

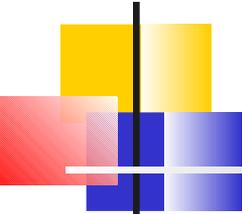
# CrossMDA: Arcabouço para integração de interesses transversais no desenvolvimento orientado a modelos

<sup>1</sup>Marcelo Pitanga Alves, <sup>2</sup>Paulo F. Pires,  
<sup>2</sup>Flávia C. Delicato, <sup>1</sup>Maria Luiza M. Campos

<sup>1</sup>**Departamento de Ciência da Computação (DCC/IM) – Núcleo de Computação Eletrônica (NCE) – Universidade Federal do Rio de Janeiro (UFRJ)**

<sup>2</sup>**DIMAp: Departamento de Informática e Matemática Aplicada - Universidade Federal do Rio Grande do Norte - Natal, RN - Brasil**





# Introdução

---

- Preocupação no desenvolvimento de sistemas
  - A busca por uma melhor **separação de interesses**
  - Tipos de Interesse [W. Hürsch e C.V. Lopes 1995]
    - **Básicos**
      - Fornecem as funcionalidades que são relevantes ao **domínio** da aplicação
    - **Especiais**
      - São utilizados para **gerenciar** ou **otimizar** os interesses básicos
      - Fornecem suporte a outras formas de computação
        - desempenho, persistência, monitoramento, etc...

# Separação de Interesse -

## Foco

- Interesses Especiais

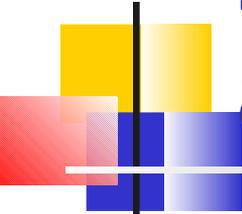
- Preocupações fáceis de identificar e difíceis de destacar
- Também chamados de interesses transversais (*crosscutting concerns*)

“Interesses requeridos em várias partes do sistema onde as suas implementações geram o espalhamento do código, resultando em um código confuso e muito difícil de se entender, manter e evoluir”.

# Separação de Interesse - abordagens

---

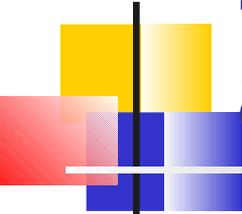
- POA (Programação Orientada a Aspectos)
  - Complementar a Programação Orientada a Objetos (POO)
  - Isolamento através de uma nova abstração
    - **Aspecto**
  - Mecanismo de composição (*weaving*)
  - Reuso do código do aspecto
- Separação horizontal de interesses
- Problema: Foco no nível de código
  - Não fornece mecanismos para **modelagem** de aspectos



# Separação de Interesse - abordagens

---

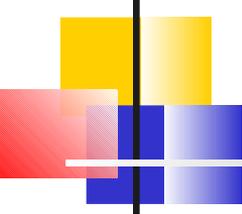
- MDA (*Model Driven Architecture*)
  - Três diferentes níveis de abstrações
    - Modelo Independente de Computação (CIM)
    - Modelo Independente de Plataforma (PIM)
    - Modelo Dependente de Plataforma (PSM)
  - Mecanismo de transformação de modelos
  - Reuso de modelos
  - Separação **vertical** de interesses
- Problema: não trata a Separação **horizontal** de interesses



# Separação de Interesse – nova abordagem

---

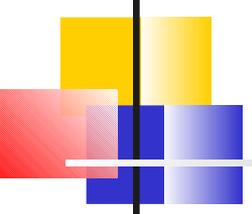
- Modelagem Orientada a Aspectos (MOA)
  - Separação horizontal de interesses
    - Concentra-se em técnicas para:
      - identificação, análise, gerenciamento e representação de interesses transversais na fase de modelagem, usando extensões UML
  - Problema
    - falta de ferramentas adequadas para:
      - Modelagem;
      - Gerência do relacionamento de elementos do negócio com um determinado interesse transversal (*weaving*)



# Motivação

---

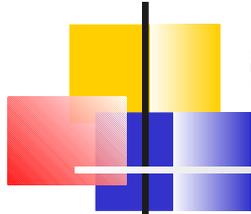
- Propor uma solução que integre as abordagens MDA e POA para tratamento de interesses transversais no **nível de modelos**



# Proposta de Solução Arcabouço CrossMDA

---

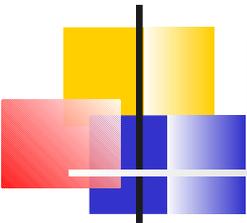
- Processo de transformação  
+
- Conjunto de serviços e ferramentas de apoio associadas
- Objetivos:
  - elevar o nível de abstração na modelagem através do uso de modelos PIM de interesses transversais independentes do modelo de negócio
  - reusar artefatos de modelos PIM de interesses transversais
  - automatizar o mapeamento do relacionamento de interesses transversais com elementos do modelo de negócio através do processo de transformação da MDA
  - facilitar o reuso de artefatos de transformação MDA
  - promover o reuso de modelos PIM de negócios.



