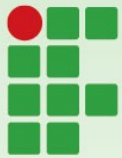


INSTITUTO FEDERAL
Santa Catarina



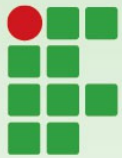
Introdução ao Processo Unificado

Prof. Edjandir Corrêa Costa
edjandir.costa@ifsc.edu.br



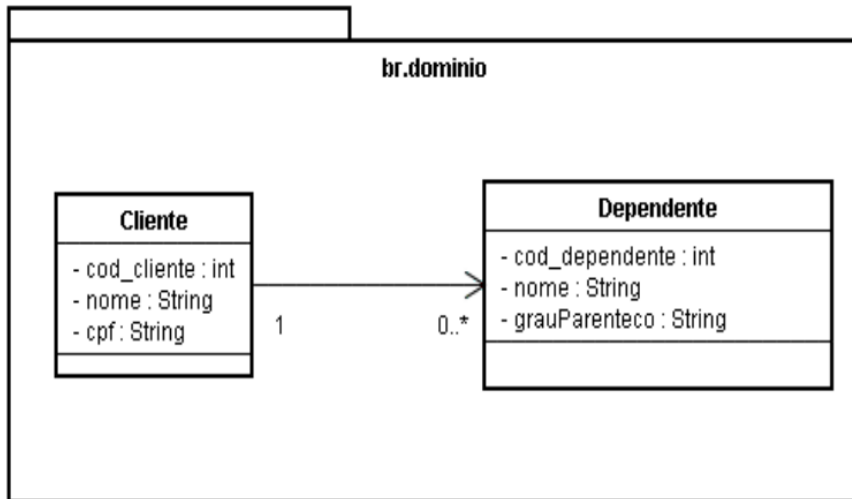
O Processo Unificado

- O processo unificado (Unified Process UP, ou em português, PU) é um processo de desenvolvimento de software;
- Existem outros processos além do PU?
 - IBM/Rational (Baseado no PU)
- RUP Rational Unified Process
 - Scott W. Ambler (Baseados no PU)
- AUP Agile Unified Process
- EUP Enterprise Unified Process



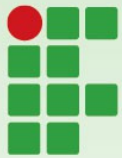
Relação do PU com a UML

- O PU usa a UML como linguagem de modelagem;
- Porque usar linguagens gráficas?



```
package br.dominio;
public class Cliente {
    private int cod_cliente;
    private String nome;
    private String cpf;
    private Dependente[] dependente;
}
```

```
package br.dominio;
public class Dependente {
    private int cod_dependente;
    private String nome;
    private String grauParenteco;
}
```

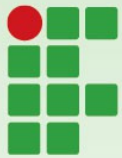


INSTITUTO FEDERAL
Santa Catarina



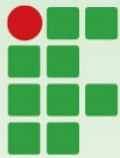
Por que usar UML?

- A UML é usada para facilitar o entendimento de aspectos complexos inerentes ao sistema Computacionais;
- A UML é uma família de notações gráficas que ajuda na descrição e no projeto de sistemas de software.



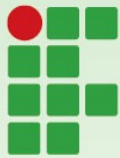
Características do PU

- Dirigido por casos de uso:
 - Ter os casos de uso como entrada (fonte) para a maioria das atividades do processo;
- Centrado na arquitetura:
 - Motivado a desenvolver o produto com base em uma arquitetura de software;
- Iterativo e incremental:



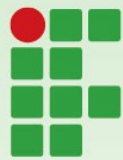
O que são Casos de Uso

- O que são casos de uso?
 - Sequências de ações que são executadas por um ou mais atores e pelo próprio sistemas;
 - Produz um ou mais resultados para um ou mais valores.
- Como o PU usa os casos de uso?
 - Os casos de uso conduzem o desenvolvimento do projeto;
 - Os casos de uso são usados desde a coleta de requisitos até os testes de aceitação.

