

Universidade Federal do Rio Grande do Norte
Centro de Ciências Exatas e da Terra
Departamento de Informática e Matemática Aplicada
Programa de Pós-Graduação em Sistemas e Computação

Dos Requisitos à Arquitetura em Linhas de Produtos de Software: Uma Estratégia de Transformações entre Modelos

Keivilany Coelho

Orientador (a): Profa. Dra. Thais Vasconcelos Batista

Introdução

- Com o objetivo de integrar as fases de definição de requisitos e arquitetura de LPS o presente trabalho propõe MaRiPLA (*Mapping Requirements to Product Line Architecture*)
 - Uma estratégia de mapeamento bi-direcional entre modelo de metas, descrito em PL-AOV-Graph e descrição arquitetural, em PL-AspectualACME, buscando garantir a correspondência entre as duas fases iniciais do desenvolvimento de LPS, com técnicas MDD, a linguagem de transformação ATL e os frameworks Xtext e ACCELEO, no ambiente Eclipse

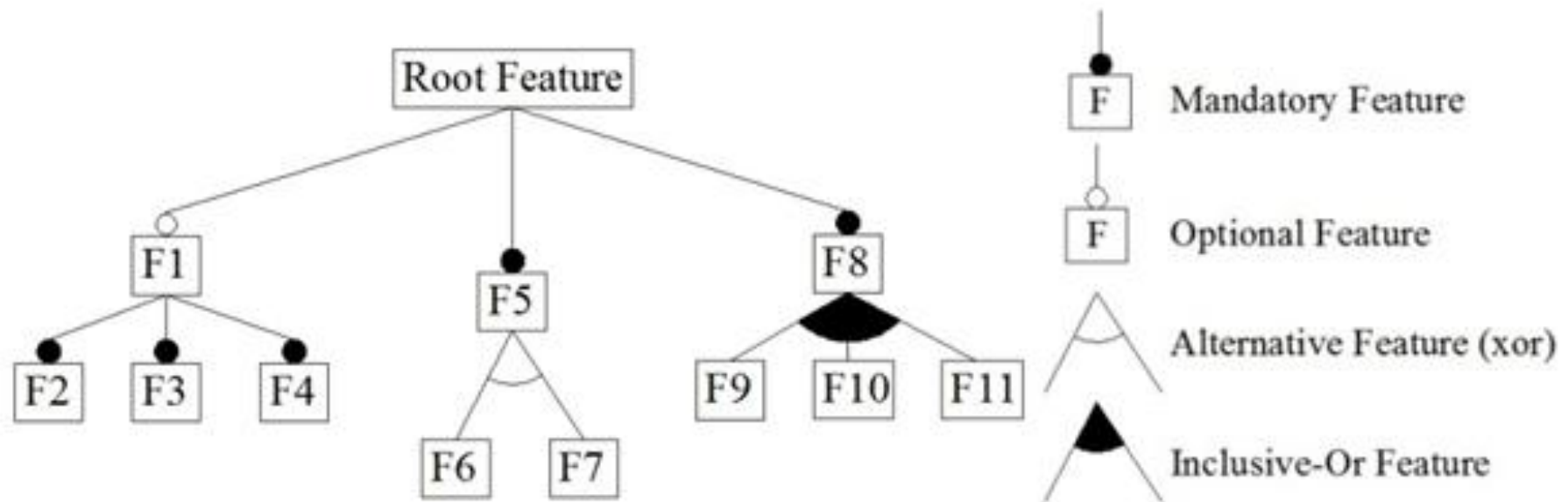
Objetivos

- Definir o mapeamento bidirecional entre PL-AOV-Graph PL-AspectualACME
 - Definir o processo de mapeamento
 - Definir as regras de mapeamento
- Desenvolver a ferramenta MaRiPLA, que automatiza as transformações entre as duas especificações, com as tecnologias:
 - Xtext, nas transformações entre descrição textual e modelo
 - ATL, nas transformações entre modelos
 - ACCELEO, nas transformações entre modelo e descrição textual

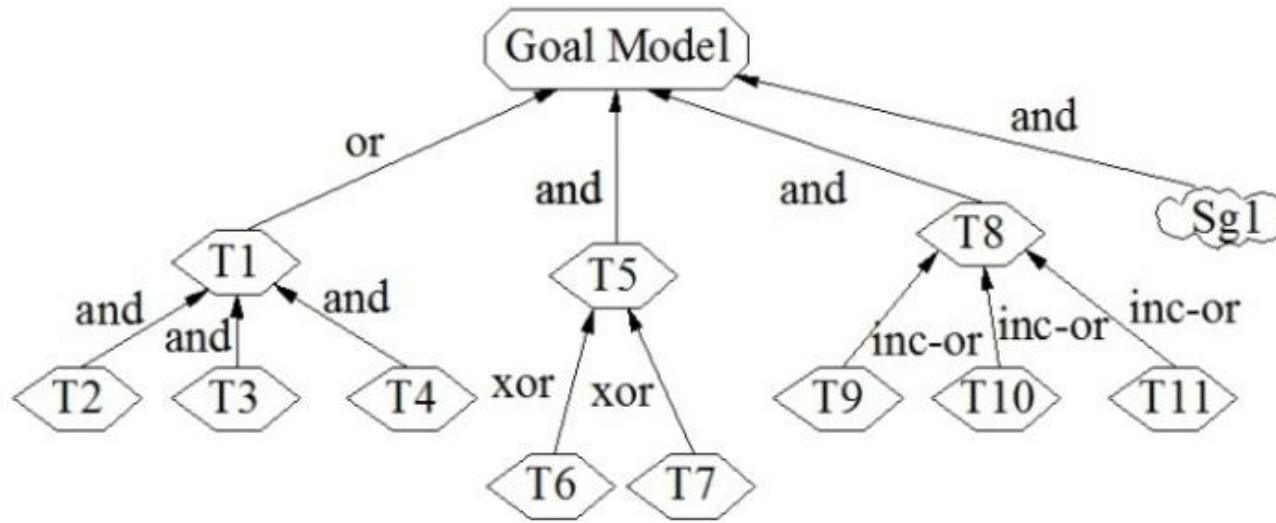
Motivação

- Na engenharia de LPS, a descrição arquitetural obtida a partir de modelos de metas permite que a arquitetura da LPS seja mais detalhada e consistente com os requisitos especificados na fase de levantamento de requisitos
- Beneficia, também, as fases posteriores do ciclo de vida do software