# CrossMDA

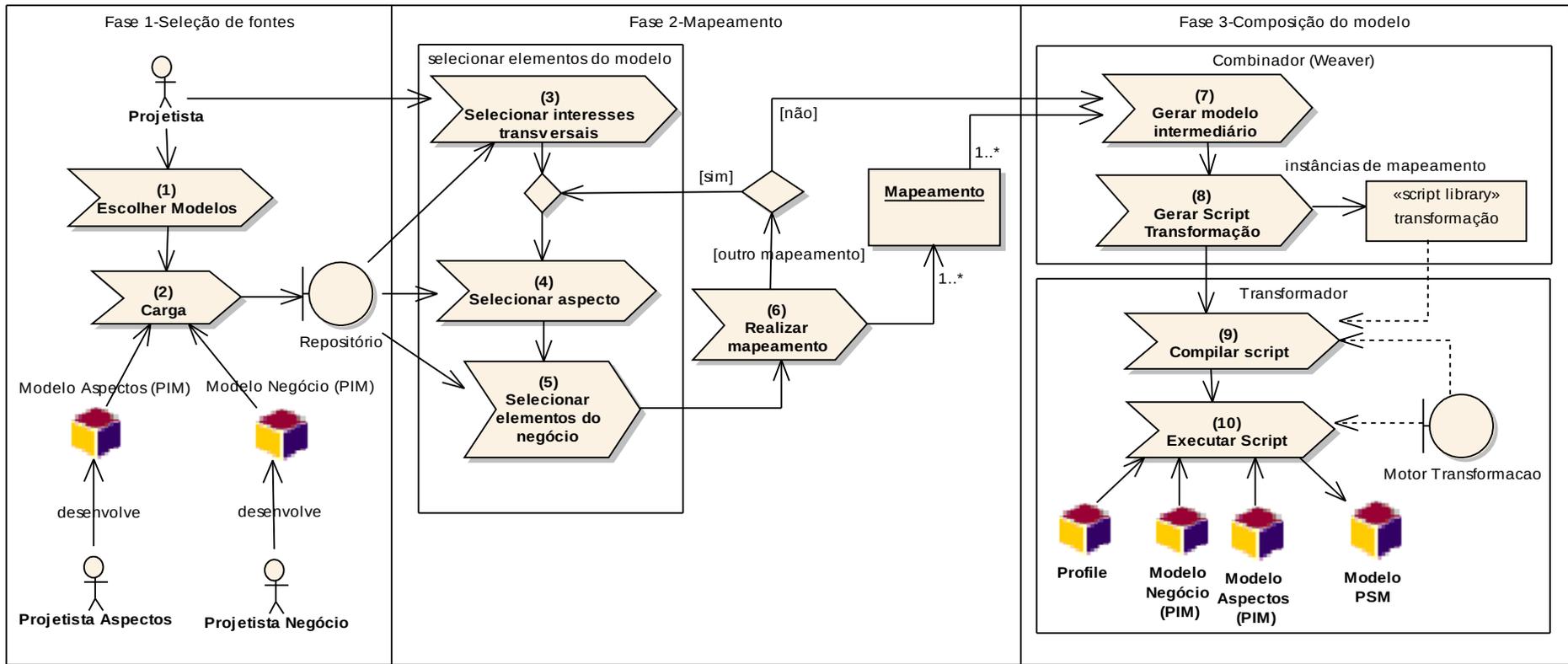
---

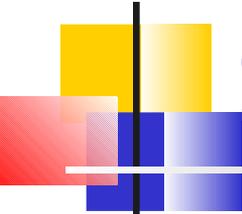
- Tratamento de aspectos no nível de modelagem
- Separação de interesses na **dimensão horizontal**
  - modelagem de aspectos independentemente dos elementos de negócio
  - modelo de aspectos
    - é uma representação abstrata de interesses transversais
    - eleva o nível da modelagem de aspectos --> PIM de aspectos
- Separação de interesses na **dimensão vertical**
  - tratada através da implementação de um processo de:
    - entrelaçamento entre modelo de aspectos e de



# Processo CrossMDA

od CrossMDA - Process

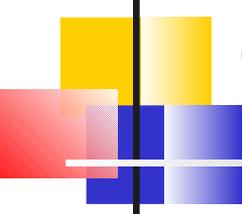




# Serviços do CrossMDA

---

- Persistência de modelos
  - NETBEANS Metadata Repository
- Perfil UML de Aspectos
  - Estereótipos
  - Valores Etiquetados (*TaggedValues*)



