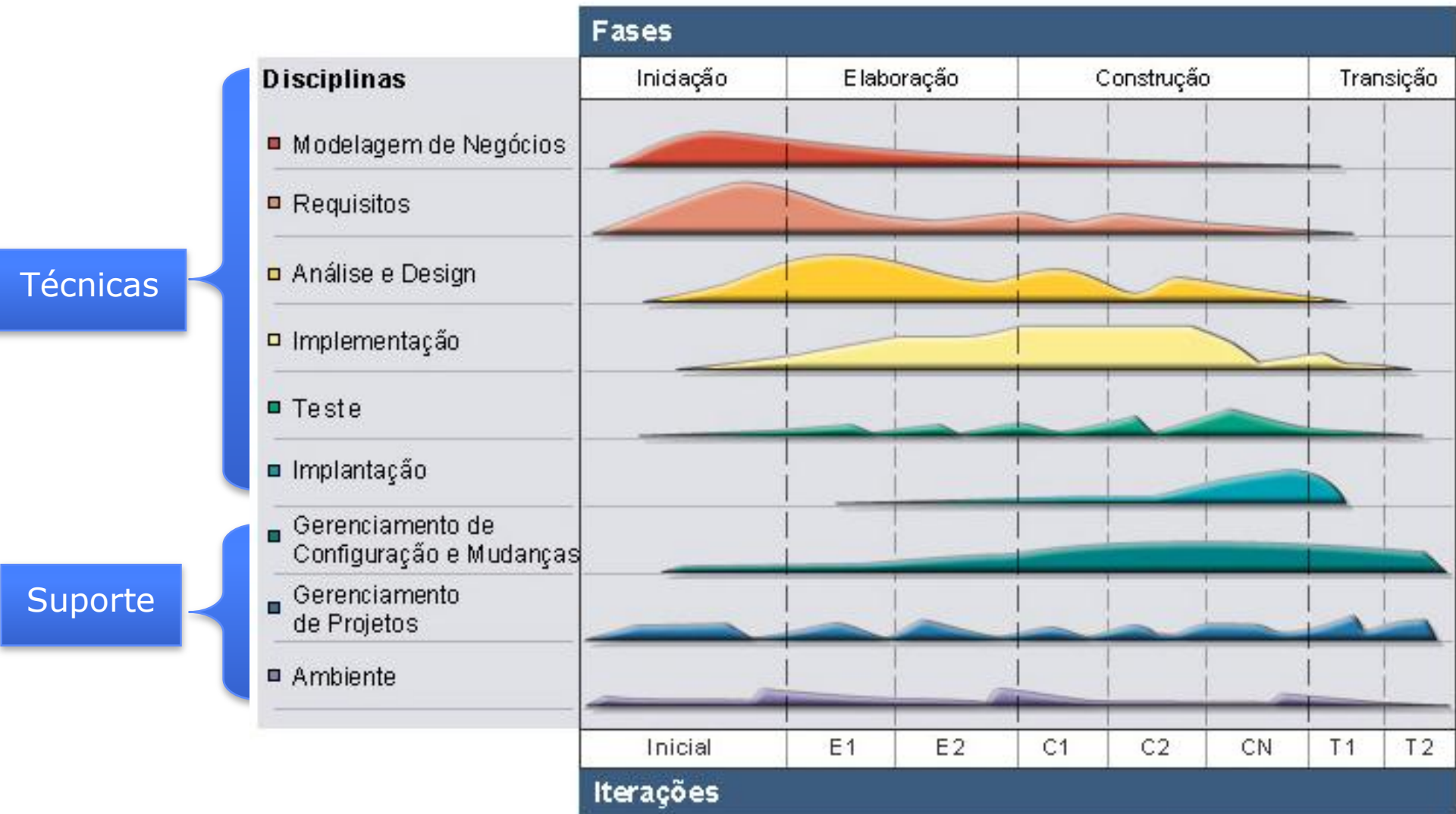
O que é?

- *Rational Unified Process* – RUP
 - Tentativa de obter o que há de melhor em cada modelo de ciclo de vida
- Oferece uma abordagem baseada em disciplinas
 - Atribui tarefas e responsabilidades dentro de uma organização de desenvolvimento

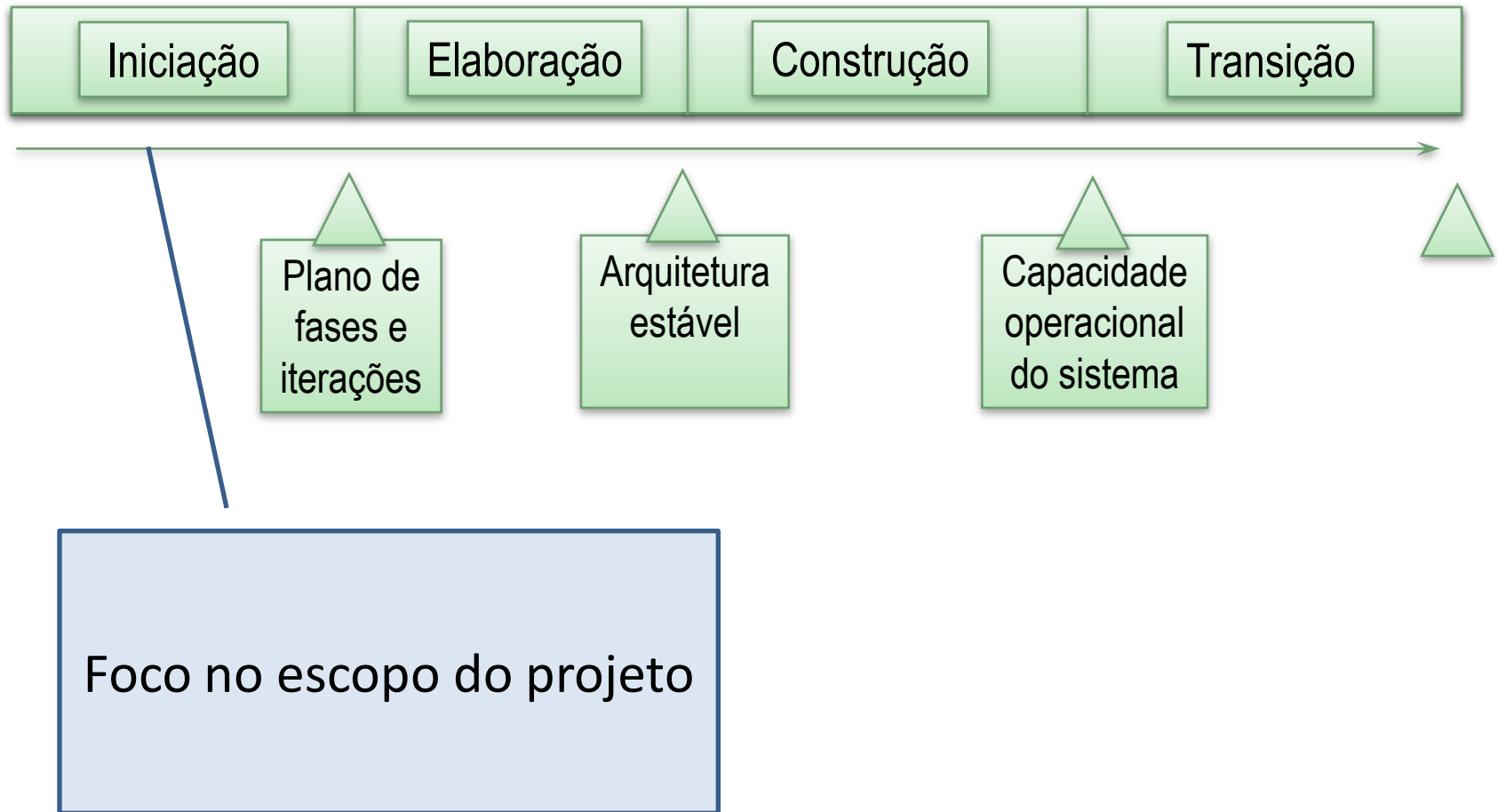
Objetivo

- Construir o sistema em várias iterações durante diferentes fases de desenvolvimento
- Benefícios esperados
 - Mitigação de riscos precoce
 - Visibilidade do progresso
 - Envolvimento e comprometimento do usuário
 - Controle sobre a complexidade
 - Aprendizado incremental
 - Menos defeitos