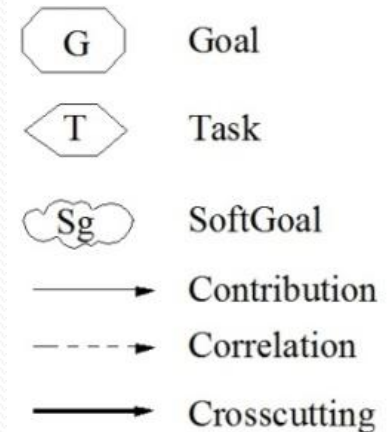
Linhas de Produto de Software e Modelos de Features



PL-AOV-Graph*



```
Goal Model {
  Task T1 (or; T1;){
    task T2 (and; T1.1;){}
    task T3 (and; T1.2;){}
    task T4 (and; T1.3;){}
  }
  Task T5 (and; T2;){
    task T6 (xor; T2.1;){}
    task T7 (xor; T2.2;){}
  }
  Task T8 (and; T3;){
    task T9 (inc-or; T3.1;){}
    task T10 (inc-or; T3.2;){}
    task T11 (inc-or; T3.3;){}
  }
  SoftGoal Sg1 (and; S1;){}
}
```



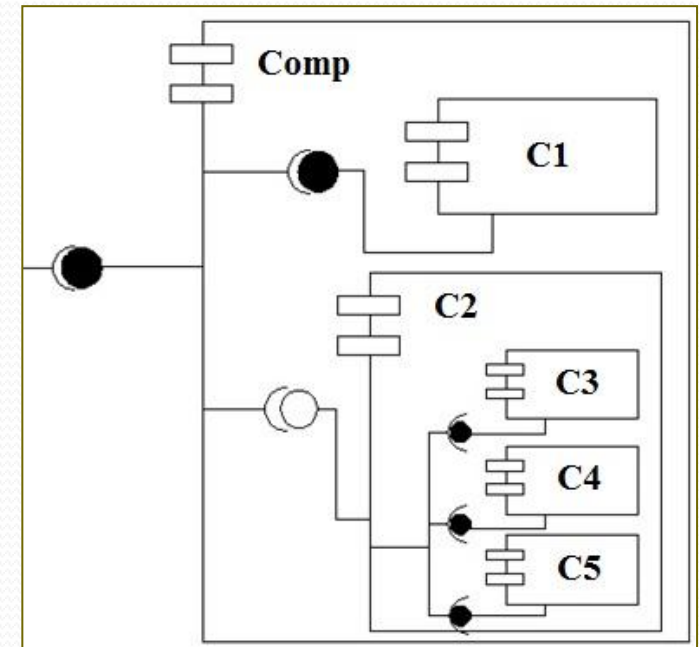
*Santos, L. O. PL-AOVgraph: Uma extensão de AOV-Graph para Linha de Produto de Software. Natal, 2010. 84p. Technical Report UERN.
<https://sites.google.com/site/plaovgraph>

PL-AspectualACME*

```

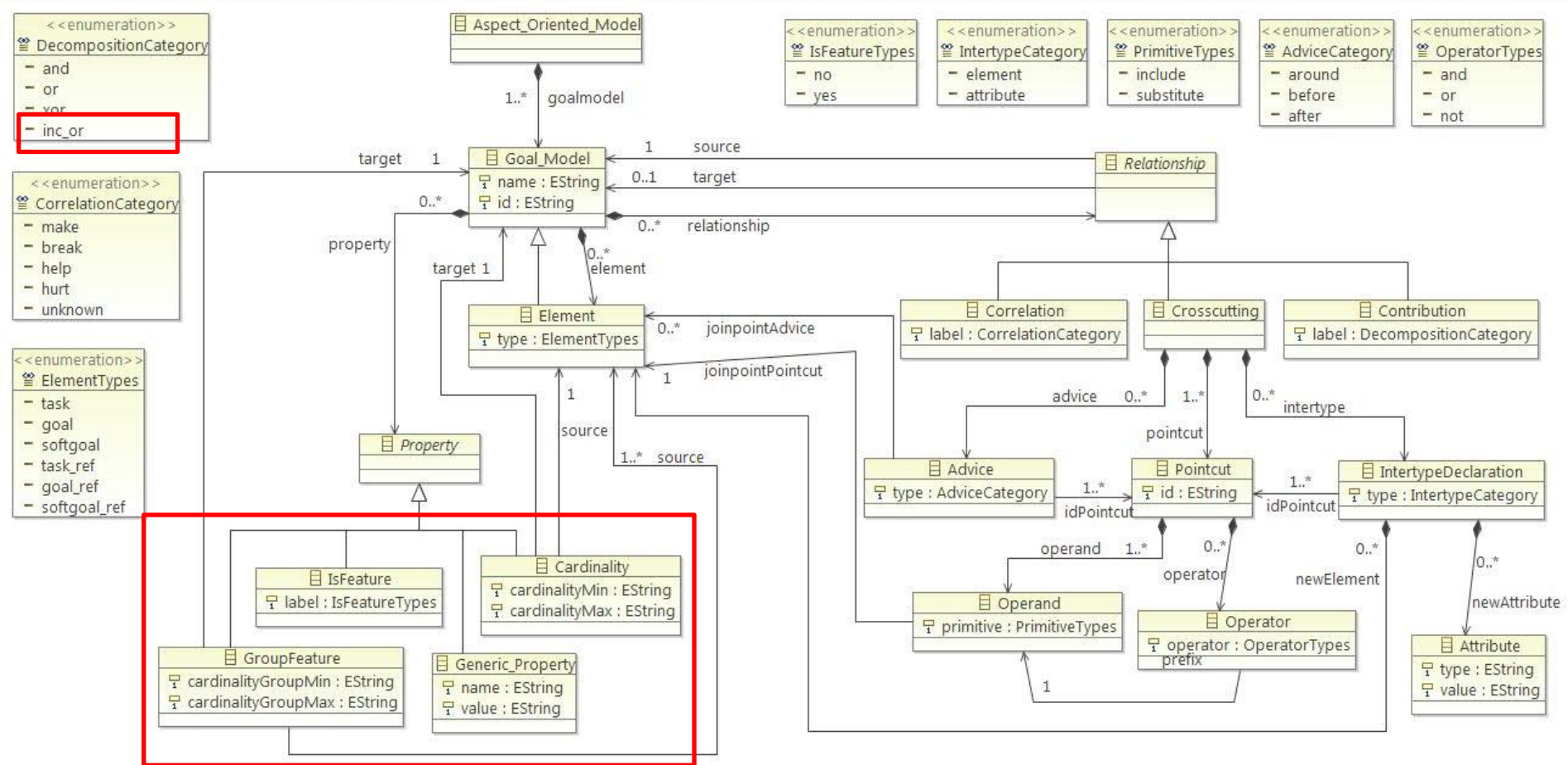
Family Exemplo : new ProductLineT extended with {
Component Comp : new VariationPointT extended with {
  Port comp;
  Port vp : new VariationPortT extended with {};
Component C1 {Port c1};
Property variants = {C2};
Representation C2 = {
  System C2 = {
    Component C2 : new OptionalT extended with {
      Port c2;
      Port c2vp : new VariationPortT extended with {};
      Property variants = {C3, C4, C5};
      Representation C3 = {
        System C3 = {
          Component C3 : new InclusiveOrT extended with {
            Port c3;
          }
          Binding C2.c2vp to C3.c3;
        }
      }
      Representation C4 = { ... }
      Representation C5 = { ... }
    }
    Binding Comp.vp to C2.c2;
  }
}