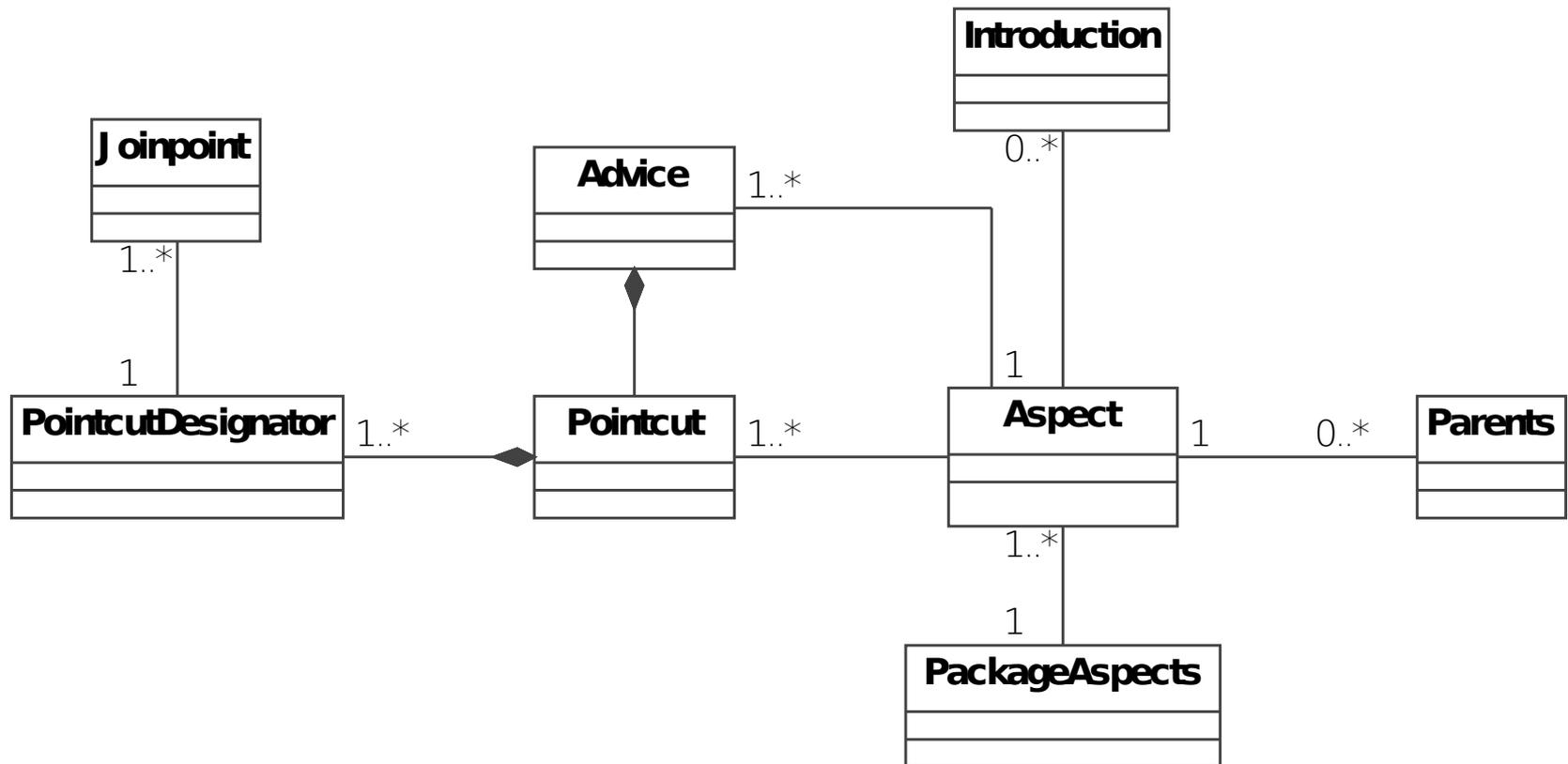
# Serviços do CrossMDA

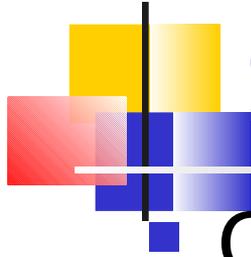
---

- Mapeamento de elementos
  - Modelo de mapeamento
    - Persistência dos relacionamentos
  - Tipos de mapeamento
    - Pontos de atuação (*pointcut*)
      - Padrão de especificação de pontos de atuação (*pointcut*) da POA e de AspectJ
    - Intertipos
      - Introduction, Declare Parents

# Serviços do CrossMDA

- Metamodelo de Mapeamento para transformação





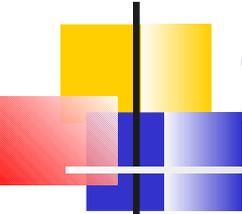
# Serviços do CrossMDA

---

## ■ Composição do modelo

### ■ Combinador

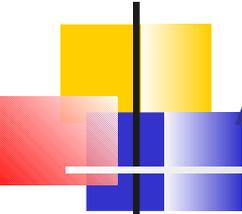
- Gera modelo intermediário
  - Representação da hierarquia de composição de uma instância de uma classe aspecto
- Geração do programa de transformação
  - Implementação da especificação formal do processo de composição de modelos
  - Uso de linguagem formal de transformação baseada na especificação MOF-QVT
    - ATL
    - ATLAS group (INRIA e LINA, Universidade de Nantes)



