

OpenUP

Centro de Informática - Universidade Federal de Pernambuco
Sistemas de Informação

Kiev Gama

kiev@cin.ufpe.br

Slides originais elaborados por Márcio Lopes Cornélio

O autor permite o uso e a modificação dos *slides* para fins didáticos



UNIVERSIDADE FEDERAL DE PERNAMBUCO

Definição

- Processo Unificado (UP) leve que aplica abordagens iterativa e incremental em um ciclo de vida estruturado
- Adota filosofia ágil e possui foco na natureza colaborativa do desenvolvimento de software
- Processo de desenvolvimento de software que é mínimo, completo e extensível

Características

- Mínimo
 - Utiliza apenas conteúdo fundamental
- Completo
 - Possui as disciplinas essenciais para o ciclo de vida de desenvolvimento de software
- Extensível
 - Pode ser adaptado para atender projetos específicos
 - Ex: adicionar ou renomear papéis

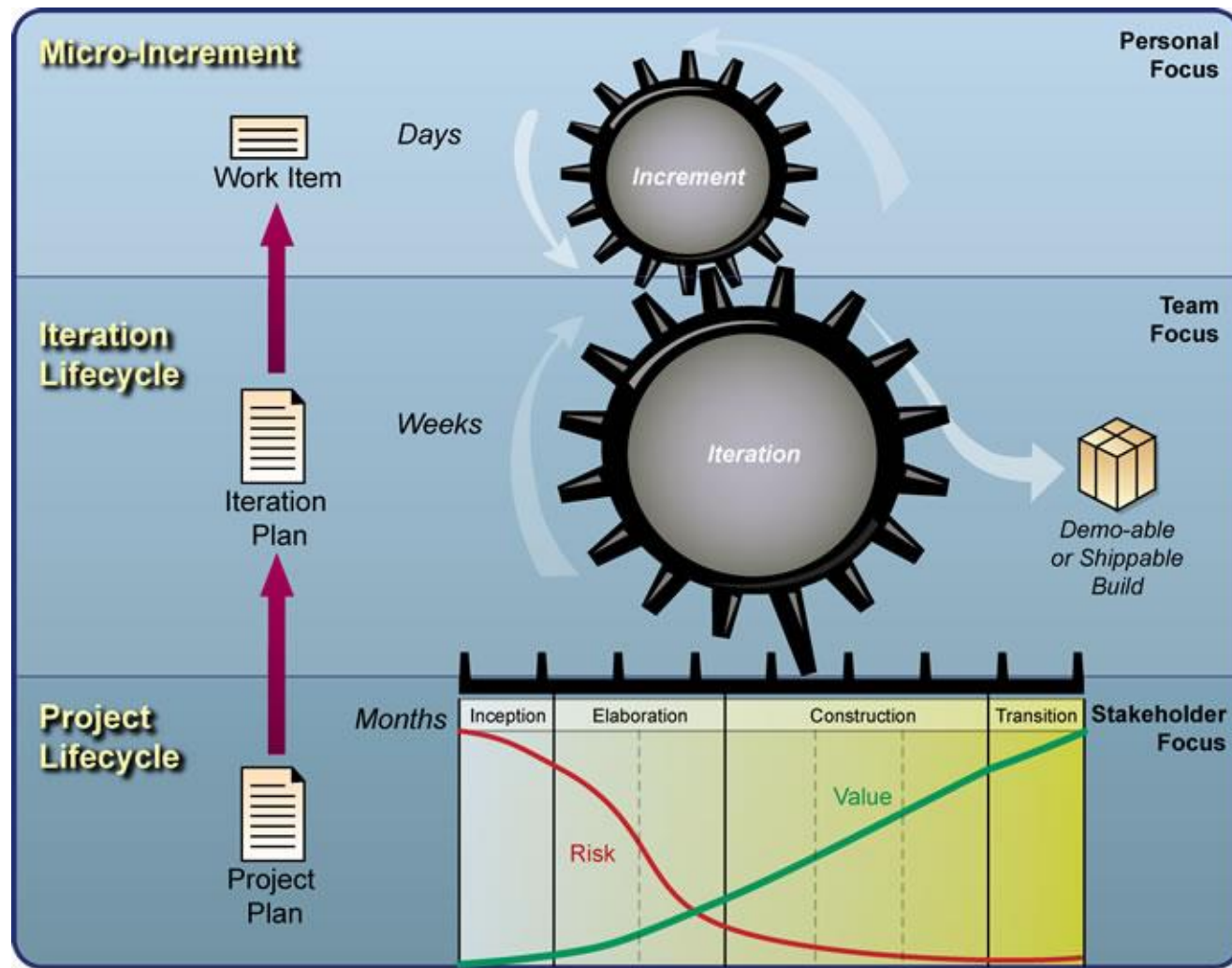
Princípios

- Colaboração para alinhar interesses e compartilhar entendimento
- Equilibrar prioridades concorrentes para maximizar valor para *stakeholder*
- Foco na arquitetura (cedo) para minimizar riscos e organizar o desenvolvimento
- Evoluir para continuamente obter *feedback* e melhoria

Elementos Básicos

- Produto de trabalho (o que é produzido)
 - Ex: documentos (visão, plano de projeto); um modelo (caso de uso, projeto)
- Tarefa (como executar o trabalho)
 - Ex: desenvolver visão do sistema, planejar iteração
- Papel (quem faz o trabalho, comportamento e responsabilidade de um indivíduo ou conjunto de indivíduos)
 - Ex: analista (obtem informações dos stakeholders e define requisitos); desenvolvedor (desenvolve partes do sistema – projeto, implementação, teste de unidade)
- Processo
 - Une tarefas, produtos e papéis, adicionando estrutura e sequenciamento

Estrutura



Micro-incrementos

- Esforço pessoal é organizado em micro-incrementos
- Unidades curtas de trabalho para alcançar os objetivos de uma iteração
 - Passo mensurável de progresso do projeto (horas ou poucos dias)
- Provêem feedback que direciona decisões em cada iteração
- Produz código testado, bem como artefatos validados
- Exemplos
 - Identificar stakeholders
 - Definir, projetar, implementar e testar um subfluxo de um caso de uso
 - Planejar iteração