```

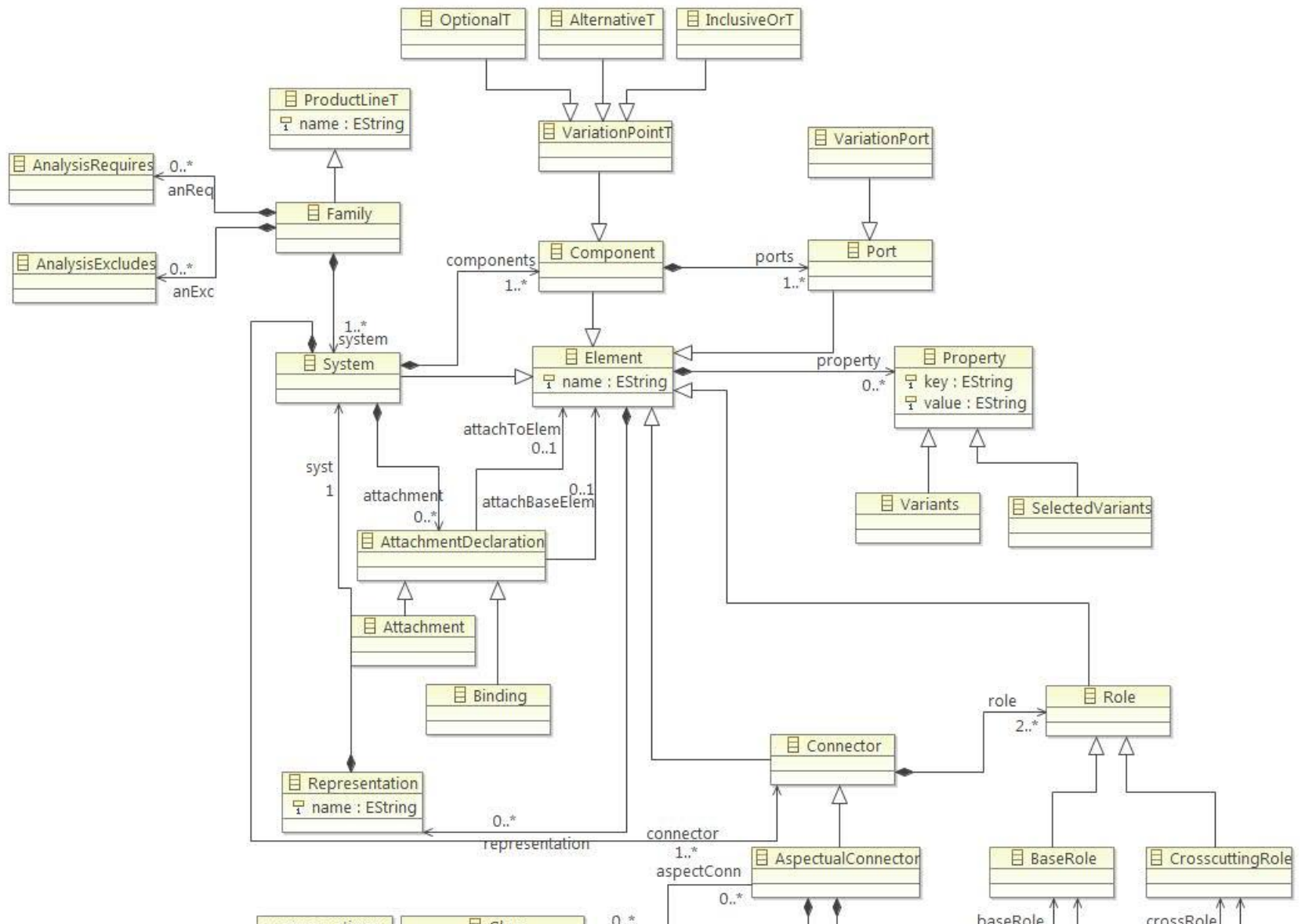


*Barbosa, EA.; Batista, T.; Garcia, A. and Silva, E. PL-AspectualACME: An Aspect-Oriented Architectural Description Language for Software Product Lines. In Proceedings of the 5th European Conference on Software Architecture (ECSA '2011)