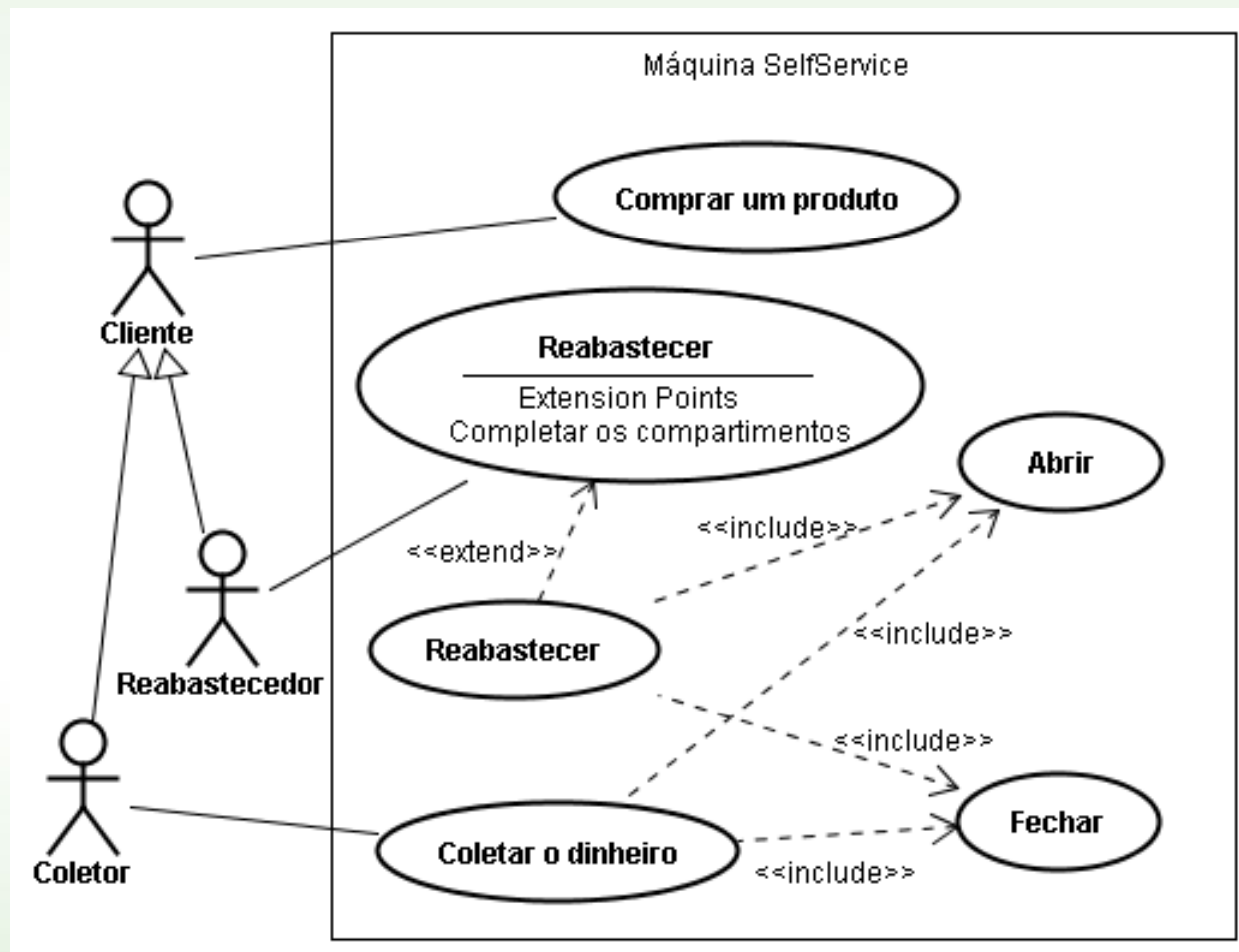


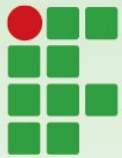
O que são Casos de Uso

- Por que casos de uso?
 - São expressos sob a perspectiva dos usuários do sistema;
 - São expressos em língua natural, intuitivamente óbvios para o leitor;
 - Oferecem uma habilidade consideravelmente maior para a compreensão dos reais requisitos do que documentos típicos;