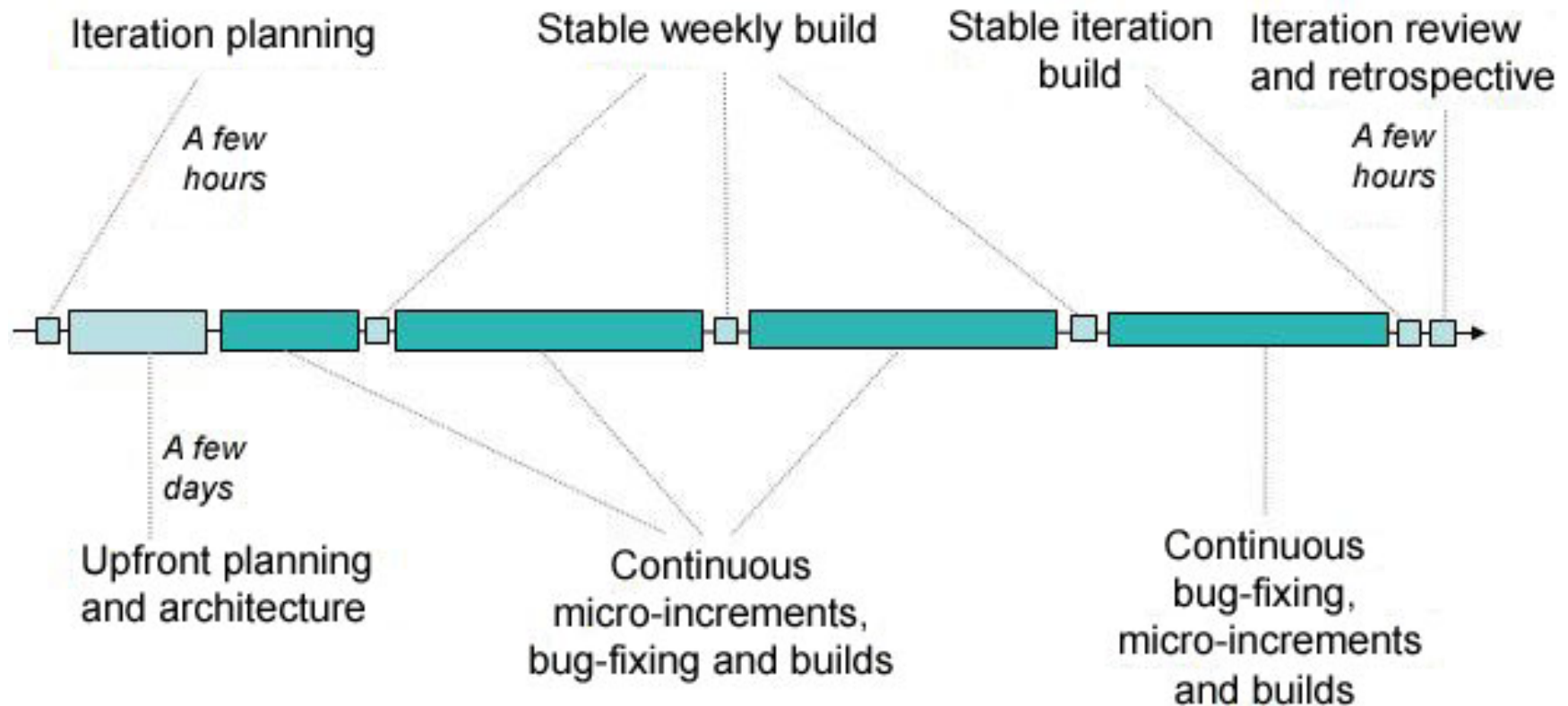
Lista de itens de trabalho

- Uma lista com todo trabalho agendado para o projeto. Cada item de trabalho pode conter referências para informação relevante para execução do mesmo
- Ponto focal para a equipe
 - Lista com todas as requisições de melhorias adicionais
 - Lista de todo o trabalho a ser priorizado, estimado e associado dentro do projeto
 - Documento para a equipe entender que micro-incrementos precisam ser entregues, obter referências de material para realização do trabalho e relatar progresso
- Exemplos
 - Casos de uso (e referências para as especificações)
 - Requisitos do sistema
 - Requisições de mudança e melhoria
 - Defeitos

Iterações

- Intervalos de tempo definidos e planejados (tipicamente semanas)
- Foco na entrega de valor incremental aos stakeholders de maneira previsível