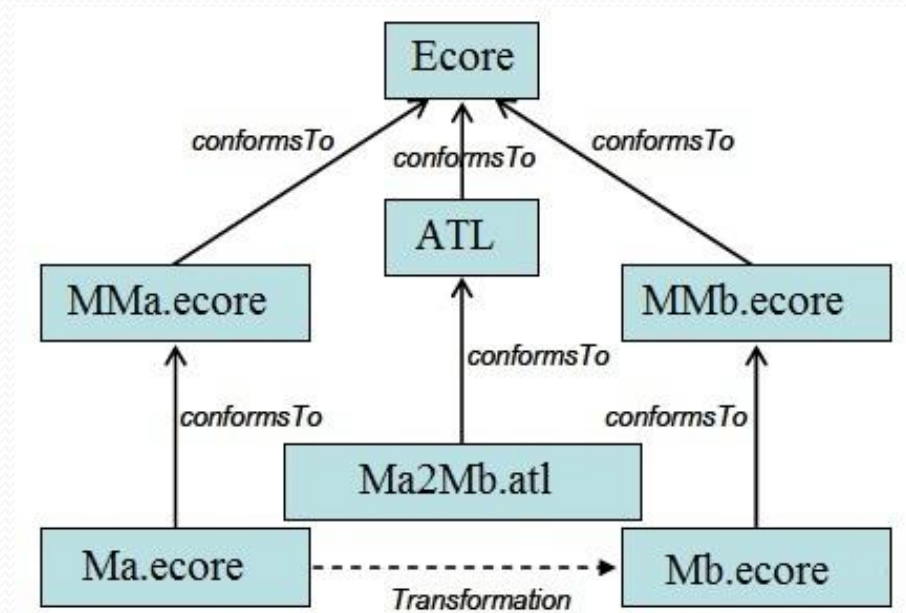
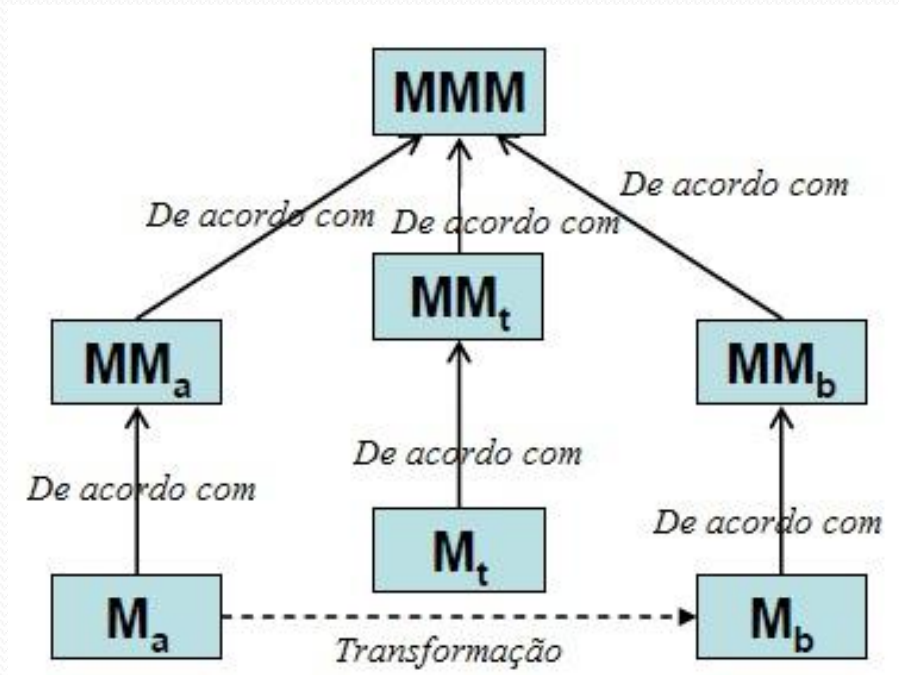
Metamodelo de PL-AOV-Graph



Metamodelo de PL-AspectualACME



Transformações entre Modelos com MDD e ATL



Regras de Transformação

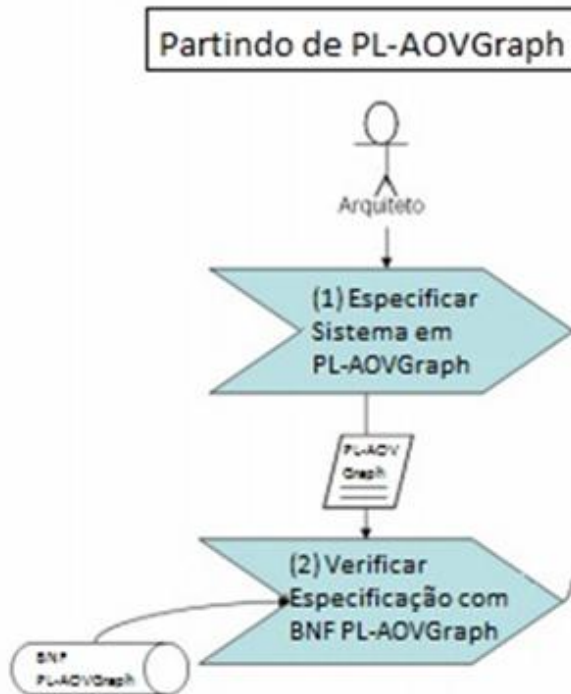
PL-AOVGraph	→	PL-AspectualACME
GoalModel		Family
Task/Goal/Softgoal		Component
Correlation Make		Analysis Requires
Correlation Break		Analysis Excludes
Demais Correlations		Property
Task_Ref, Goal_Ref, SoftGoal_Ref		Property Selected Variants
Relacionamento Transversal		Attachments com AspectualConnector