# Serviços do CrossMDA

---

- Transformador de modelos
  - Compilar e executar o programa de transformação para gerar o modelo PSM
  - Motor de transformação da ATL
    - Máquina virtual (*ATLvm*)
      - parser (análise sintática)
      - compilador (gerar o byte-code (*asm*) )
      - executor (carga e execução do programa de transformação)

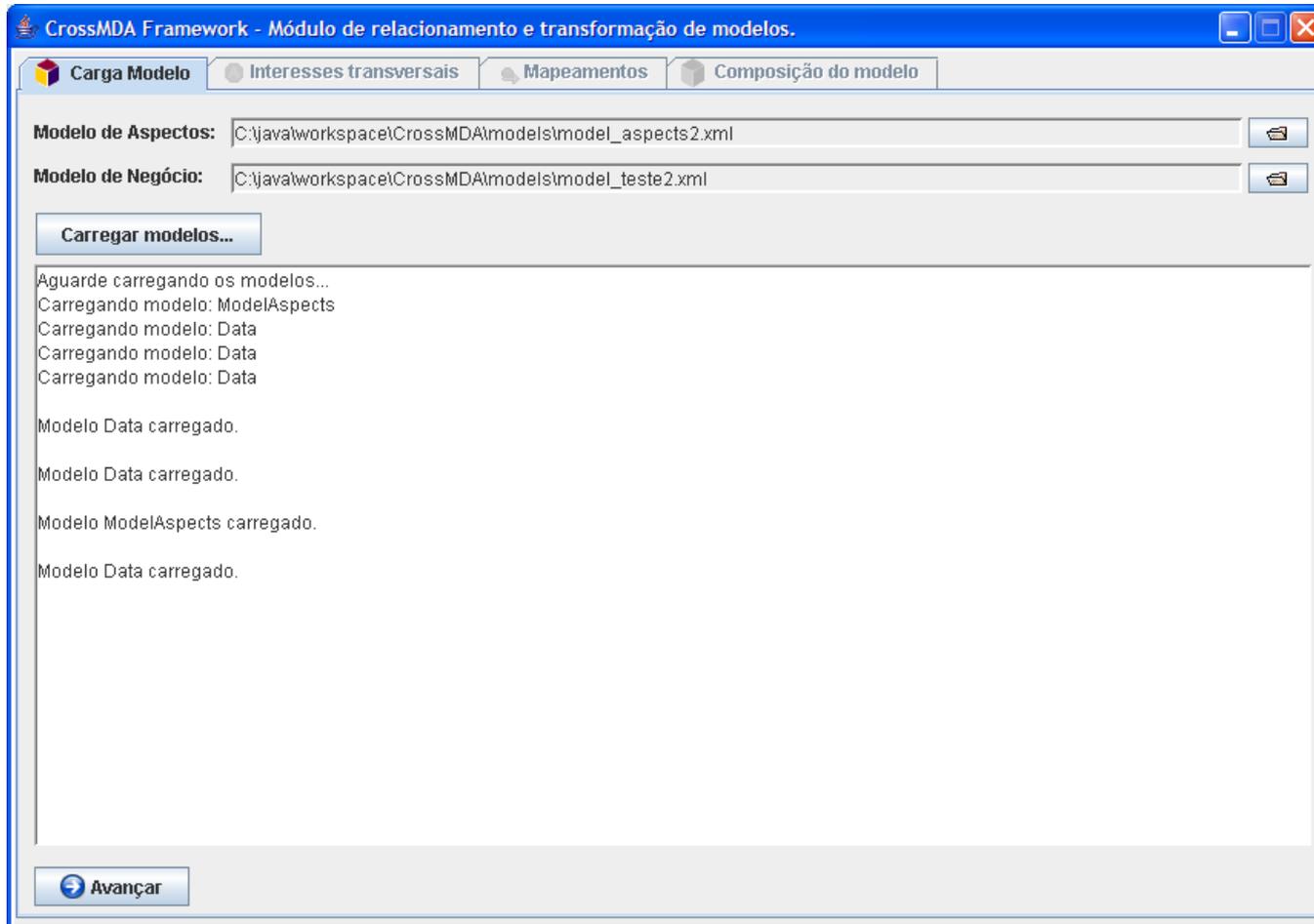