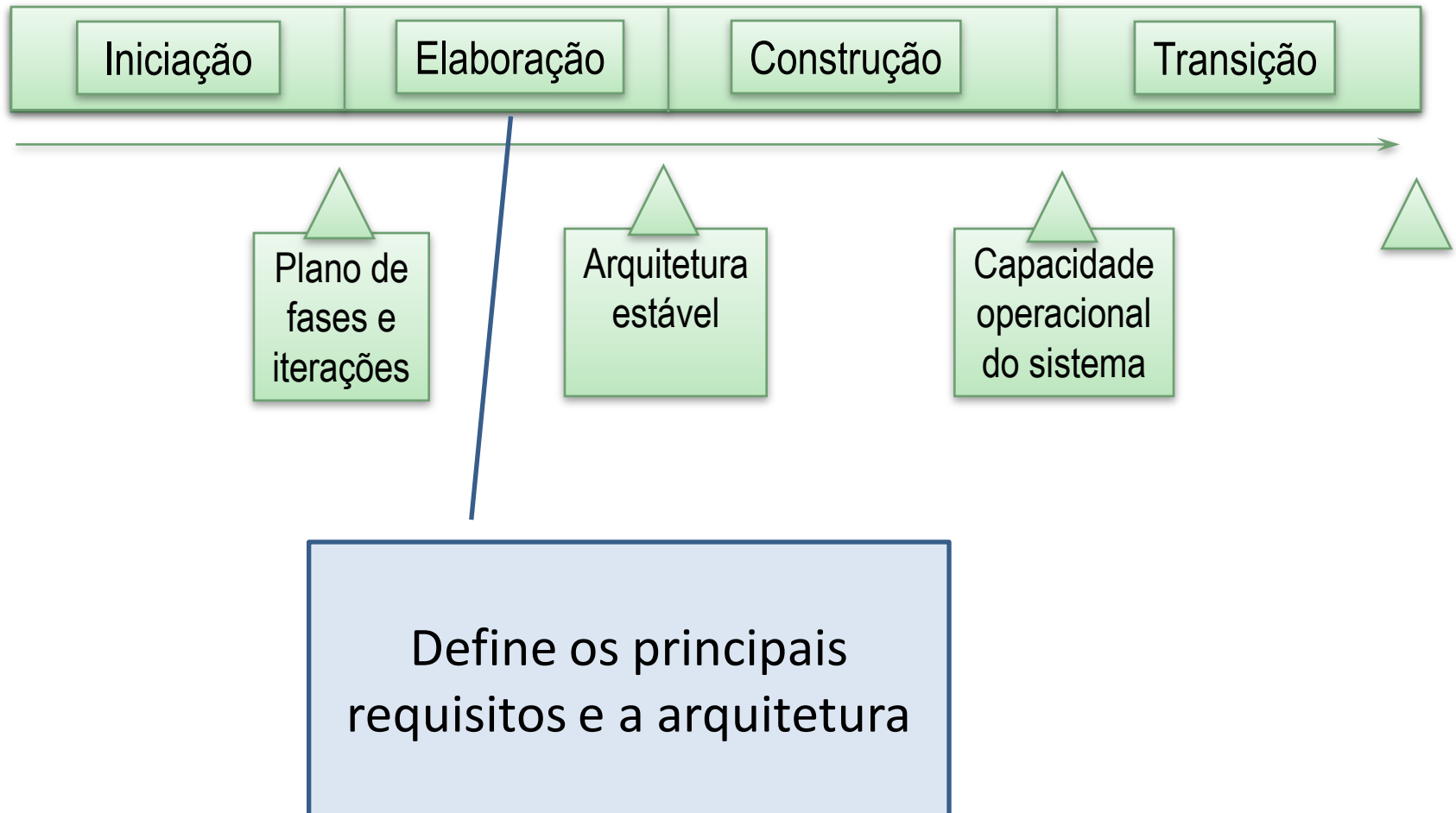
Estrutura (1/3)



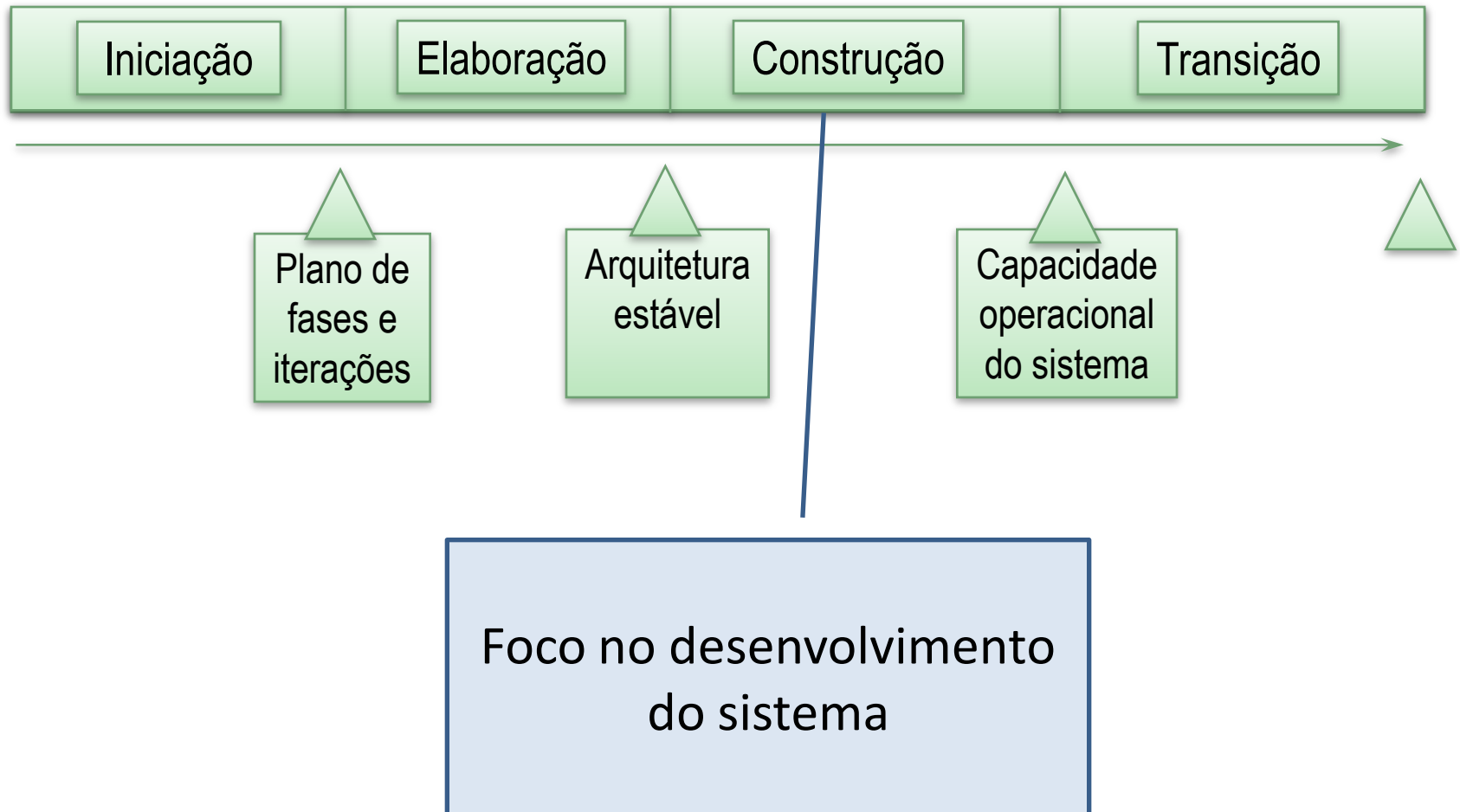
Estrutura (2/3)