Regras de Transformação

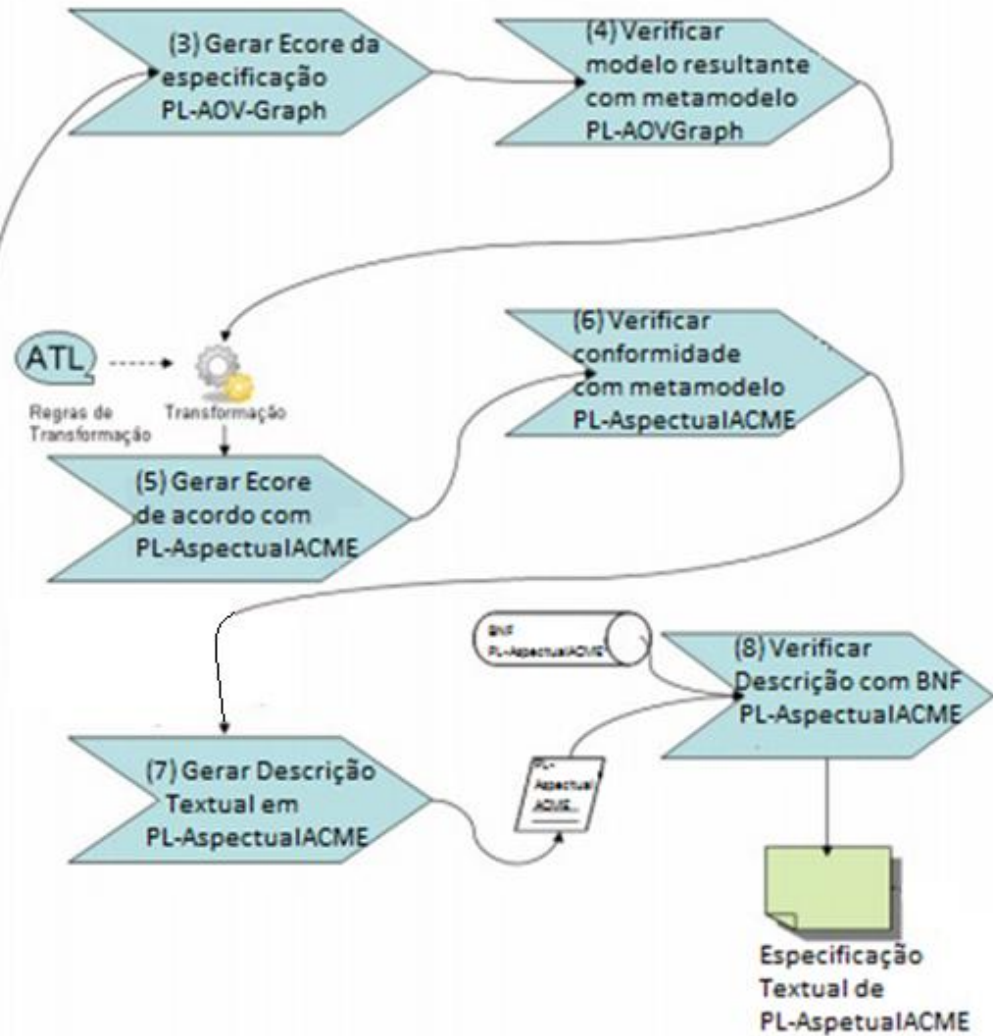
PL-AspectualACME	→	PL-AOVGraph
Family		GoalModel
Component		Task/Goal/Softgoal
Analysis Requires		Correlation Make
Analysis Excludes		Correlation Break
Property		Property
Property Selected Variants		Task_Ref, Goal_Ref, SoftGoal_Ref
Attachments com AspectualConnector		Relacionamento Transversal