# Avaliação de CrossMDA

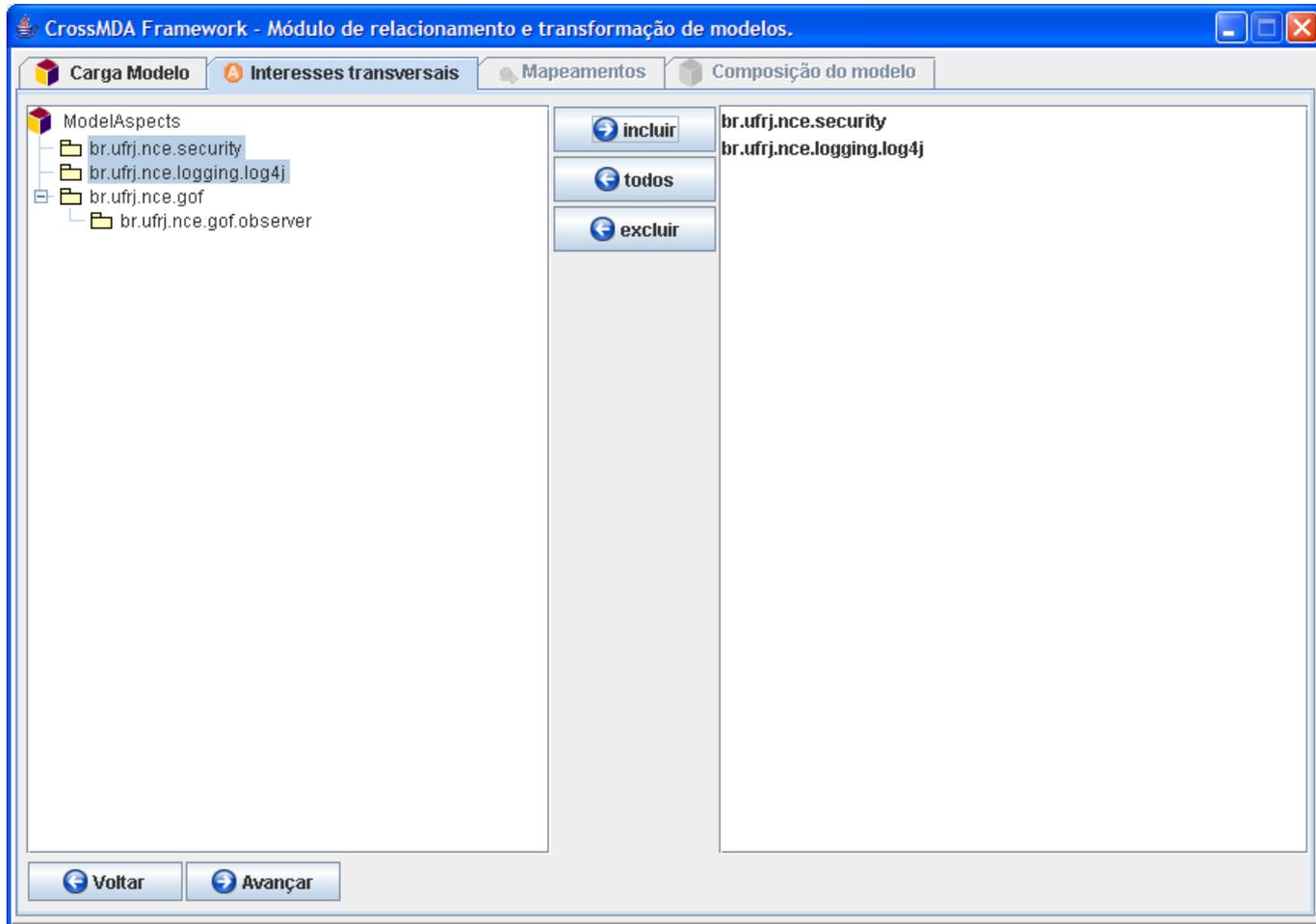
---

- Primeiro estudo de caso: E-commerce
  - Modelo PIM de Aspectos
    - Logging e Autenticação
  - Modelo PIM de negócio (módulo de pedido e cliente)
  - Relacionar Aspectos x Negócio e gerar programa de transformação
  - Geração do modelo PSM de Aspectos
- Validação do Protótipo CrossMDA
  - Desenvolvido em Java
  - InterFace Swing
  - Versão 1.0RC1