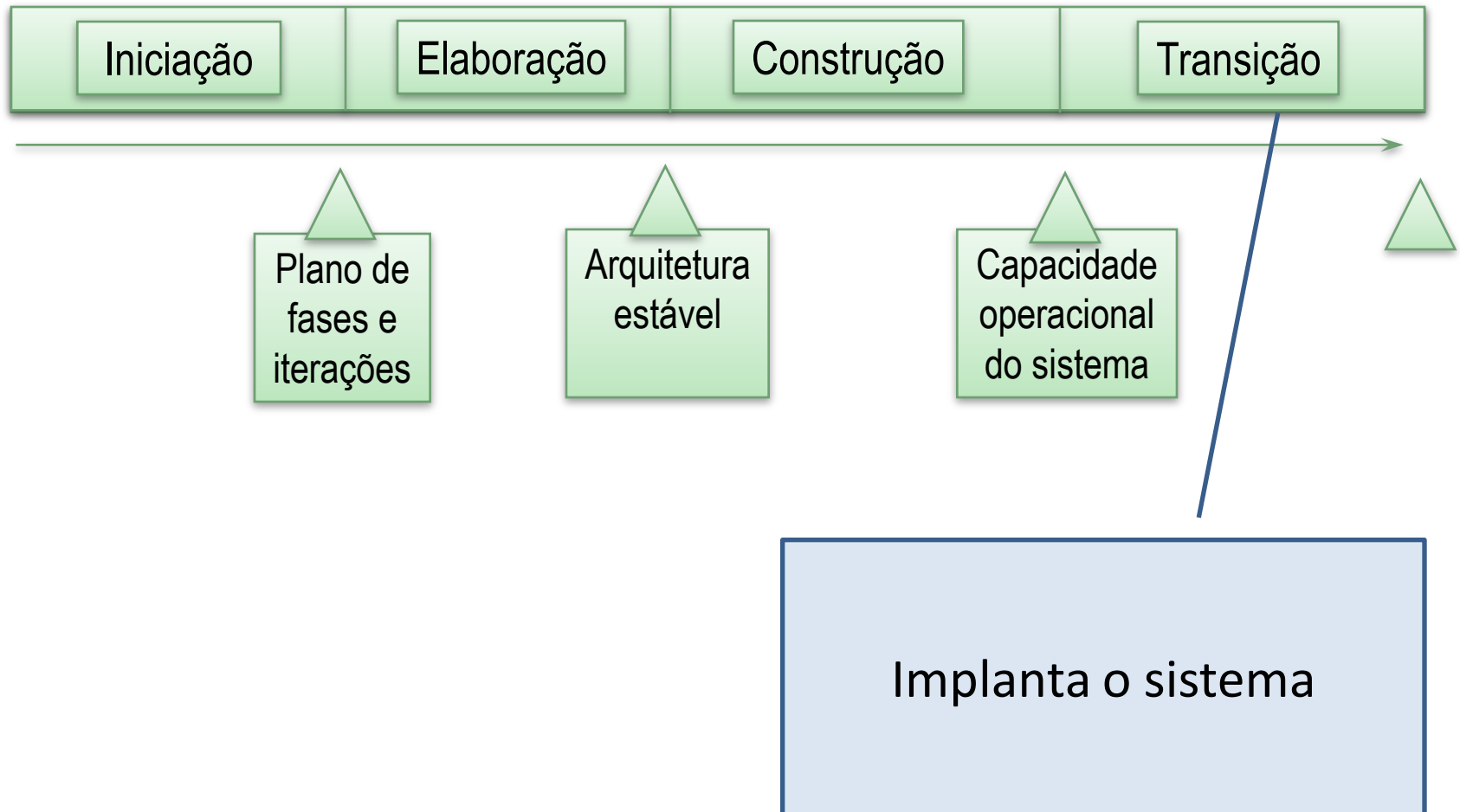
Estrutura (2/3)



Estrutura (2/3)

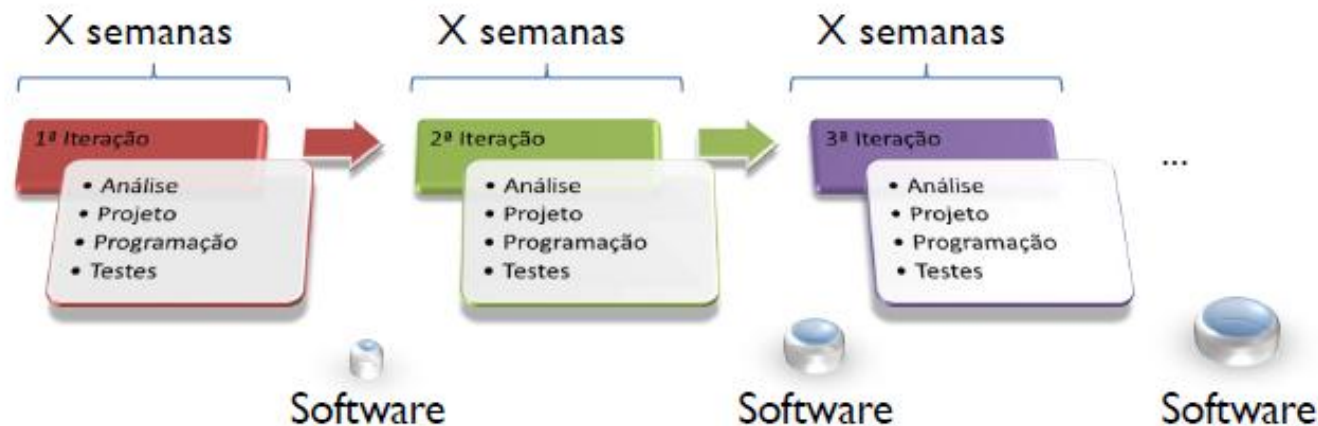


Estrutura (2/3)



Estrutura (3/3)

- O desenvolvimento é organizado em “mini-projetos”
 - Cada “mini-projeto” é uma iteração
 - Cada iteração tem duração curta e fixa (de 2 a 6 semanas)
 - Cada iteração tem atividades de análise, projeto, programação e testes
 - O produto de uma iteração é um software parcial



Principais características

- O resultado de cada iteração é um software incompleto
 - Em desenvolvimento (não pode ser colocado em produção)
- Avaliação parcial no final de cada iteração
- Avaliação contínua dos riscos do projeto
- Podem ser necessárias diversas iterações (10 a 15) para ter uma versão do sistema pronta para entrar em produção
- Fortemente orientado na utilização de casos de uso (UML - *Unified Modeling Language*)
- Centrado em arquitetura

Exemplo

- Analisar os requisitos no início do projeto
 - Casos de uso e Lista de requisitos não funcionais
- Priorizar os casos de uso
 - Significativos para a arquitetura como um todo
 - Alto valor de negócio e Alto risco
- Em cada iteração
 - Selecionar alguns casos de uso por ordem de prioridade para serem analisados em detalhes
 - Atribuir tarefas para a iteração a partir da análise detalhada desses casos de uso
 - Fazer projeto e programação de parte do software
 - Testar a parte do software recém projetada e programada e criar a baseline da iteração
 - Apresentar a baseline da iteração ao usuário

Desenvolvimento ágil

O que é?