O que são Casos de Uso



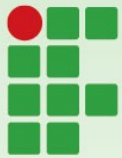


INSTITUTO FEDERAL
Santa Catarina



PU: Entendendo a Arquitetura

- O que é uma arquitetura?
 - Organização fundamental do sistema como um todo;
 - Entre aspectos importantes de uma arquitetura estão incluídos elementos estáticos e dinâmicos;
 - Arquitetura também descreve questões de desempenho, escalabilidade, reuso, restrições tecnológicas (os requisitos não funcionais);

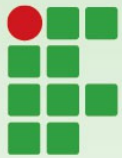


INSTITUTO FEDERAL
Santa Catarina



PU: Entendendo a Arquitetura

- A arquitetura do sistema em construção é o alicerce fundamental sobre o qual construirá o produto;
- A arquitetura deve ser vista e compreendida por todos da equipe;
- Arquitetura Arquitetura é o mecanismo mecanismo para chegar a um produto produto robusto, flexível, expansível e de custo viável.



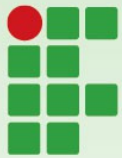
INSTITUTO FEDERAL
Santa Catarina



PU: Entendendo a Arquitetura

Por que centrar o desenvolvimento na arquitetura?

- Entender a visão global, simplificando o entendimento de sistemas complexos;
- Organizar o esforço de desenvolvimento, dividindo o software em porções discretas;
- Facilitar as possibilidades de reuso, facilitando o reuso de componentes dentro das porções discretas;



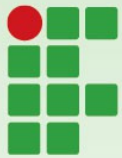
INSTITUTO FEDERAL
Santa Catarina



PU: Entendendo a Arquitetura

Por que centrar o desenvolvimento na arquitetura?

- Facilitar a evolução do sistema, permitindo mudanças mais fáceis dentro das porções principalmente quando não muda as responsabilidades;
- Dirigir os casos de uso, fornecendo condições de sempre adicionar mais casos de uso.



INSTITUTO FEDERAL
Santa Catarina



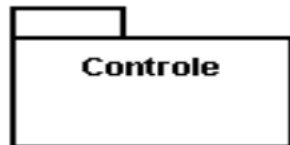
Arquitetura: Um exemplo

Padrão Arquitetural: Camadas

- Organizar a estrutura lógica de larga escala de um sistema em camadas distintas de responsabilidades



Janelas
Relatórios
XHTML, Javascript, entre outros



Trata as requisições da camada de visão

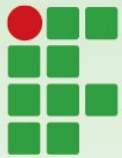


Trata as requisições da camada de controle
Implementa as regras de domínio



Trata as requisições da camada de domínio
Persiste os objetos da camada de domínio (em um BD, por exemplo)

Observe que, neste padrão de arquitetura, cada camada é representada por um pacote, em linguagem UML.

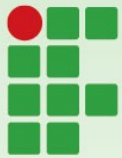


INSTITUTO FEDERAL
Santa Catarina



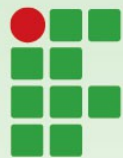
PU: Iterativo Incremental

- Iteração é um mini-projeto que resulta em uma versão do sistema liberada internamente ou externamente;
- É pressuposto que a cada mini -projeto posterior incrementa em funcionalidade o mini-projeto anterior.