Processo de Mapeamento

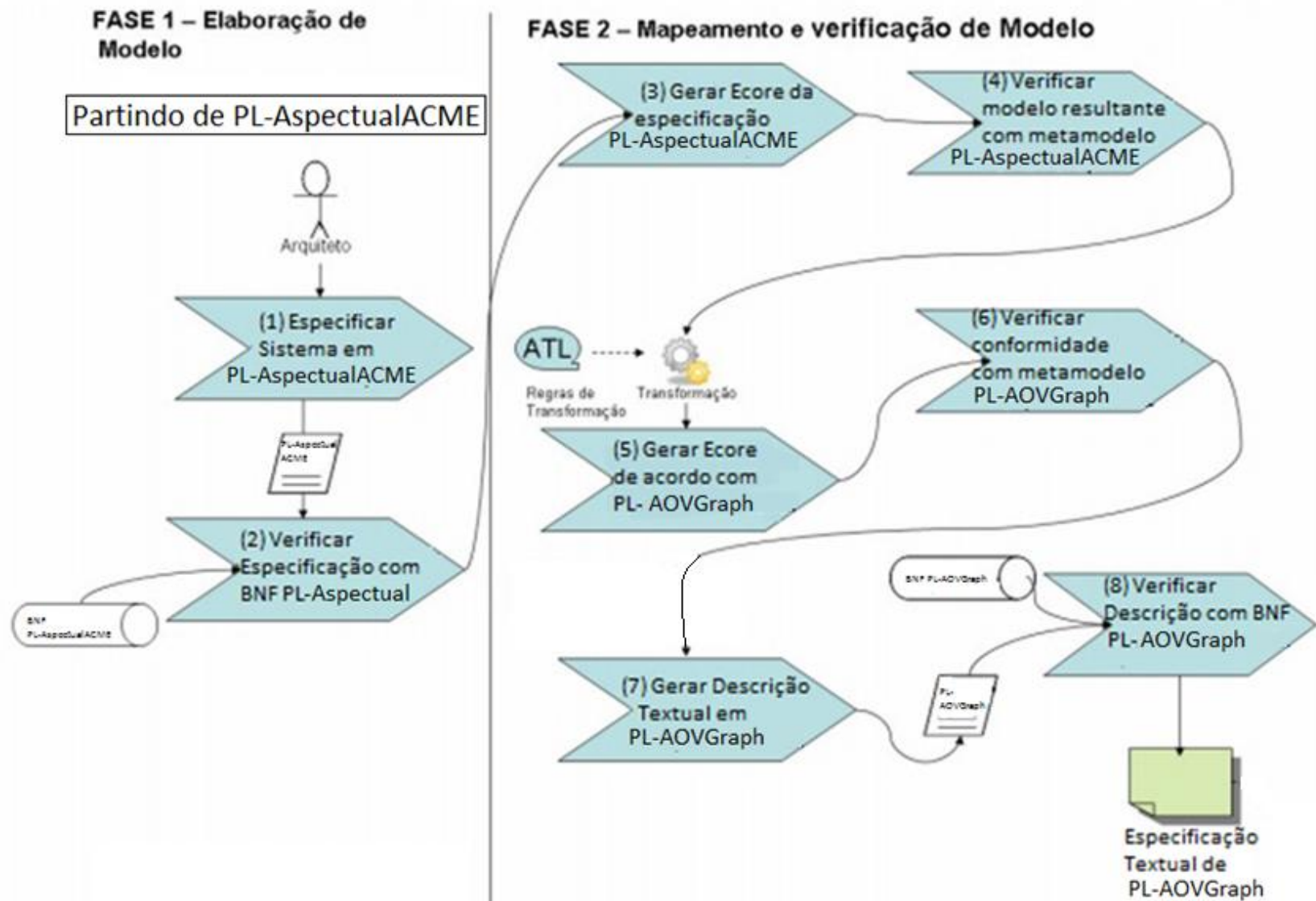
FASE 1 – Elaboração de Modelo



FASE 2 – Mapeamento e verificação de Modelo



Processo de Mapeamento

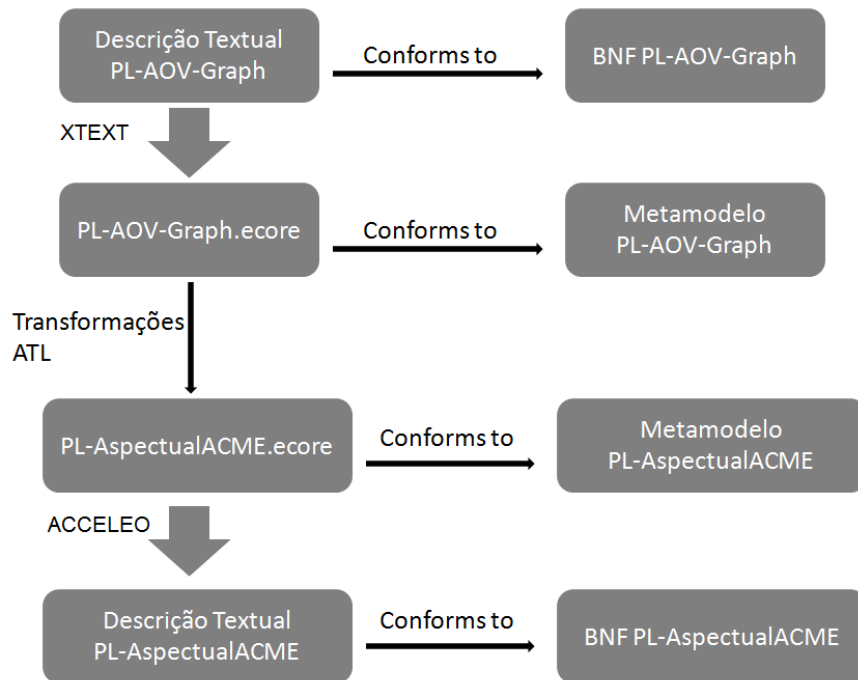


Tecnologias

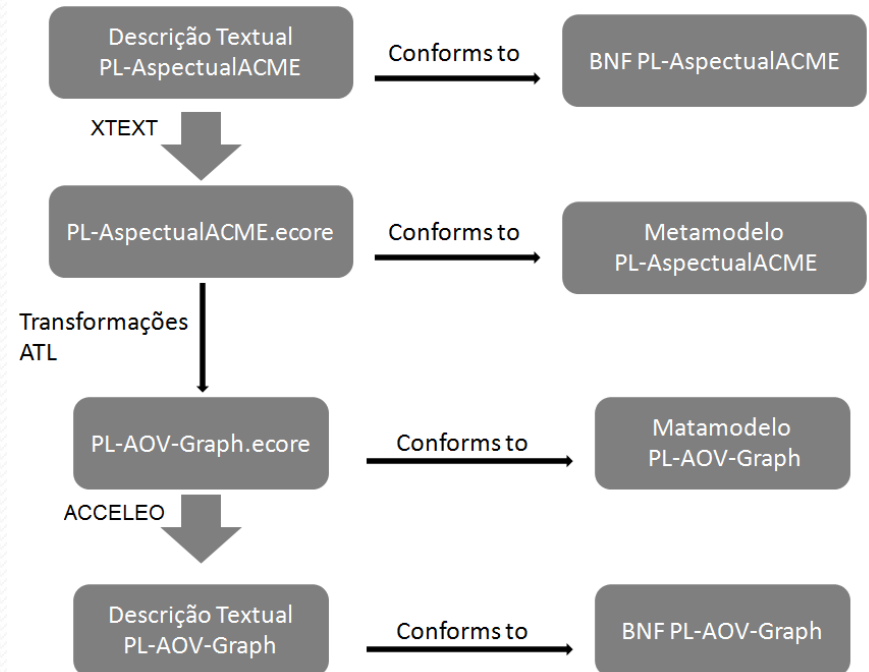
- XTEXT – Framework java que fornece uma API para desenvolvimento de DSL
 - www.xtext.org/
- ATL – Atlas Transformation Language – Linguagem de transformação de modelos baseada em OCL
 - <http://www.eclipse.org/atl/>
- ACCELEO – Gerador de código, que transforma modelos em código , com tecnologia MDA
 - <http://www.acceleo.org/pages/home/en>