- Plano de iteração define o que deve ser entregue na iteração e o resultado é uma versão estável (testada) e executável
- Estrutura como micro-incrementos são aplicados para liberar construções estáveis e coesas do sistema que incrementalmente progride em relação aos objetivos da iteração
- Planejamento de iteração, estimativa e progresso são centrados em itens de trabalho. Plano de iteração é criado com seleção dos itens de trabalho de maior prioridade

Iterações



Plano de Iteração

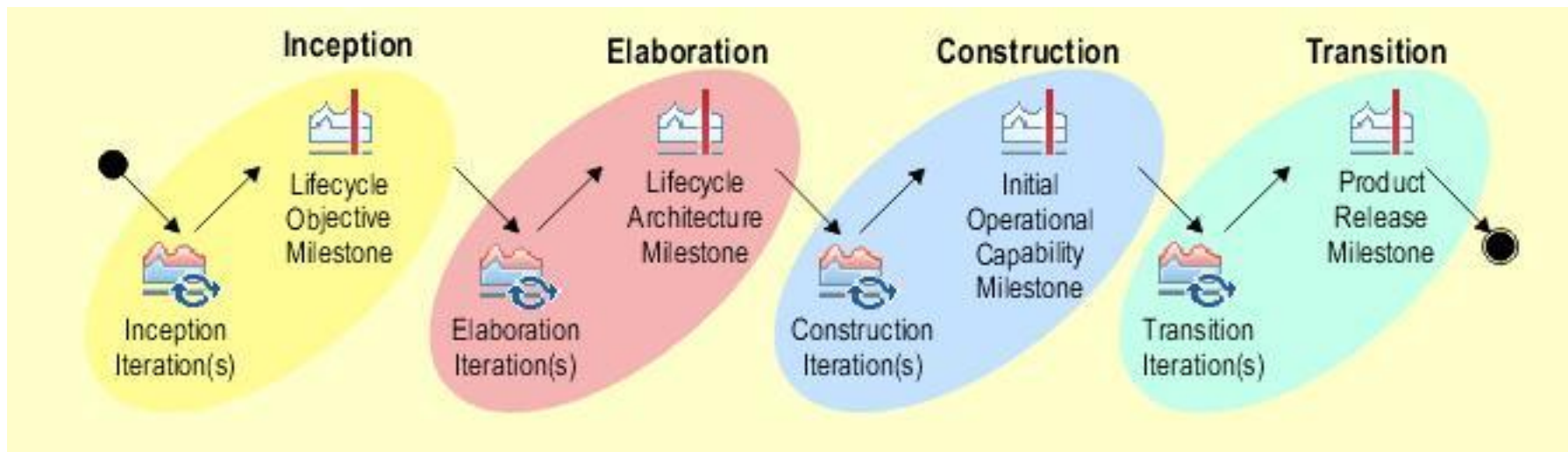
- Objetivo
 - fornecer à equipe um lugar central para informações a respeito dos objetivos da iteração,
 - plano detalhado com as atribuições das tarefas
 - dos resultados das avaliações.
- Também ajuda a equipe a monitorar o progresso da iteração e mantém os resultados da avaliação da iteração, que podem ser úteis para melhorar a próxima iteração.