Fases do PU

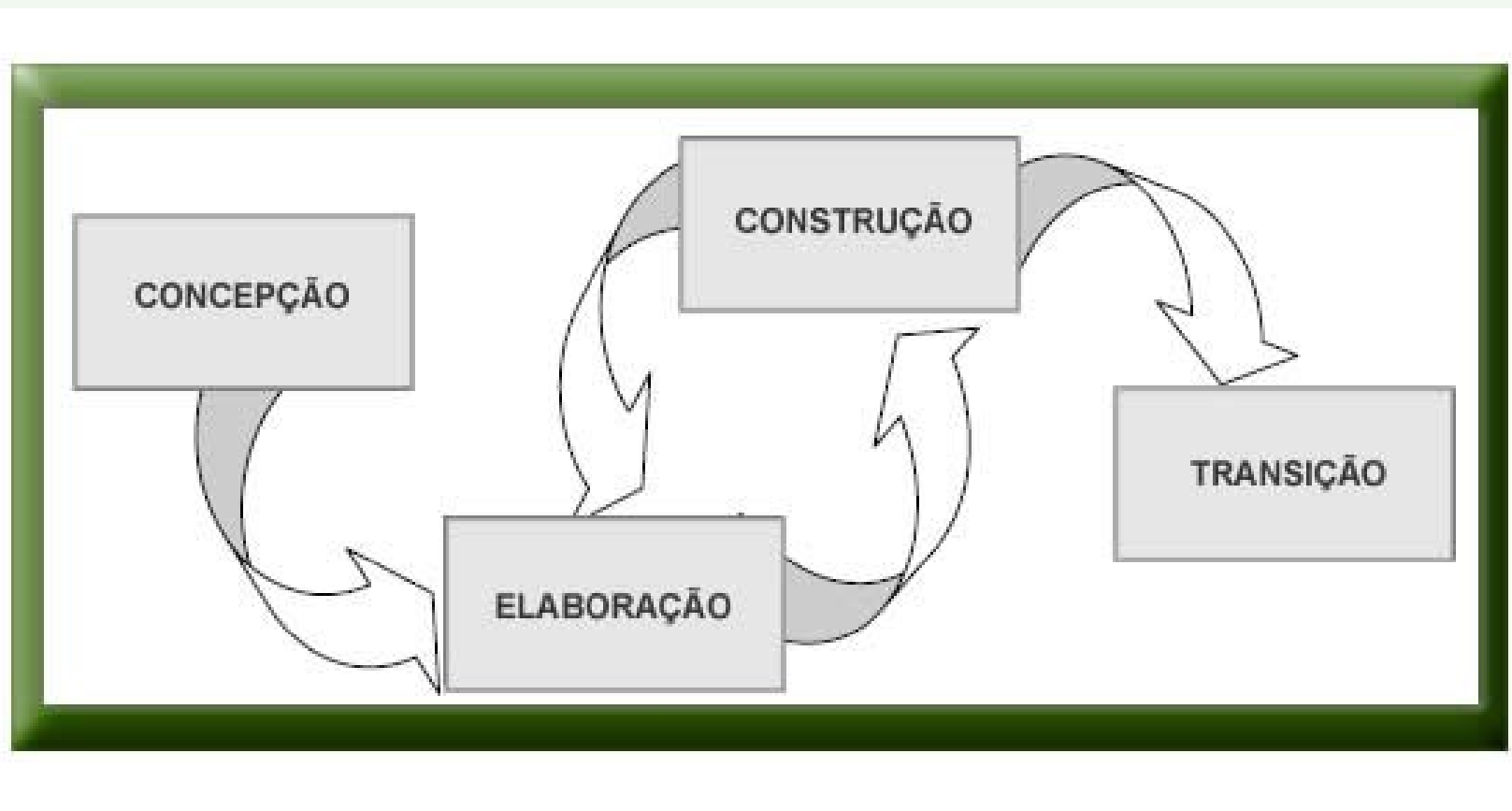
- Concepção ou iniciação (Inception) – Visão aproximada, casos de negócio, escopo e estimativas vagas;
- Elaboração (Elaboration) – Visão refinada, implementação iterativa da arquitetura central, resolução dos altos riscos, identificação da maioria dos requisitos e estimativas mais realistas;
- Construção (Construction) – Implementação iterativa dos elementos restantes de menor

