

# Arquitetura de Software

- Fundamentos e Tendências -

Prof. Marco Fagundes  
mfagunde@tre-pa.gov.br

## Agenda

- Objetivos e Contexto
- Arquitetura de Software (A.S.)
- A.S. no Processo de Desenvolvimento
- O Modelo ArcADe
- Passado, Presente e Futuro

## Objetivos

- **Objetivos Geral:**
  - Enfatizar a importância da arquitetura para o sucesso de um projeto de software
- **Objetivos específicos:**
  - Apresentar os conceitos básicos da arquitetura de software
  - Destacar o papel da arquitetura de software no processo de desenvolvimento
  - Apresentar o modelo ArcADe
  - Mostrar as tendências na área da arquitetura de software

## Contexto

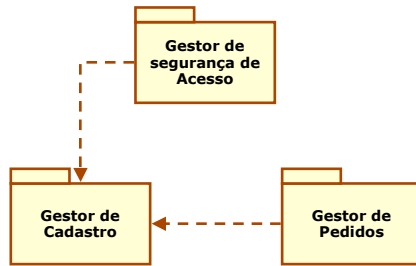
O aumento do **tamanho** e da **complexidade** dos sistemas de *software*

Redução de tempo e custo de desenvolvimento e manutenção do *software*

Adotar uma **arquitetura** correta pode ajudar na gerência complexidade e trazer diversos benefícios

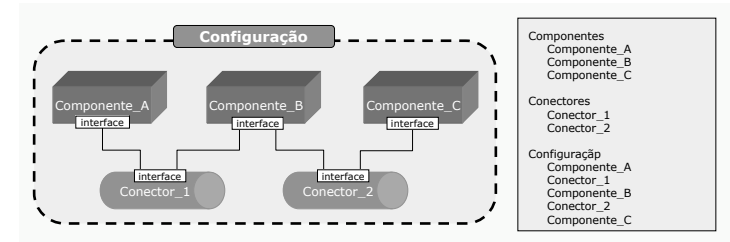
## Arquitetura de Software

O diagrama abaixo representa uma arquitetura?



## Arquitetura de Software

A arquitetura do software define a **estrutura** do software, que compreende os **componentes** com suas propriedades visíveis externamente e os **relacionamentos** entre eles.



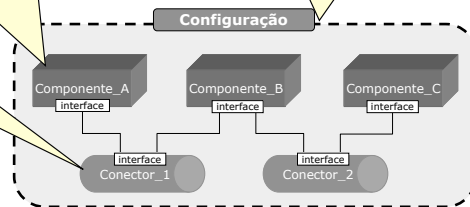
## Arquitetura de Software

- Existem várias definições sobre arquitetura do software
- Semelhanças no núcleo através de três abstrações básicas: Componentes, Conectores e Configuração

Modela a computação e o armazenamento de informações  
Por exemplo: Cliente, Servidor, Subsistema ou Sistema

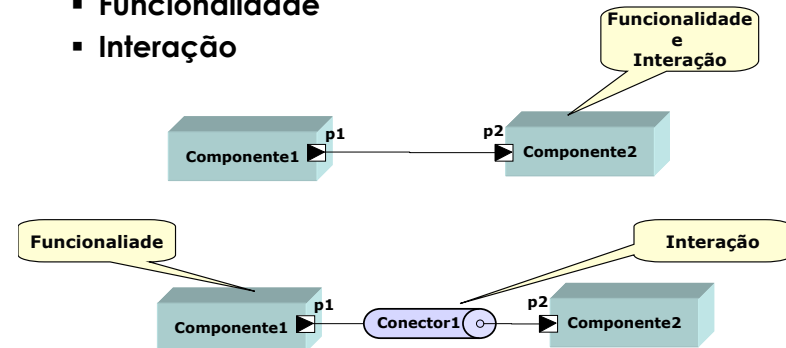
Grafo de componentes e conectores ligados, descrevendo uma estrutura arquitetural