Observações

- Uma iteração é focada no desenvolvimento de um *Build* (executável) que será entregue ao cliente e dura no máximo **algumas semanas**. O executável entregue deverá ter sido **exaustivamente testado** e, quando possível, integrado a outros executáveis já entregues.
- É importante destacar que o principal papel da aplicação de testes é **garantir que o artefato atende aos requisitos especificados pelos Stakeholders**.

Ciclo de Vida do Projeto

- Estruturado em quatro fases
 - Concepção
 - Elaboração
 - Construção
 - Transição



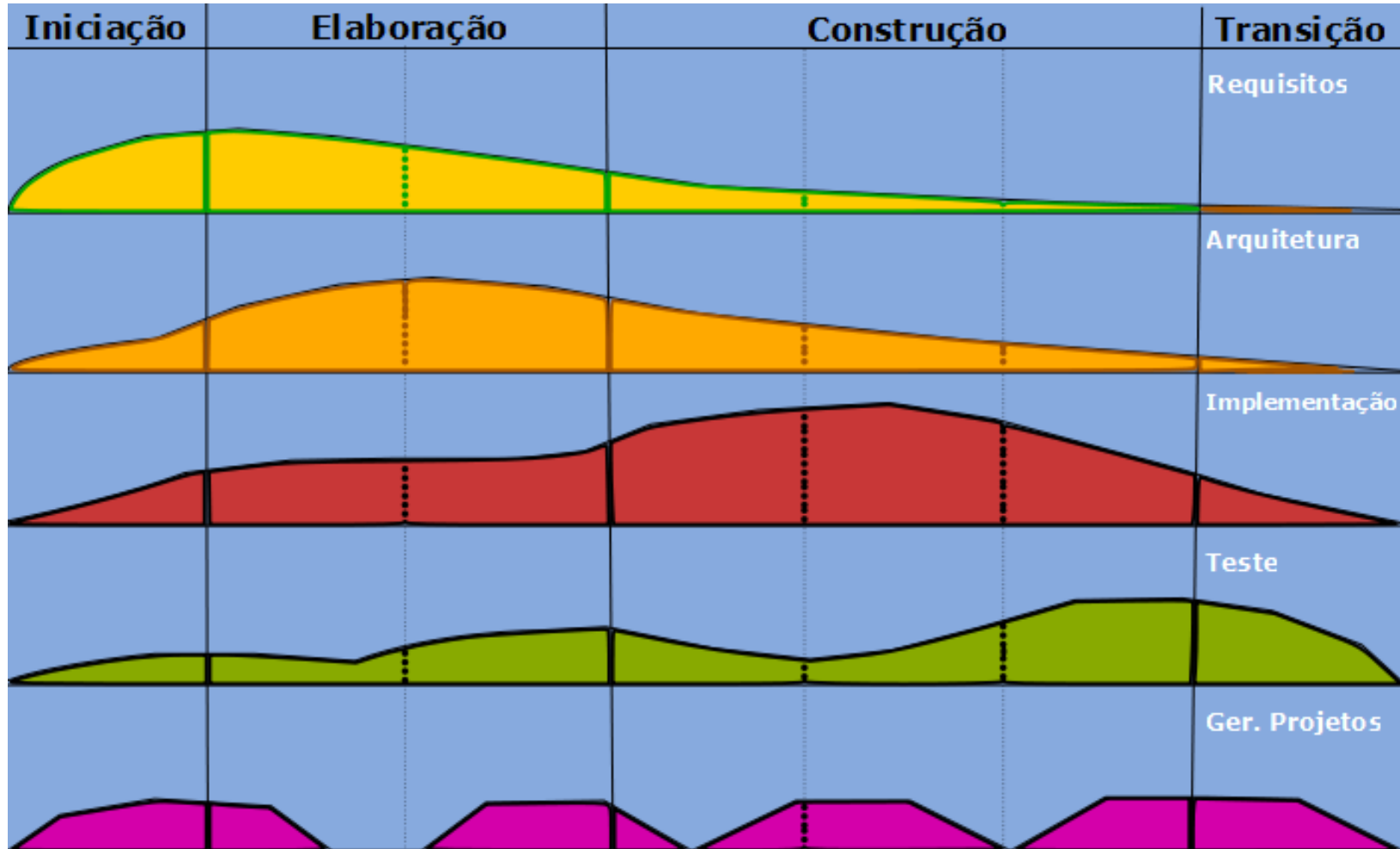
Ciclos

- Iniciação
 - Fase em que se enfatiza o processo de análise de negócios e análise de requisitos do negócio analisado, dando uma ênfase menor a arquitetura e implementação;
- Elaboração
 - Fase em que se enfatiza o processo de desenvolvimento da análise arquitetural da solução proposta;

Ciclos

- Construção
 - Fase em que se enfatiza o processo de implementação da solução proposta, bem como, testes e integração;
- Transição
 - Fase em que se enfatiza o processo de implantação do release, com importante foco na realização do teste beta e reconfiguração necessária do sistema, além de foco no processo de treinamento do usuário e conversão dos dados legados.

Distribuição das Disciplinas pelas Fases



Marco

- É importante enfatizar que cada fase é encerrada por um **marco**, ou seja, um **conjunto de atividades e artefatos** gerados pela equipe de desenvolvimento que caracterizam o encerramento da fase.

Marcos do Open Up

Iniciação	Elaboração	Construção	Transição
Objetivos do Ciclo de Vida	Arquitetura do Ciclo de Vida	Recurso Operacional Inicial	Liberação do Produto