- São dadas respostas rápidas e flexíveis a mudanças
 - O projeto é replanejado continuamente
 - São feitas entregas incrementais e constantes do software, refletindo as mudanças solicitadas

Manifesto Ágil (1/2)

Indivíduos e interações

sobre

Processos e ferramentas

Software funcionando

sobre

Documentação abrangente

Colaboração com o cliente

sobre

Negociação de contrato

Responder a mudanças

sobre

Seguir um plano

- “Mesmo havendo valor nos itens à direita, valorizamos mais os itens à esquerda.”

Manifesto Ágil (2/2)

- 12 Princípios ágeis
 - Satisfazer o cliente
 - Acolher modificações nos requisitos
 - Entregar o software com frequência
 - Trabalhar junto ao cliente
 - Manter os indivíduos motivados
 - Promover conversas face a face
 - Medir o progresso com software funcionando
 - Manter um ritmo constante de trabalho
 - Cuidar da qualidade
 - Buscar por simplicidade
 - Trabalhar com equipes auto-organizadas
 - Ajustar o comportamento da equipe buscando mais efetividade

Métodos ágeis

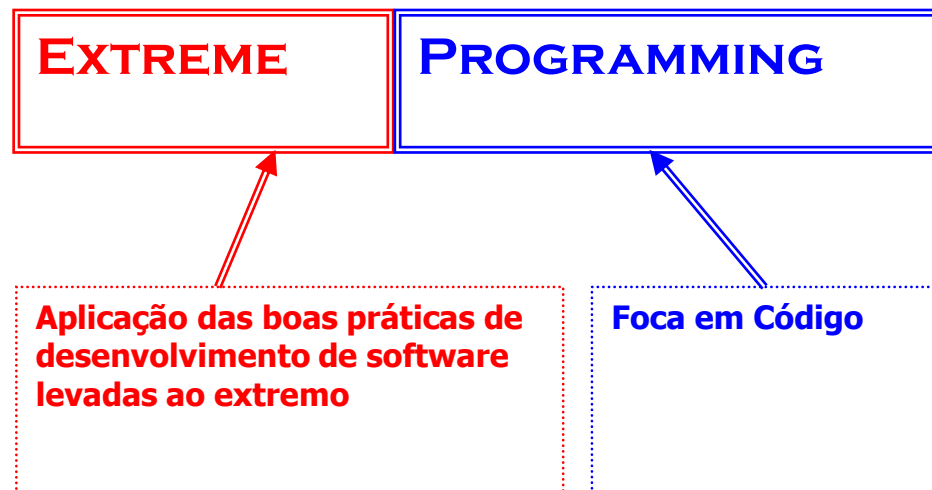
- XP (*Extreme Programming*)
- Lean Programming
- DAS (Desenvolvimento Adaptativo de Software)
- DSDM (Método de Desenvolvimento Dinâmico de Sistemas)
- Scrum
- Crystal
- FDD (Desenvolvimento Guiado por Características)

Métodos ágeis

- **XP (*eXtreme Programming*)**
- Lean Programming
- DAS (Desenvolvimento Adaptativo de Software)
- DSDM (Método de Desenvolvimento Dinâmico de Sistemas)
- **Scrum**
- Crystal
- FDD (Desenvolvimento Guiado por Características)

XP – eXtreme Programming (1/2)

- Metodologia ágil bastante utilizada, desenvolvida para:
 - Equipes médias e pequenas (2 a 12 pessoas)
 - Requisitos vagos e em constante evolução



XP – eXtreme Programming (2/2)

- Principais características
 - Cliente disponível ou presente no dia-a-dia do projeto
 - Reuniões diárias de 15 a 20 minutos
 - Programação em par
 - Dois desenvolvedores trabalham no mesmo problema
 - Código Coletivo: todos conhecem o código
 - Releases curtos: entregas frequentes para o cliente
 - Desenvolvimento guiado por testes
 - Integração contínua

Scrum (1/2)

- Abordagem ágil para o gerenciamento de projetos
 - Mais voltado para o gerenciamento de projetos com equipes pequenas e requisitos instáveis ou desconhecidos
- Espinha dorsal: *Sprint*
 - Ideia: Iterações curtas e focadas melhoram a visibilidade para acompanhar o desenvolvimento

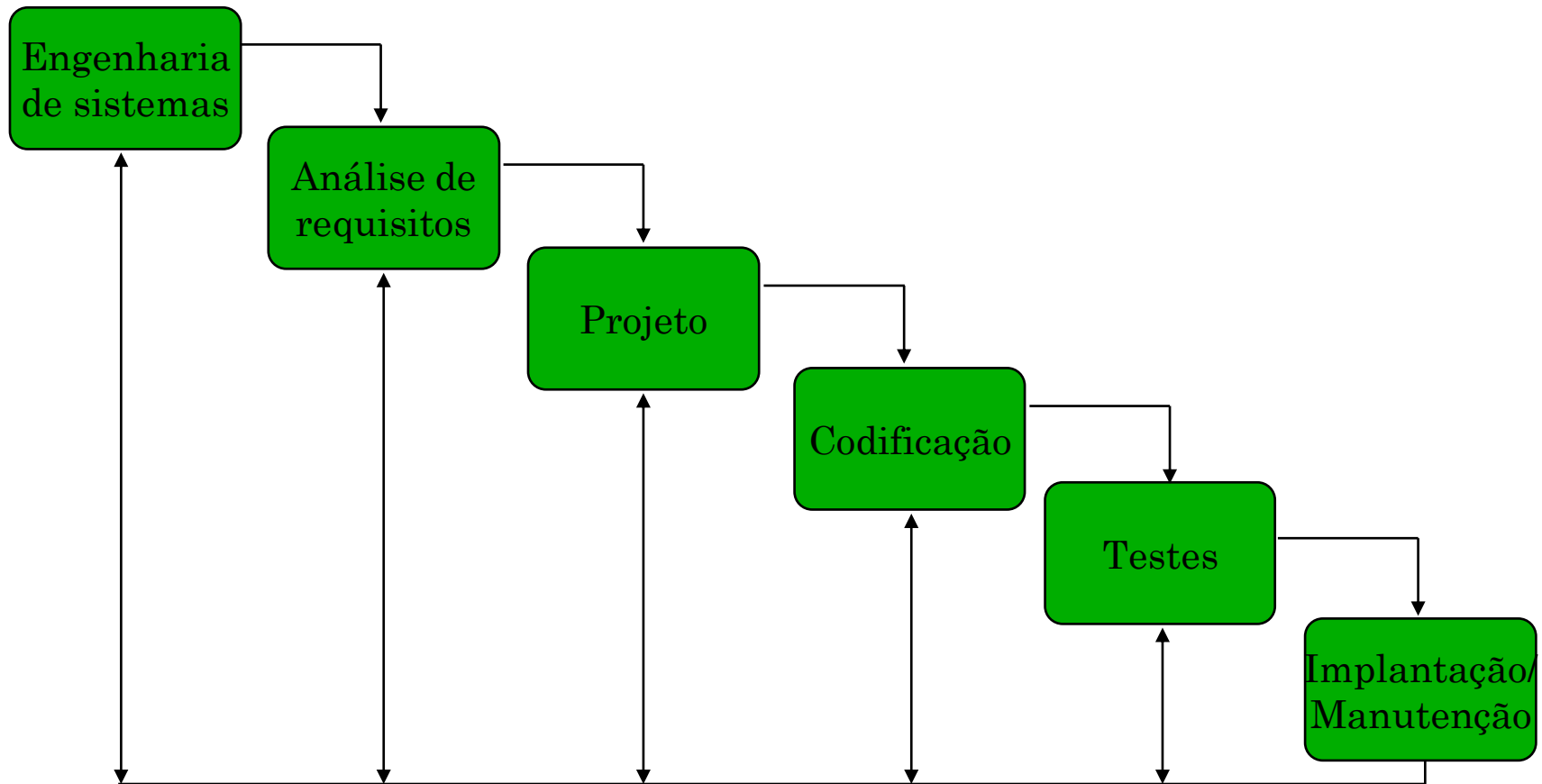
Scrum (2/2)