# implementação Protótipo CrossMDA



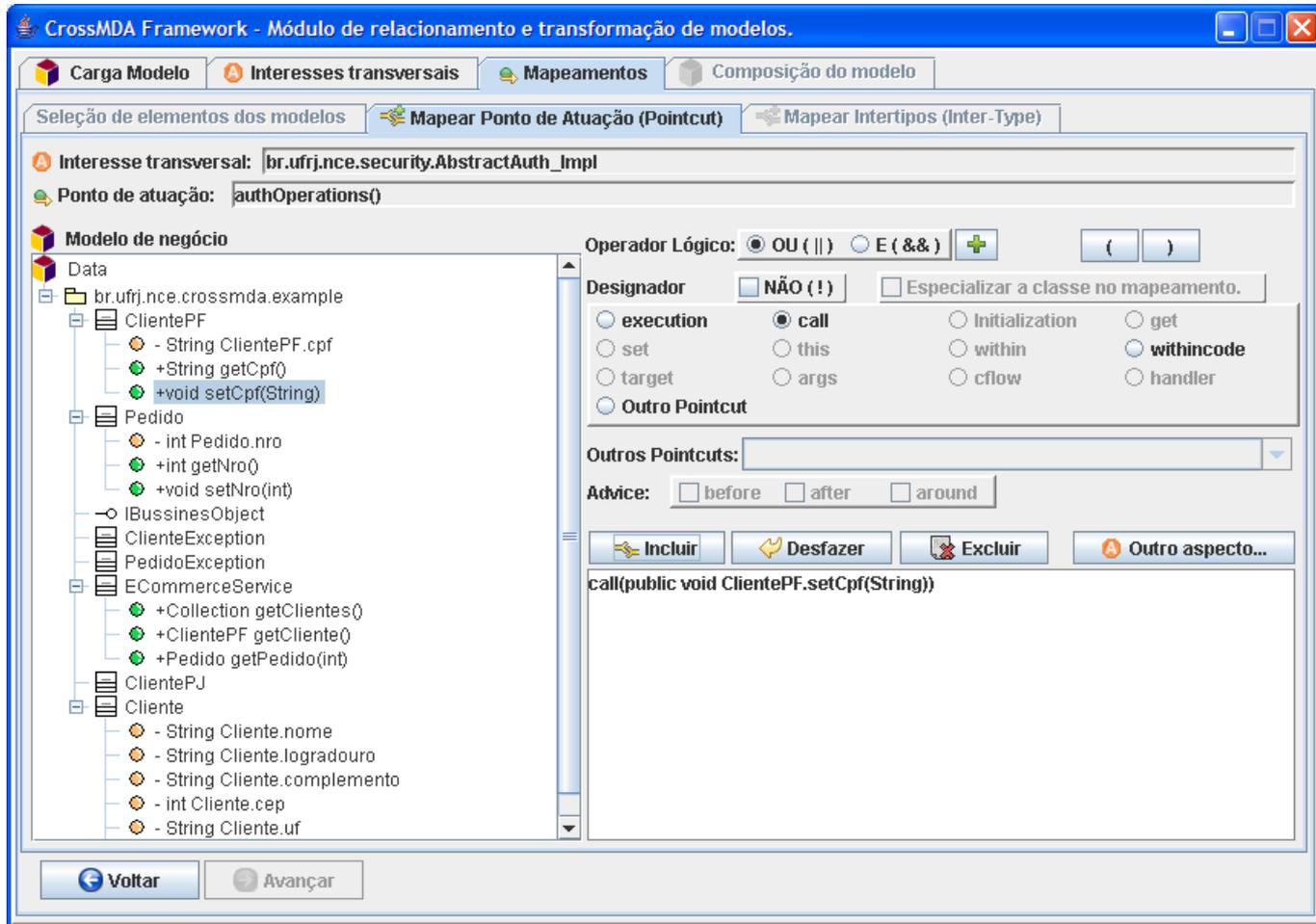
# Contribuição de implementação



# Contribuição de implementação

The screenshot displays the 'CrossMDA Framework - Módulo de relacionamento e transformação de modelos.' interface. The window has a blue title bar and a menu bar with 'Carga Modelo', 'Interesses transversais', 'Mapeamentos', and 'Composição do modelo'. Below the menu bar, there are two tabs: 'Seleção de elementos dos modelos' and 'Mapear Ponto de Atuação (Pointcut)'. The main workspace is divided into two panels. The left panel, titled 'Interesses transversais selecionados', shows a tree view of 'ModelAspects' with sub-items for 'br.ufrj.nce.security' (containing 'AbstractAuth' with a pointcut) and 'br.ufrj.nce.logging.log4j' (containing 'AbstractTrace' with two pointcuts). The right panel, titled 'Modelo para Composição', shows a tree view of 'Weaving' with sub-items for 'br.ufrj.nce.security.AbstractAuth\_Impl' (containing 'advice' with a pointcut and 'dependency' pointing to 'ClientePF <<Entity>>'). At the bottom of the interface, there are buttons for 'Pointcut...', 'Inter-Type...', 'Voltar', and 'Avançar'.