Modela a interação entre os componentes  
Por exemplo: RPC



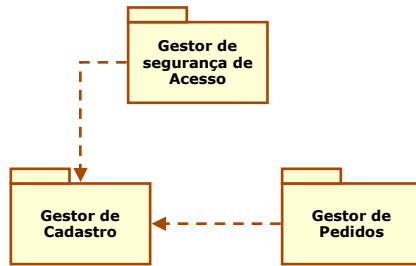
## Arquitetura de Software

- A Arquitetura de Software enfatiza a separação de interesses (**concerns**)
  - Funcionalidade
  - Interação



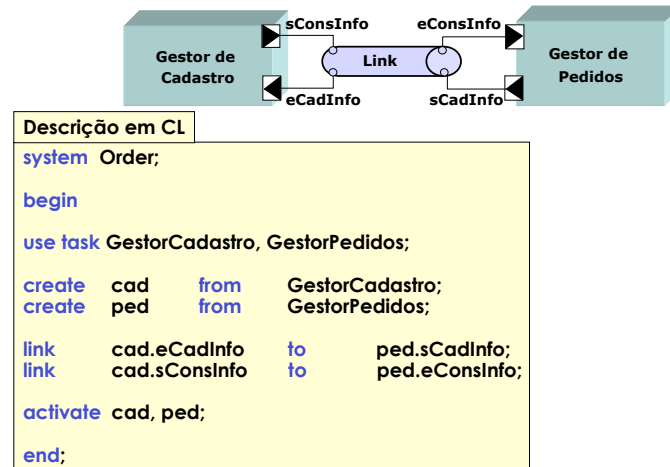
## Arquitetura de Software

O diagrama abaixo **não** representa uma arquitetura



## Arquitetura de Software

Exemplo de uma arquitetura



## Vantagens da A.S.

- Facilita a combinação de abordagens de reuso de software (ex. Estilos, COTS ou Padrões de Projeto)
- Possibilita análise da descrição da arquitetura nas fases iniciais do desenvolvimento (ex. propriedades não funcionais ou conformidade com um estilo)
- Facilita a evolução do software
- Permite uma melhor comunicação entre os *stakeholders*

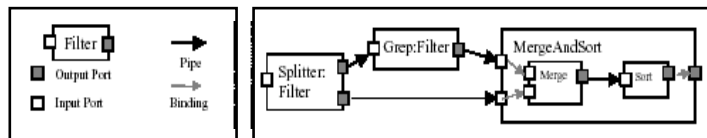
## Estilos arquiteturais

Um estilo arquitetural consiste de um **vocabulário** de elementos de projeto e um conjunto de regras de **configuração** que governam a combinação desses elementos

- Elementos Arquiteturais (de Projeto)
  - Componentes
  - Conectores
- Exemplo de regras de configuração
  - Uma camada pode somente se comunicar com a camada adjacente no estilo Camadas

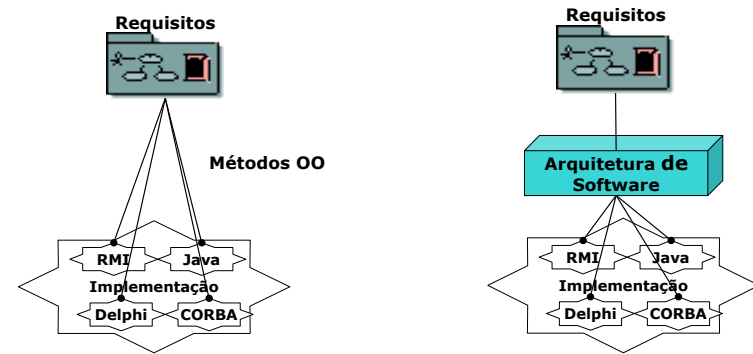
## Estilos arquiteturais

- Definem uma família e não apenas um sistema
- A descrição da arquitetura é uma instância de um estilo
- Exemplos de Estilos
  - Cliente-servidor
  - Pipe-filter
  - Camadas (*Layered*)
- Exemplo de uma A.S. no estilo *Pipe-filter*:



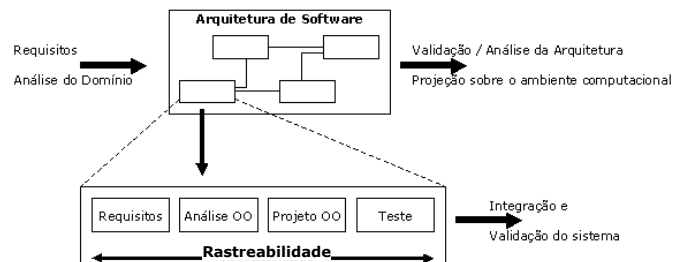
## A.S. no Processo de Desenvolvimento

- Reduz o gap semântico entre os requisitos e o código
- Fornece uma base para as outras fases de desenvolvimento do software



## A.S. no Processo de Desenvolvimento

- A.S. e a O.O. de forma complementar



## O Modelo ArcADe

### Contexto

A Arquitetura de Software desempenha um papel importante para o sucesso do projeto

### Problema

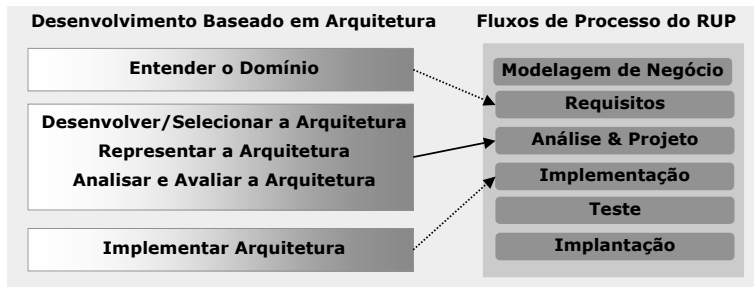
Ausência da integração natural entre a Arquitetura de Software e processos de software comumente utilizados

### O Modelo ArcADe

Um modelo de processo que integra a arquitetura de software com um processo de desenvolvimento largamente utilizado (RUP)

## Desenv. Baseado em Arquitetura X Fluxos do RUP

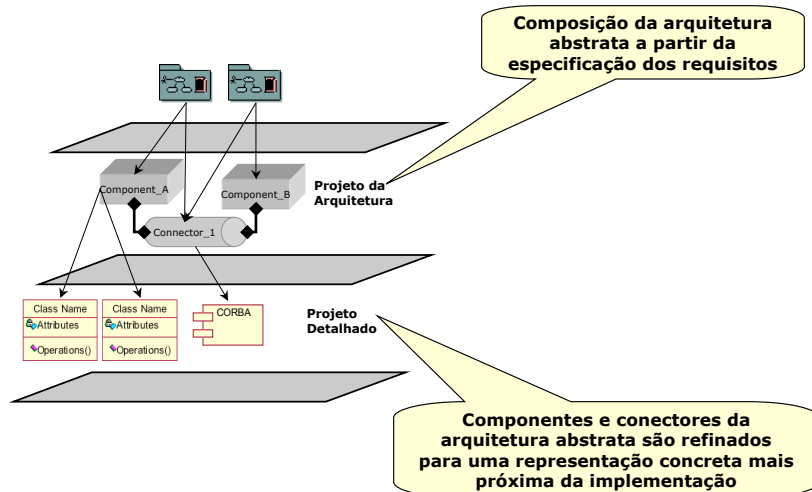
- Foco na arquitetura do software
- Arquitetura serve como base para as atividades de análise, projeto e implementação do software
- Representação da arquitetura (ADLs)