- Principais características:
 - Pequenas equipes de trabalho organizadas para maximizar comunicação e minimizar supervisão
 - Equipes sem nível hierárquico nem papéis

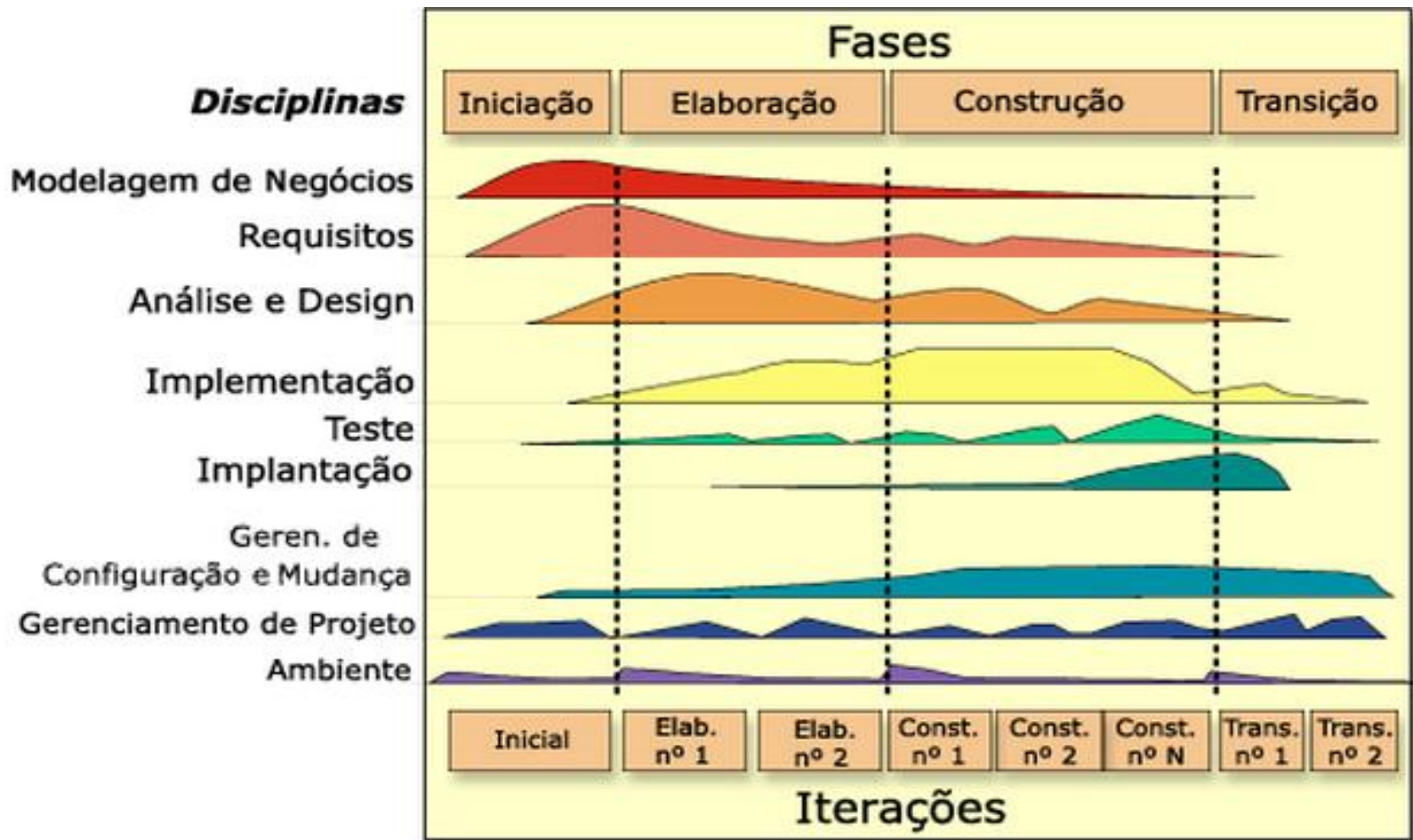


Exemplos

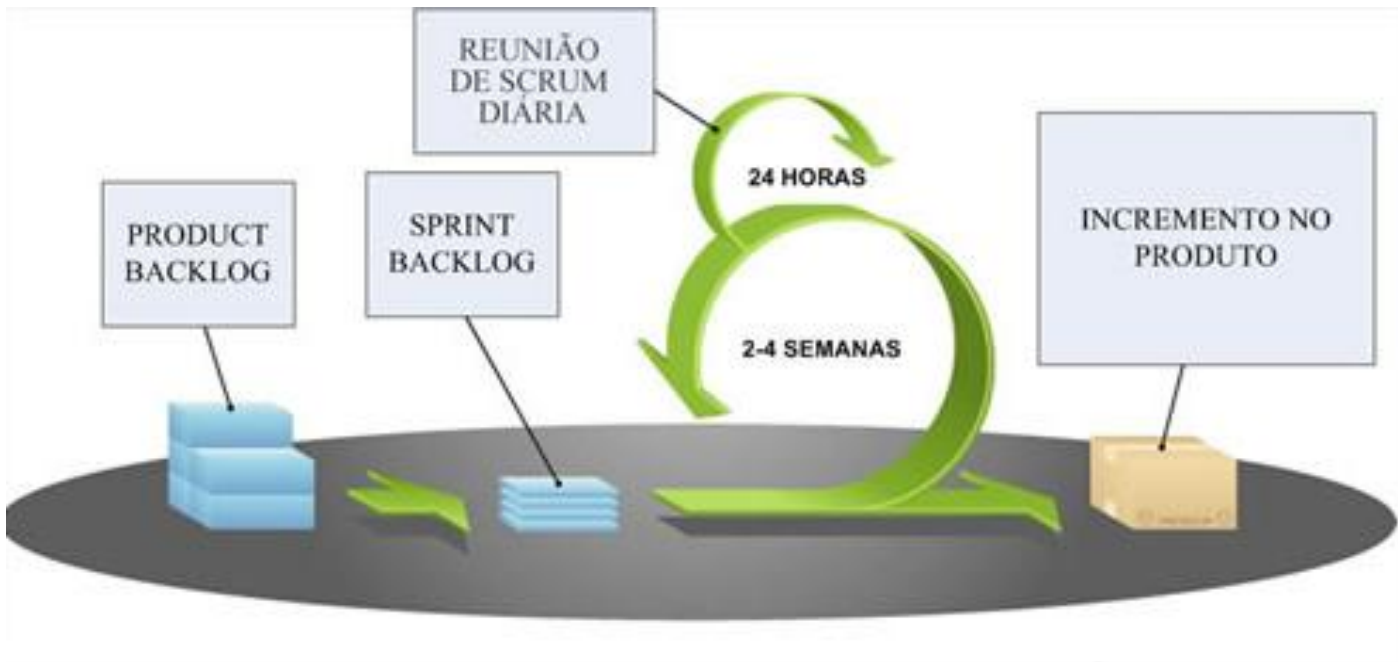
“Exemplos” genéricos (1/3)