# Contribuição de implementação

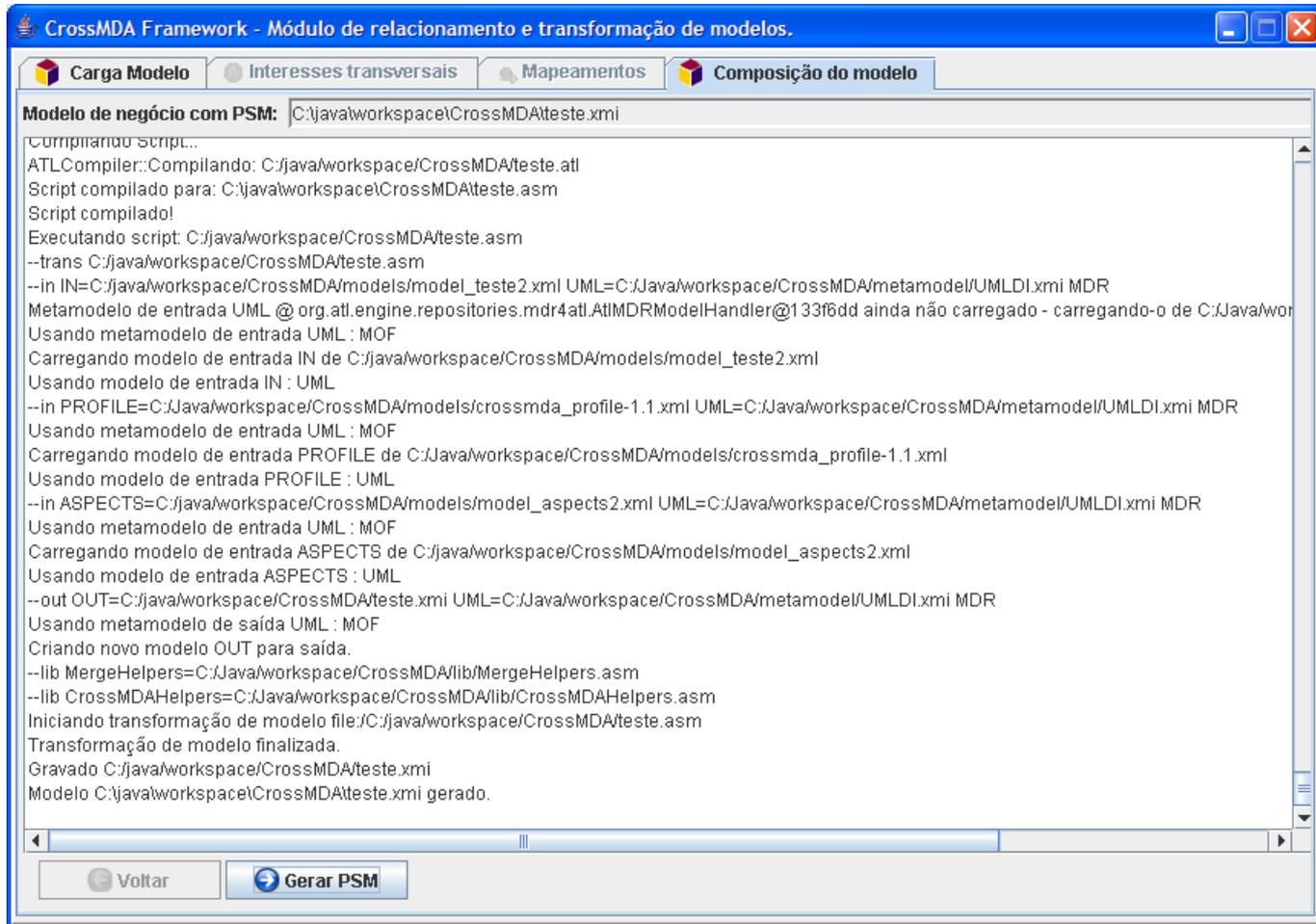


# Contribuição de implementação

The screenshot displays the 'CrossMDA Framework - Módulo de relacionamento e transformação de modelos.' interface. The main window is titled 'Mapear Intertipos (Inter-Type)' and contains the following elements:

- Navigation Tabs:** Carga Modelo, Interesses transversais, Mapeamentos, and Composição do modelo.
- Sub-Tabs:** Seleção de elementos dos modelos, Mapear Ponto de Atuação (Pointcut), and Mapear Intertipos (Inter-Type).
- Left Panel (Interesses transversal selecionado):** Shows a tree view with 'ModelAspects' containing 'AbstractAuth' and a selected method '- authenticate()'. A green dot indicates the selected element.
- Right Panel (Modelo de negócio):** Shows a tree view of a business model with classes like 'ClientePF', 'Pedido', 'IBussinesObject', 'ECommerceService', and 'Cliente'. Methods like '+String getCpf()', '+void setCpf(String)', '+int getNro()', and '+void setNro(int)' are listed.
- Configuration Panel (Right):** Contains settings for 'Tipo de mapeamento' (Parents selected), 'Tipo do Declare Parents' (Extends selected), and buttons for 'Incluir' and 'Excluir' under both 'Parents' and 'Introduction' sections. An 'Outro aspecto...' button is also present.
- Bottom Section:** Labeled 'Declare Parents' and 'Introductions', with empty text boxes for configuration.
- Footer:** 'Voltar' and 'Avançar' buttons.

# Contribuição de implementação



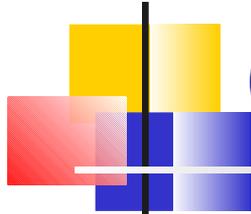
CrossMDA Framework - Módulo de relacionamento e transformação de modelos.