<ul style="list-style-type: none"> -Escopo do sistema -Requisitos do sistema -Custo geral do sistema -Riscos em potencial 	<ul style="list-style-type: none"> -Baseline da Arquitetura -Riscos em potencial -Componentes do Sistema -Reusabilidade 	<ul style="list-style-type: none"> -Qualidade do sistema -Versões Alfa e Beta -Release do Sistema 	<ul style="list-style-type: none"> -Teste Beta -Conversão do BD -Treinamentos -Distribuição
<ul style="list-style-type: none"> -Documento de Visão -Lista de Riscos -Plano de Iteração -Glossário -Modelo de Caso de Uso -Protótipos 	<ul style="list-style-type: none"> -Protótipo -Modelo de Design -Modelo de Dados -Modelo de Implantação 	<ul style="list-style-type: none"> -Release do Sistema -Casos de Testes -Material de Suporte 	<ul style="list-style-type: none"> -Release -Material de Suporte -Casos de Testes -Pacote de Distribuição
FASES	MARCOS	OBJETIVOS	ARTEFATOS

Plano de Projeto

- Reúne informação necessária para gerenciar o projeto num nível estratégico. Identifica iterações e seu objetivos
 - Descreve como o projeto está organizado
 - Identifica práticas a serem seguidas
 - Define os parâmetros de rastreamento do projeto
 - Especifica os objetivos das iterações (alto nível) e seus marcos

Links e referências

- Open UP – Site oficial
 - <http://epf.eclipse.org/wikis/openup/>
- OPEN UP - PROCESSO UNIFICADO ABERTO
 - <http://www.open2up.blogspot.com/>
- OpenUP: Um processo ágil [IBM]
 - <http://ibm.co/AfO47S>
- Arkhi: Metodologias Ágeis
 - <http://blog.arkhi.com.br/tag/metodologias-ageis/>
- Blog do Fusco
 - <http://elvisfusco.com.br/tag/metodologia-agil/>
- Estendendo o OpenUP para Atender as Áreas de Processo Relacionadas a Garantia da Qualidade e Medição do CMMI-DEV Nível 2
 - <http://bit.ly/yxUIsE>
- OpenUP manual @ Scribd
 - <http://www.scribd.com/doc/51358285/OpenUP-manual>
- Utilizando OpenUP/Basic para Desenvolvimento de Aplicações WEB
 - <http://www.scribd.com/doc/35414358/60/Estendendo-o-OpenUP-Basic>