“Exemplos” genéricos (2/3)



“Exemplos” genéricos (3/3)



Exemplos de descrição

Nome	Identificar e registrar requisitos
Descrição	<p>Nesta atividade deve-se realizar o levantamento de requisitos com o cliente (ou fornecedor de requisitos sugerido por ele), preferencialmente por meio de entrevistas e análise de documentação fornecida sobre o produto. A partir desse levantamento, deve-se identificar os requisitos de cliente, os requisitos funcionais e não-funcionais de software e registrar as dependências e relacionamentos entre eles.</p> <p>Uma vez que os requisitos tenham sido formalizados pelo analista de sistemas, deve-se marcar uma reunião com a equipe do projeto para apresentação do documento de Visão e a posterior análise e aprovação dos demais participantes do projeto. Caso ajustes sejam sugeridos, devem ser resolvidos para que se registre o comprometimento da equipe, atestando que compreenderam os requisitos a serem desenvolvidos.</p> <p>Após a conclusão da atividade os documentos produzidos devem ser submetidos a aprovação da garantia da qualidade e colocados sob gerência de configuração, conforme pertinente.</p>
Critérios de Entrada	<ul style="list-style-type: none"> ▪Projeto iniciado ▪Cliente (ou fornecedor de requisitos) identificado
Critérios de Saída	<ul style="list-style-type: none"> ▪Levantamento dos requisitos concluído ▪Conclusão da primeira versão do documento de requisitos ▪Rastreabilidade registrada ▪Aprovação da equipe do projeto para os requisitos ▪Ajustes identificados solucionados, conforme pertinente
Responsáveis	Analista de Sistemas
Participantes	Desenvolvedor, Cliente, Gerente de projeto, Auditor de Qualidade
Pós-atividade	Detalhamento de casos de uso
Artefatos Requeridos	Variados
Artefatos Gerados	<ul style="list-style-type: none"> ▪Documento de requisitos (visão) ▪Laudos de avaliação interna ▪Registros de não conformidades ▪Registro de rastreabilidade
Ferramentas	Ferramenta de Modelagem, Word, Excel

Processo de Desenvolvimento LENS

Fase 1: Planejamento do Projeto e Especificação dos Requisitos e dos Testes do Software

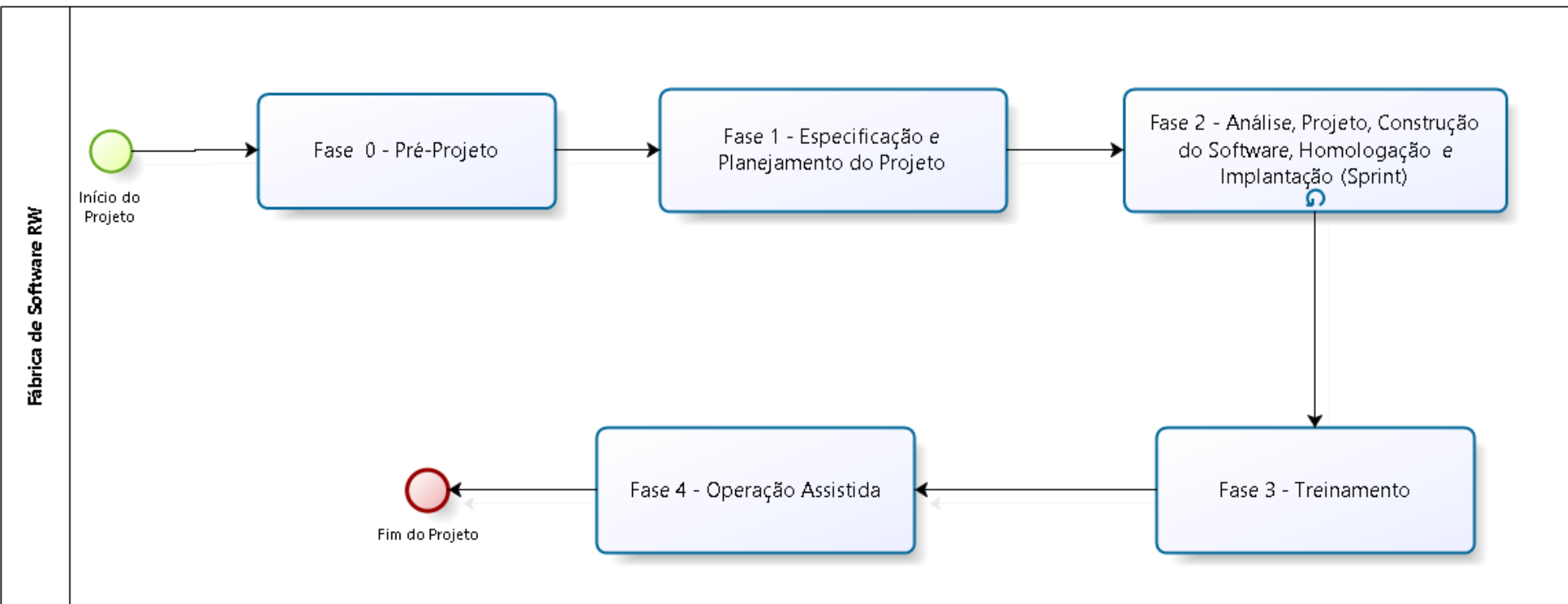
O objetivo desta fase é desenvolver o plano do projeto e as especificações dos requisitos e dos testes do software

Nome	1 Planejamento do Processo
Descrição	O objetivo desta atividade é definir o plano do processo para o projeto e instanciar o ambiente de apoio à execução do processo.