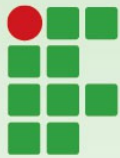


INSTITUTO FEDERAL
Santa Catarina



Fases do PU





INSTITUTO FEDERAL
Santa Catarina

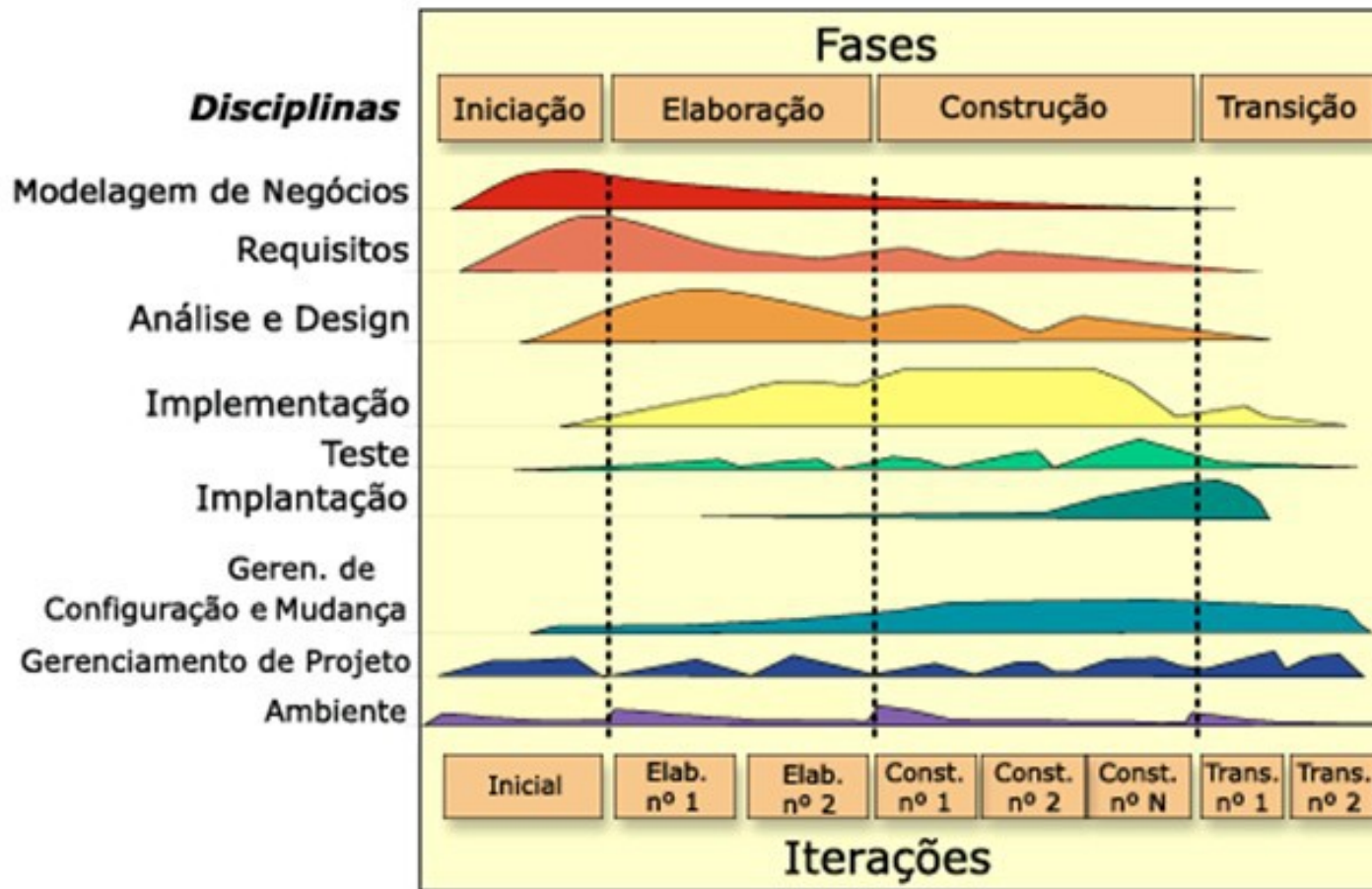


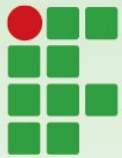
Fluxos de trabalho

- O que é fluxo de trabalho?
 - Cada fluxo de trabalho indica um conjunto de atividades e papéis que as executam;
 - Os fluxos permeiam as 4 fases do PU.
 - Fluxos de trabalho são sinônimos de disciplinas (RUP).