MaRiPLA – Mapping Requirements to Product Line Architecture

Dos Requisitos à Arquitetura

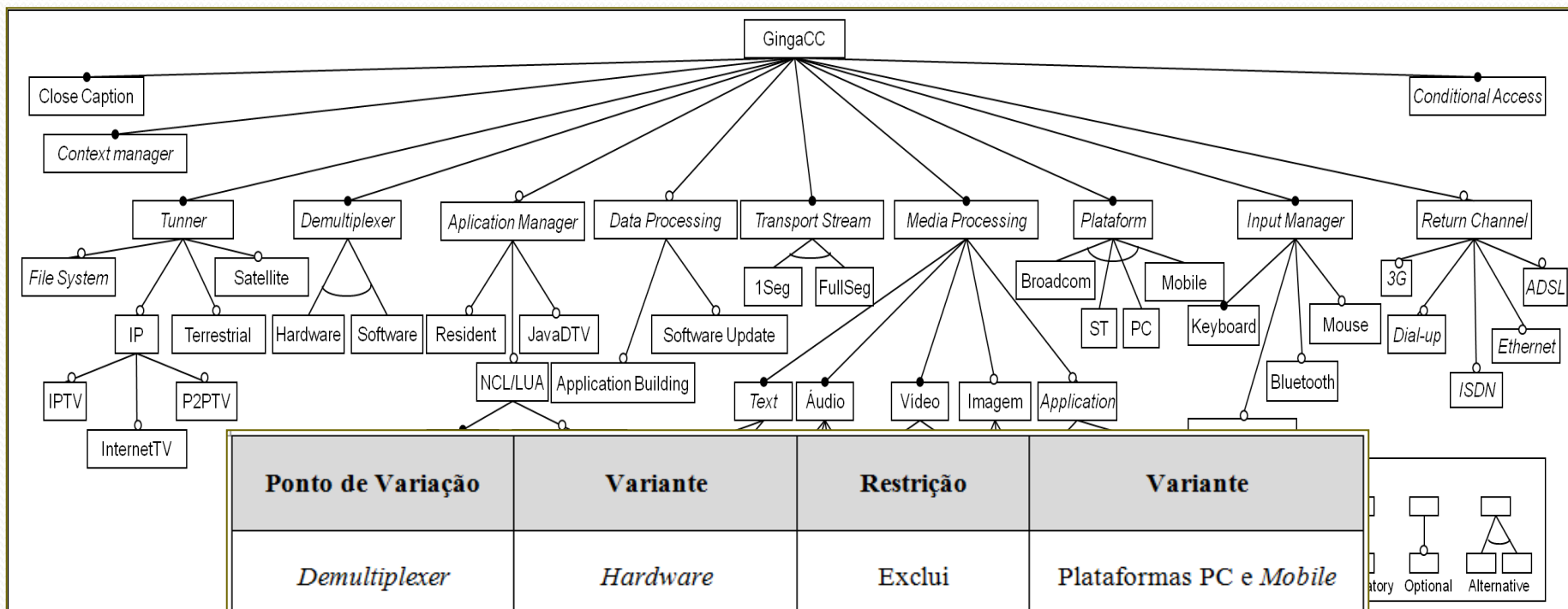


Da Arquitetura aos Requisitos



Estudo de Caso: Ginga ForAll

- Trabalho de refatoramento do middleware de TV digital ginga, fornecendo uma versão baseada em LPS



Ponto de Variação	Variante	Restrição	Variante
<i>Demultiplexer</i>	<i>Hardware</i>	Exclui	Plataformas PC e Mobile
<i>Application Manager</i>	JavaDTV ou NCL/Lua	Requer	<i>Data Processing</i> com <i>Application Building</i>
	NCL/Lua <i>Complementary</i>	Requer	NCL/Lua <i>Basic</i>
	NCL/Lua <i>Basic</i> + <i>Complementary</i>	Requer	Todos os formatos de mídia do <i>Media Processing</i>