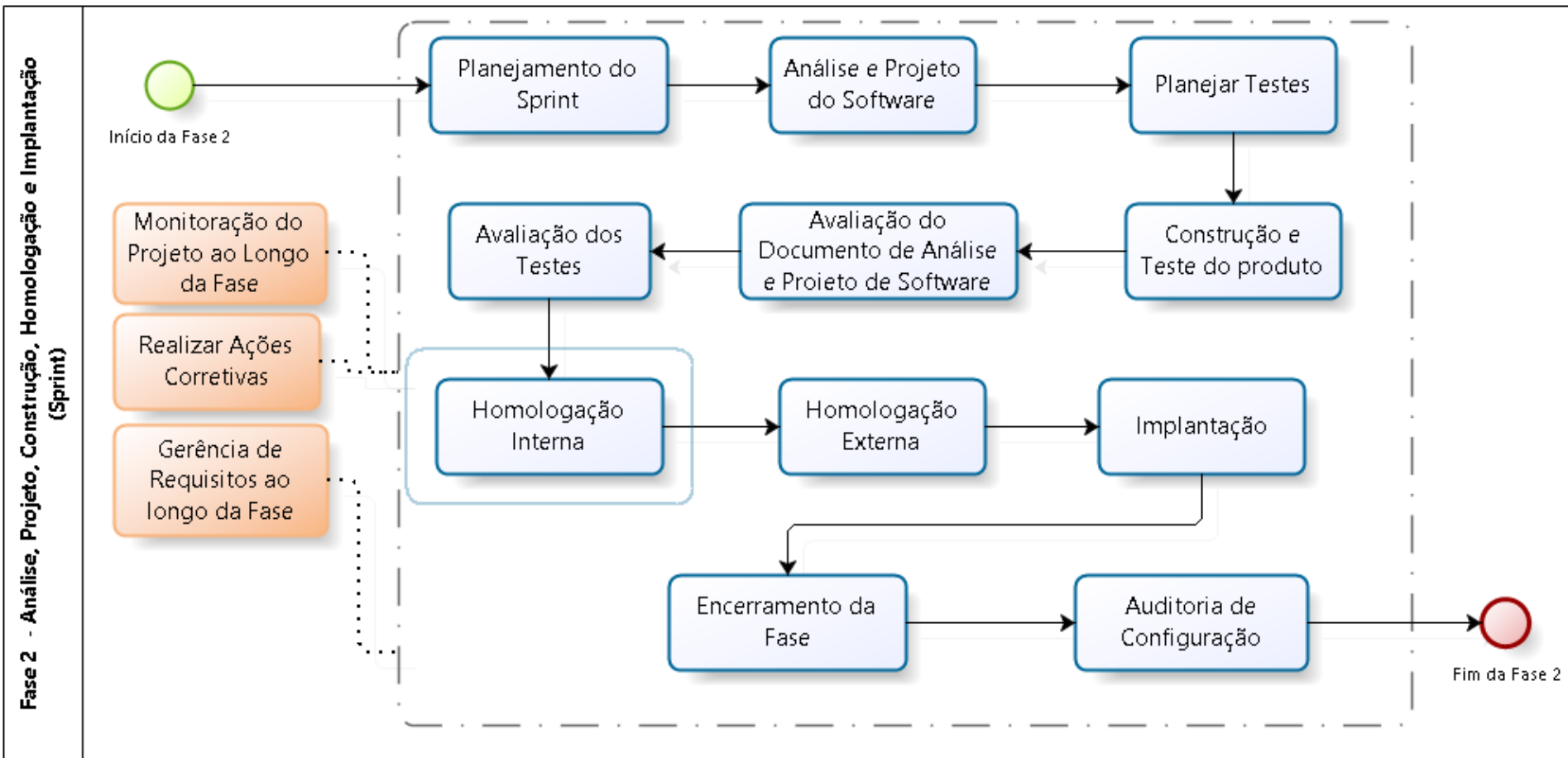
Nome	1.1 Identificar/Entender Requisitos do Cliente e Requisitos do Software
Descrição	Identificar/entender os requisitos do cliente e requisitos funcionais e não-funcionais do software. A identificação/entendimento dos requisitos deve ser realizada através de reuniões com o cliente. Os requisitos funcionais do software devem ser identificados considerando que cada requisito será especificado através de um caso de uso. O diagrama de casos de uso também deve ser elaborado nesta atividade e documentado no documento de Levantamento de Requisitos e Modelo de Análise e Projeto. As atas de reunião de identificação/entendimento dos requisitos devem ser registradas. Caso a identificação dos requisitos tenha sido realizado previamente, um aprofundamento dos requisitos que fazem parte do escopo do projeto deve ser realizado nesta atividade.
Tipo	Gerência de requisitos
Crítérios de Entrada	Sempre que houver necessidade de se executar um ou mais ciclos de desenvolvimento para geração de um produto.
Crítérios de Saída	Levantamento de Requisitos elaborado.
Responsáveis	Analista de Sistemas
Participantes	Cliente
Pré-atividade	-
Artefatos Requeridos	Solicitação de Serviço; Levantamento Requisitos; Proposta de Desenvolvimento; Proposta Técnica e Comercial;
Artefatos Gerados	Levantamento de Requisitos; Modelo de Análise e Projeto; Ata de reunião;
Ferramentas	MS Word; Enterprise Architect

Processo de Desenvolvimento Empresa X (1/4)

- Práticas ágeis

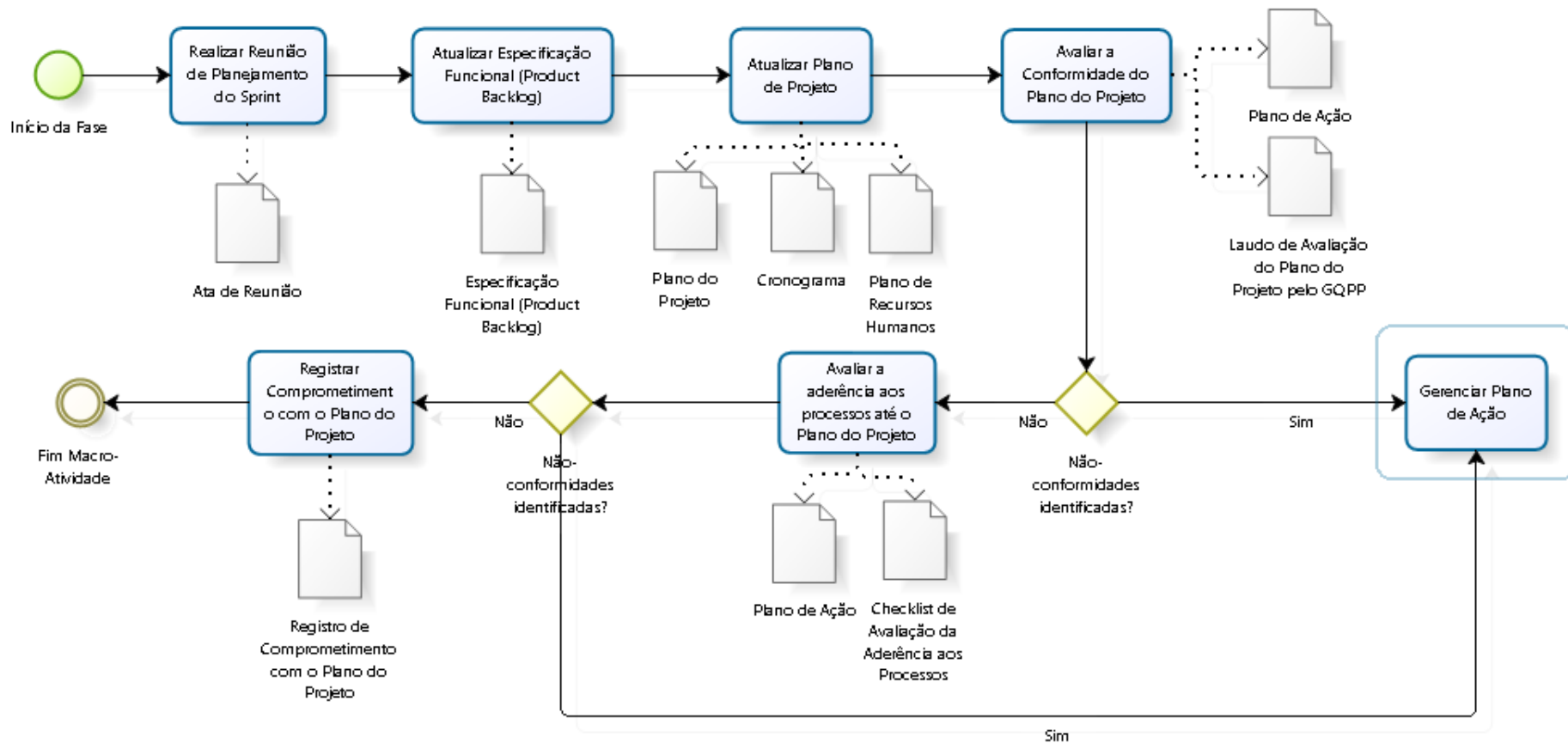


Processo de Desenvolvimento Empresa X (2/4)



Processo de Desenvolvimento Empresa X (3/4)

Planejamento do Sprint



Processo de Desenvolvimento Empresa X (4/4)

Basic **Extended** Advanced Presentation Action

[Add New Extended Attribute](#)

Descrição



Realizar reunião para planejamento da condução do sprints. Os seguintes pontos devem ser abordados durante esta reunião: requisitos acrescentados, cancelados ou modificados; prioridade dos requisitos; requisitos que serão implementados; objetivo do sprint; data das homologações; soluções para os requisitos; como demonstrar cada requisito para o cliente etc.