Fluxos de trabalho



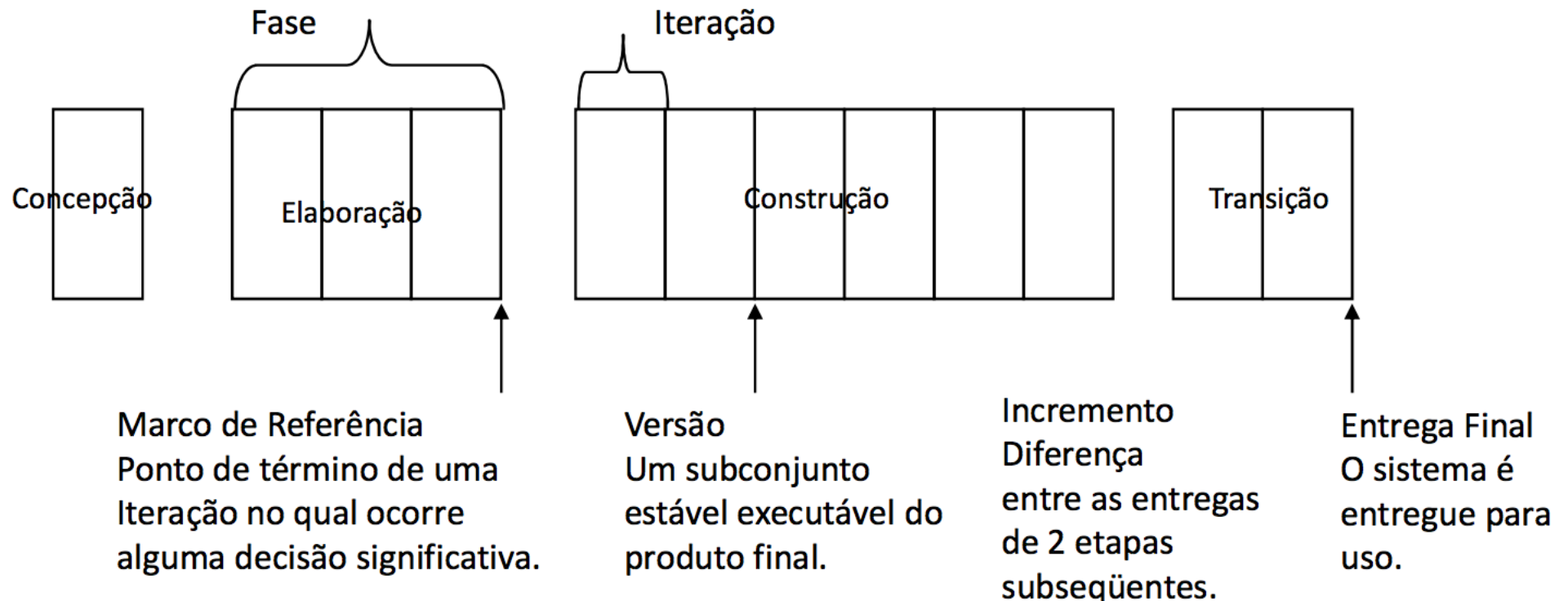


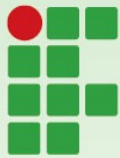
INSTITUTO FEDERAL
Santa Catarina



Fases x Iterações

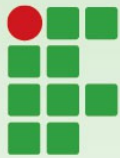
As fases contém iterações sequenciais. O marco principal contempla a entrega de uma versão do software.





Iterações e Fluxo de trabalho

- Qual a relação entre iterações e fluxos de trabalho?
 - As iterações podem conter todos os fluxos de trabalho;
 - As iterações do início ficam nos fluxos requisitos e análise;
 - Estas iterações podem chegar ao fluxo de projeto, mas dificilmente implementação e testes;
 - As iterações do final executam os fluxos de implementação e testes.



INSTITUTO FEDERAL
Santa Catarina



Atividades de cada iteração

- Planejar a iteração;
- Executar as disciplinas (fluxos de trabalho);
- Fazer a análise ao final da iteração;
- Revisar o plano de projeto;
- Ir para a próxima iteração, se existir.