PL-AOV-Graph Ginga ForAll

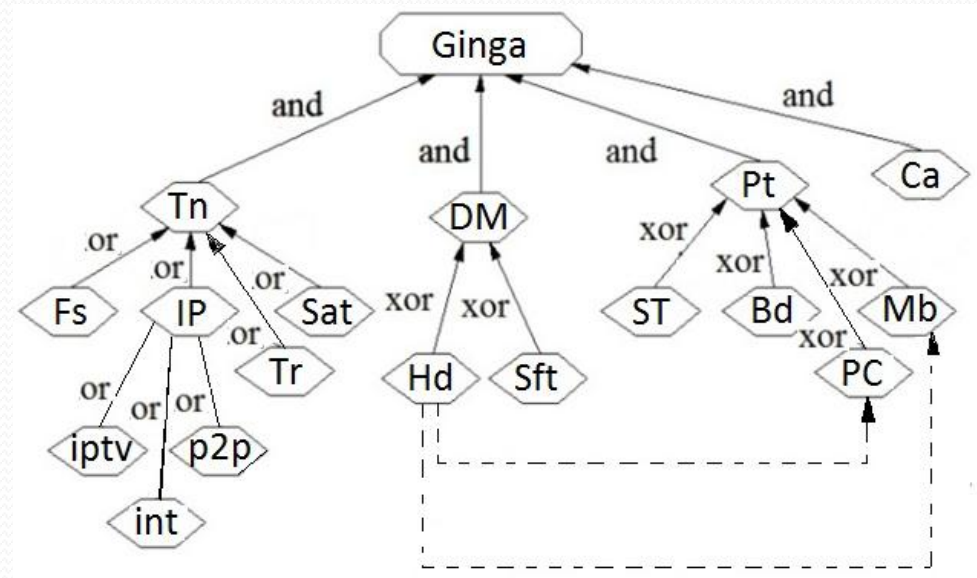
```

Goal_model (GingaCC; GM1){
  Task Tunner (and; T1) {
    Task FileSystem (or; T1.1) {}
    Task IP (or; T1.2) {
      Task IPTV (or; T1.2.1) {}
      Task InternetTV (or; T1.2.2) {}
      Task P2PTV (or; T1.2.3) {}
    }
    Task Terrestrial (or; T1.3) {}
    Task Satellite (or; T1.4) {}
  }
  Task Platform (and; T2){
    Task ST (xor; T2.1){}
    Task Broadcom (xor; T2.2){}
    Task PC (xor; T2.3){}
    Task Mobile (xor; T2.4){}
  }
  Task ConditionalAccess (and; T3){}
  Task Demultiplexer (and; T4) {
    Task Hardware (xor; T4.1) {}
    Task Software (xor; T4.2) {}
  }
}
    
```

```

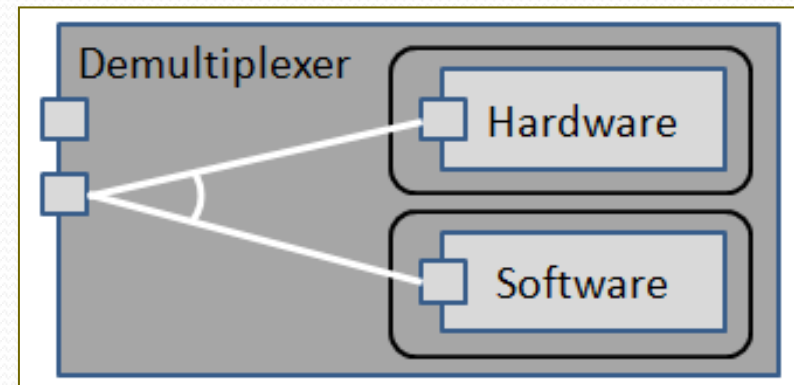
Correlation (break){
  Source = task_ref = (Hardware;T4.1;)
  Target = task_ref = (PC;T2.3;)
}

Correlation (break){
  Source = task_ref = (Hardware;T4.1;)
  Target = task_ref = (Mobile;T2.4;)
}
    
```



PL-AspectualACME Ginga ForAll

```
Component Demultiplexer : new VariationPointT
    extended with {
    Port demux;
    Port demuxVp : new VariationPortT extended with {};
    Property Variants = {Hardware, Software};
    Representation Hardware = {
        System Hardware = {
            Component Hardware : new AlternativeT extended
            with {
                Port hard;
            }
        }
    }
    Binding Demultiplexer.demuxVp to Hardware.hard;
}
Representation Software = {
    System Software = {
        Component Software : new AlternativeT extended
        with {
            Port Soft;
        }
    }
    Binding Demultiplexer.demuxVp to Software.soft;
}
}
```



Modelos

platform:/resource/MaRiPLA/Modelos2/PLAOV.xmi

- ◆ Aspect Oriented Model
 - ◆ Goal Model Ginga
 - ◆ Element Tunner
 - ◆ Element FileSystem
 - ◆ Contribution or
 - ▷ ◆ Element IP
 - ▷ ◆ Element Terrestrial
 - ▷ ◆ Element Satellite
 - ◆ Contribution and
 - ◆ Element Platform
 - ◆ Element ST
 - ◆ Contribution xor
 - ▷ ◆ Element Broadcom
 - ▷ ◆ Element PC
 - ▷ ◆ Element Mobile
 - ◆ Contribution and
 - ◆ Element ConditionalAccess
 - ◆ Contribution and
 - ◆ Element Demultiplexer
 - ◆ Element Hardware
 - ◆ Contribution xor
 - ◆ Correlation break
 - ◆ Correlation break

platform:/resource/MaRiPLA/Modelos2/PLASP.xmi

- ◆ Family Ginga
 - ◆ Analysis Excludes
 - ◆ Analysis Excludes
 - ◆ System GingaSys
 - ◆ Variation Point T Tunner
 - ▷ ◆ Representation FileSystem
 - ▷ ◆ Representation IP
 - ▷ ◆ Representation Terrestrial
 - ▷ ◆ Representation Satellite
 - ◆ Variants Variants
 - ◆ Port TunnerPort
 - ◆ Variation Port TunnerVariationPort
 - ◆ Variation Point T Platform
 - ▷ ◆ Representation ST
 - ▷ ◆ Representation Boadcom
 - ▷ ◆ Representation PC
 - ▷ ◆ Representation Mobile
 - ◆ Variants Variants
 - ◆ Port PlatformPort
 - ◆ Variation Port PlatformVariationPort
 - ▷ ◆ Component ConditionalAccess
 - ▷ ◆ Variation Point T Demultiplexer
 - ▷ ◆ Variation Point T MediaProcessing²⁰

Transformações Model2Model

- Modelos Resultantes...

Próximos Passos

- Concluir Implementação da Ferramenta:
 - Transformação Text2Model com XTEXT
 - Transformação Model2Text com ACCELEO
 - Plug-In
- Avaliação das regras de transformação
- Avaliação dos modelos
- Avaliar com outros estudos de caso

Tópicos em aberto

- Técnicas e critérios para avaliação dos modelos gerados
- Nomeação de features no modelo de features
 - Como os nomes das features em um modelo de features poderia auxiliar o engenheiro de software na definição de tasks, goals ou softgoals em PL-AOVgraph?
- Task/Goal/Softgoal vai sempre gerar componente?
 - Como definir heurísticas para determinar que feature específica vai gerar um componente, conector, porta, etc?
- Representação de relacionamento transversal no modelo de features



Obrigada!