CrITÉrios de Entrada

Plano do Projeto em versão anterior e estar se iniciando uma nova fase

CrITÉrios de Saída

Planejamento do sprint discutido

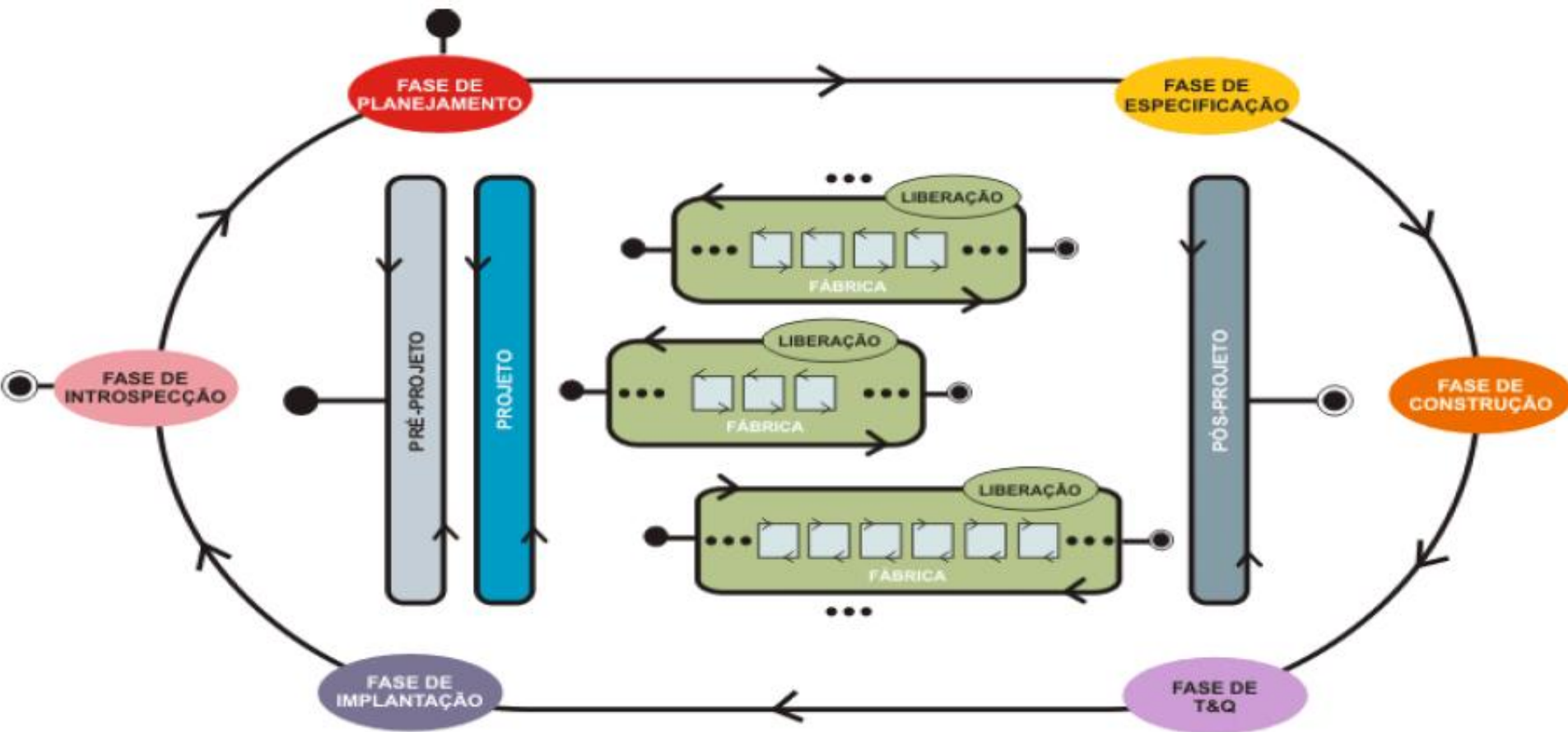
Responsáveis

- | | | | |
|-----------------------------------|---|--|---|
| <input type="checkbox"/> Diretor | <input type="checkbox"/> Gerente da Fábrica | <input checked="" type="checkbox"/> Coordenador do Projeto | <input type="checkbox"/> Analista Desenvolvedor |
| <input type="checkbox"/> GQPP | <input type="checkbox"/> GCO | <input type="checkbox"/> MED | <input type="checkbox"/> Analista de Testes |
| <input type="checkbox"/> Clientes | <input type="checkbox"/> Designados para a Tarefa | | |

Processo de Desenvolvimento Athenas

- RUP + Scrum

[Silva *et al.*, 2011]



Trabalho

Definição

- O trabalho consistirá no desenvolvimento de um software para determinado contexto, aplicando os conceitos de Engenharia de Software apresentados em sala
- Realizado em grupos de 6 alunos (um grupo será com 5)

Sistemas

- S1 – Ferramenta de apoio para gerência de projetos
- S2 – Ferramenta de apoio para medição
- S3 – Ferramenta de apoio para auditorias de qualidade
- S4 – Ferramenta de apoio para gerenciamento do conhecimento
- S5 – Ferramenta para acompanhamento de implementação de iniciativas de melhoria

Equipe

- Principais Papéis
 - Engenheiro de processos
 - Gerente de projetos
 - Analista de sistemas
 - Desenvolvedor
 - Testador
 - Analista de gerência de configuração
 - Analista de medição
 - Auditor da qualidade

Etapa 1 (1/2)

- Dada as características do sistema a ser produzido, elaborar um processo de desenvolvimento de software que atenda aos resultados esperados do nível D do MR-MPS
- Apresentar
 - Modelo de ciclo de vida
 - Práticas de ES

Etapa 1 (2/2)

- Definir:
 - Fluxograma do processo
 - Descrição do processo
 - Nome, Descrição, Critérios de Entrada, Critérios de Saída, Responsável, Participantes, Artefatos de Entrada, Artefatos de Saída, Ferramentas
- Utilizar qualquer notação e/ou ferramenta
 - Somente uma

Entrega

- Enviar documentos produzidos via Moodle **até às 23:59 do dia 7/maio**
- Elaborar slides com resumo dos resultados e apresentar na aula do **dia 8/maio**
 - Resultados obtidos
 - Dificuldades enfrentadas
 - Dúvidas
- 20 min para cada grupo
 - Todos devem participar; nota será individual

Referências

- Slides Engenharia de Software – Professor Leonardo Murta
- Slides Engenharia de Software – Professor Marcelo Schots
- Slides Processo de Software – Professora Ana Regina Rocha
- Pressman, R.S.; “Engenharia de Software”; 6ª edição, Ed. McGraw-Hill, 2006
- Slides Processo de Software, Professor Gleison Santos
- [Silva *et al.*, 2011] Silva, T., Magela, R., Santos, G. et al. “Implantação do Nível F do MR-MPS Combinando Características do Processo Unificado com Práticas SCRUM”, WAMPS, 2011, Campinas.

Obrigada!