## Sobre o ArcADe

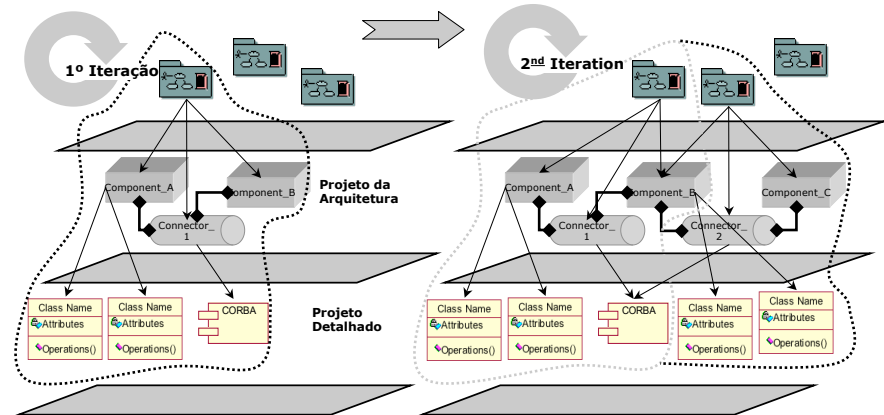
- O Modelo de Processo ArcADe (*Architecture-based Analysis and Design*) integra a A.S. com elementos (conceitos, métodos e técnicas) do RUP
- O ArcADe adota a AS para definir as etapas do desenvolvimento
- O modelo utiliza os elementos do RUP para organizar o processo em um fluxo de trabalho