Carga Modelo | Interesses transversais | Mapeamentos | Composição do modelo

Modelo de negócio com PSM: C:\java\workspace\CrossMDA\teste.xml

```
Corripilando script...
ATLCompiler::Compilando: C:\java\workspace\CrossMDA\teste.atl
Script compilado para: C:\java\workspace\CrossMDA\teste.asm
Script compilado!
Executando script: C:\java\workspace\CrossMDA\teste.asm
--trans C:\java\workspace\CrossMDA\teste.asm
--in IN=C:\java\workspace\CrossMDA\models\model_teste2.xml UML=C:\Java\workspace\CrossMDA\metamodel/UMLDI.xml MDR
Metamodelo de entrada UML @ org.atl.engine.repositories.mdr4atl.AtIMDRModelHandler@133f6dd ainda não carregado - carregando-o de C:\Java\wor
Usando metamodelo de entrada UML : MOF
Carregando modelo de entrada IN de C:\java\workspace\CrossMDA\models\model_teste2.xml
Usando modelo de entrada IN : UML
--in PROFILE=C:\Java\workspace\CrossMDA\models\crossmda_profile-1.1.xml UML=C:\Java\workspace\CrossMDA\metamodel/UMLDI.xml MDR
Usando metamodelo de entrada UML : MOF
Carregando modelo de entrada PROFILE de C:\Java\workspace\CrossMDA\models\crossmda_profile-1.1.xml
Usando modelo de entrada PROFILE : UML
--in ASPECTS=C:\java\workspace\CrossMDA\models\model_aspects2.xml UML=C:\Java\workspace\CrossMDA\metamodel/UMLDI.xml MDR
Usando metamodelo de entrada UML : MOF
Carregando modelo de entrada ASPECTS de C:\java\workspace\CrossMDA\models\model_aspects2.xml
Usando modelo de entrada ASPECTS : UML
--out OUT=C:\java\workspace\CrossMDA\teste.xml UML=C:\Java\workspace\CrossMDA\metamodel/UMLDI.xml MDR
Usando metamodelo de saída UML : MOF
Criando novo modelo OUT para saída.
--lib MergeHelpers=C:\Java\workspace\CrossMDA\lib\MergeHelpers.asm
--lib CrossMDAHelpers=C:\Java\workspace\CrossMDA\lib\CrossMDAHelpers.asm
Iniciando transformação de modelo file:C:\java\workspace\CrossMDA\teste.asm
Transformação de modelo finalizada.
Gravado C:\java\workspace\CrossMDA\teste.xml
Modelo C:\java\workspace\CrossMDA\teste.xml gerado.
```

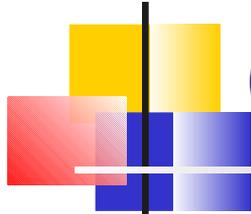
Voltar | Gerar PSM



# Conclusões

---

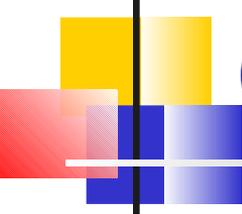
- Abordagem de transformação MDA
  - Automatizar o processo de integração dos interesses transversais através da transformação do PIM para PSM
    - permite que a modelagem de interesses transversais seja ortogonal à modelagem dos processos de negócio
- Técnicas de POA auxilia nas atividades de mapeamento dos relacionamentos e na composição do modelo



# Conclusões

---

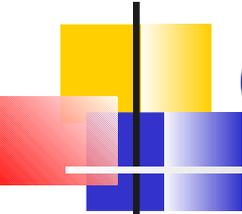
- Promover um alto grau de reuso de artefatos
  - Artefatos de transformação
    - *templates* em ATL para gerar programas de transformação
      - facilita a manutenção
      - permite novas implementações sem alterar o código de CrossMDA



# Conclusões

---

- Artefatos de modelo
  - Modelos PIM de aspectos
    - desenvolvidos por qualquer projetista
    - reutilizados em várias transformações
  - Modelos PIM de negócio
    - reaproveitados e entrelaçados com diferentes modelos de aspectos
      - gerar sistemas que necessitem de diferentes requisitos computacionais



# Conclusões

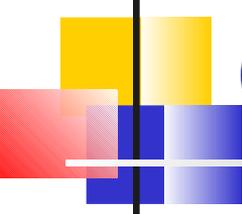
---

- Geração de modelos PSM de aspectos
  - Uso tecnologias MDA padrão
    - XMI
      - Permite que o modelo gerado possa ser utilizado por ferramentas de transformação modelo-texto
- Metamodelos
  - Uso do Metamodelo da UML para PIM e PSM
    - Independência de ferramenta de modelagem
- Baixo grau de acoplamento
  - Independência entre modelos PIM e PSM de aspectos e de negócio
    - evolução sem interferência mútua

# Novos Trabalhos (em andamento)

---

- Processo de combinação de aspectos em CrossMDA
  - Gerar novos aspectos através da combinação de outros aspectos
    - PIM -> PIM
- Transformação modelo-texto
  - Gerar código fonte dos modelos PSM de aspectos
    - Trabalho de conclusão de graduação na UFRN
    - Orientação da Profa. Flávia C. Delicato, D.Sc.
- Novo estudo de caso



# CrossMDA

---

- Obrigado!
- Perguntas?
- contato: [mpitanga@gmail.com](mailto:mpitanga@gmail.com)