## Níveis de Abstração do ArcADe

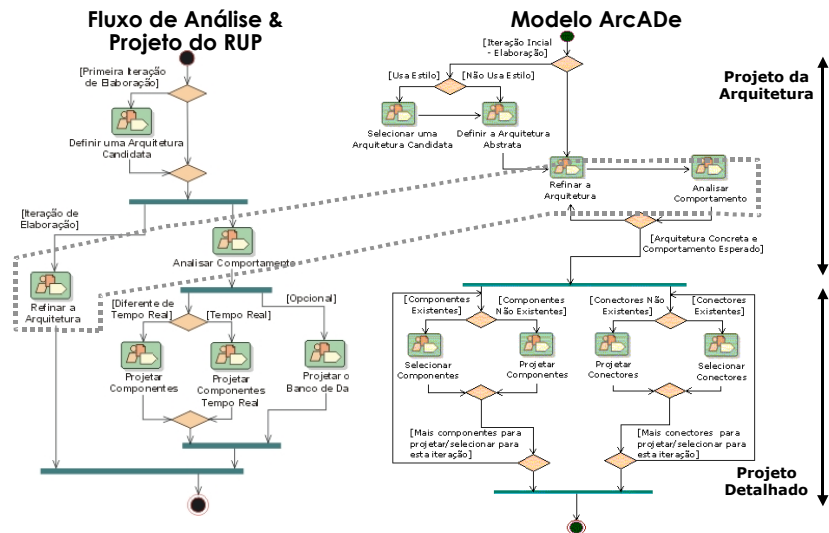


## O Modelo ArcADe

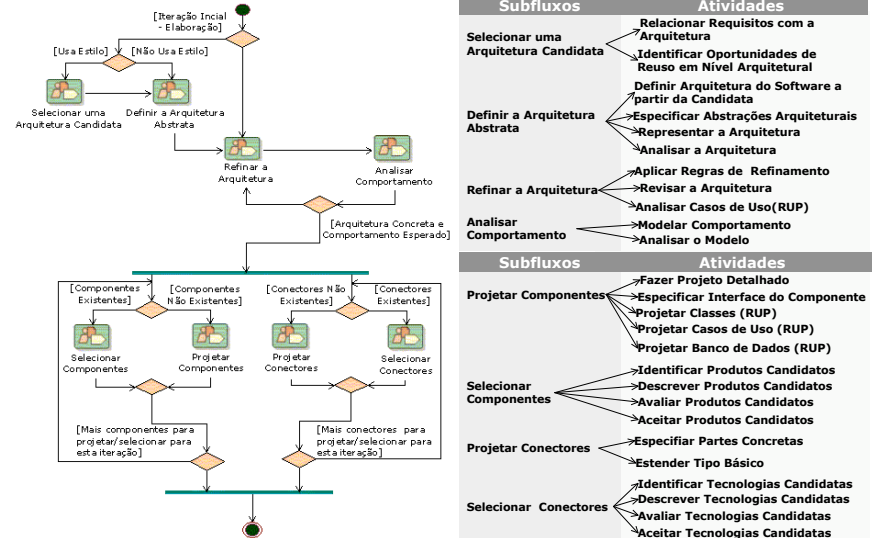
- Desenvolvimento iterativo e incremental no ArcADe



## ArcADe vs RUP



## O Modelo ArcADe

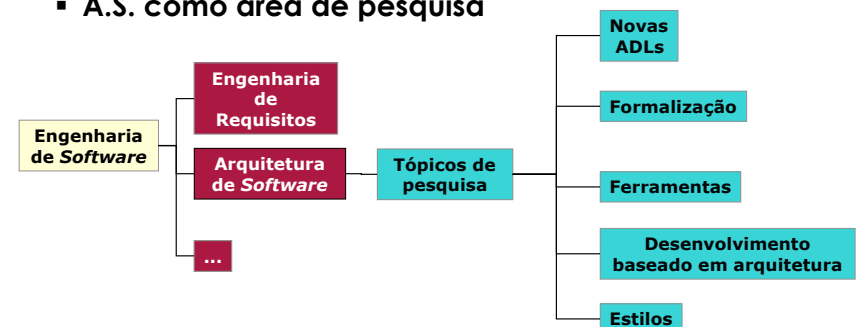


## A.S. Passado, Presente e Futuro

- No passado:
  - forma *ad hoc* - descrições informais diagramas de caixas linhas
  - Profissionais começaram a reconhecer o papel crítico da A.S. para o sucesso do projeto
  - Os projetistas de linguagens começaram a considerar uma notação específica e formal para representar a arquitetura

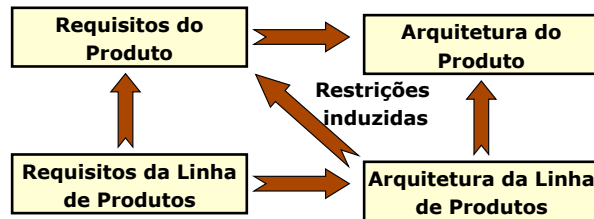
## A.S. Passado, Presente e Futuro

- No presente:
  - Arquitetura é tratada como uma atividade importante e explícita no desenvolvimento de software
  - Diversos congressos científicos e livros dedicados à A.S
  - A.S. como área de pesquisa



## A.S. Passado, Presente e Futuro

- Três avanços importantes:
  - Linguagens de Descrição de Arquiteturas e Ferramentas
  - Engenharia de Linha de Produtos
  - Codificação e Disseminação



## A.S. Passado, Presente e Futuro

- No futuro:
  - Evolução natural:
    - Difusão e Assimilação da A.S. como disciplina
    - Ferramentas para o projeto arquitetural
  - Tendências:
    - Adoção de linguagens como a UML e XML para modelagem arquitetural
    - Computação Ubíqua (“pervasiva”)

## Pontos Principais

- A A.S. tem recebido crescente reconhecimento e atenção
- No processo de desenvolvimento, a A.S. desempenha um papel importante como ponte entre requisitos e implementação
- Diversas contribuições para facilitar e difundir a A.S.
- O Modelo ArcADe integra a arquitetura de software com elementos (conceitos, métodos e técnicas) do RUP