A Tool to Convert a PIM model into a CORBA IDL Specification

Thaís Vasconcelos Batista
Teresa Raquel Nascimento

thais@ufrnet.br, teresa@ufrnet.br

Informatics Department
Federal University of Rio Grande do Norte
Natal – RN, Brazil
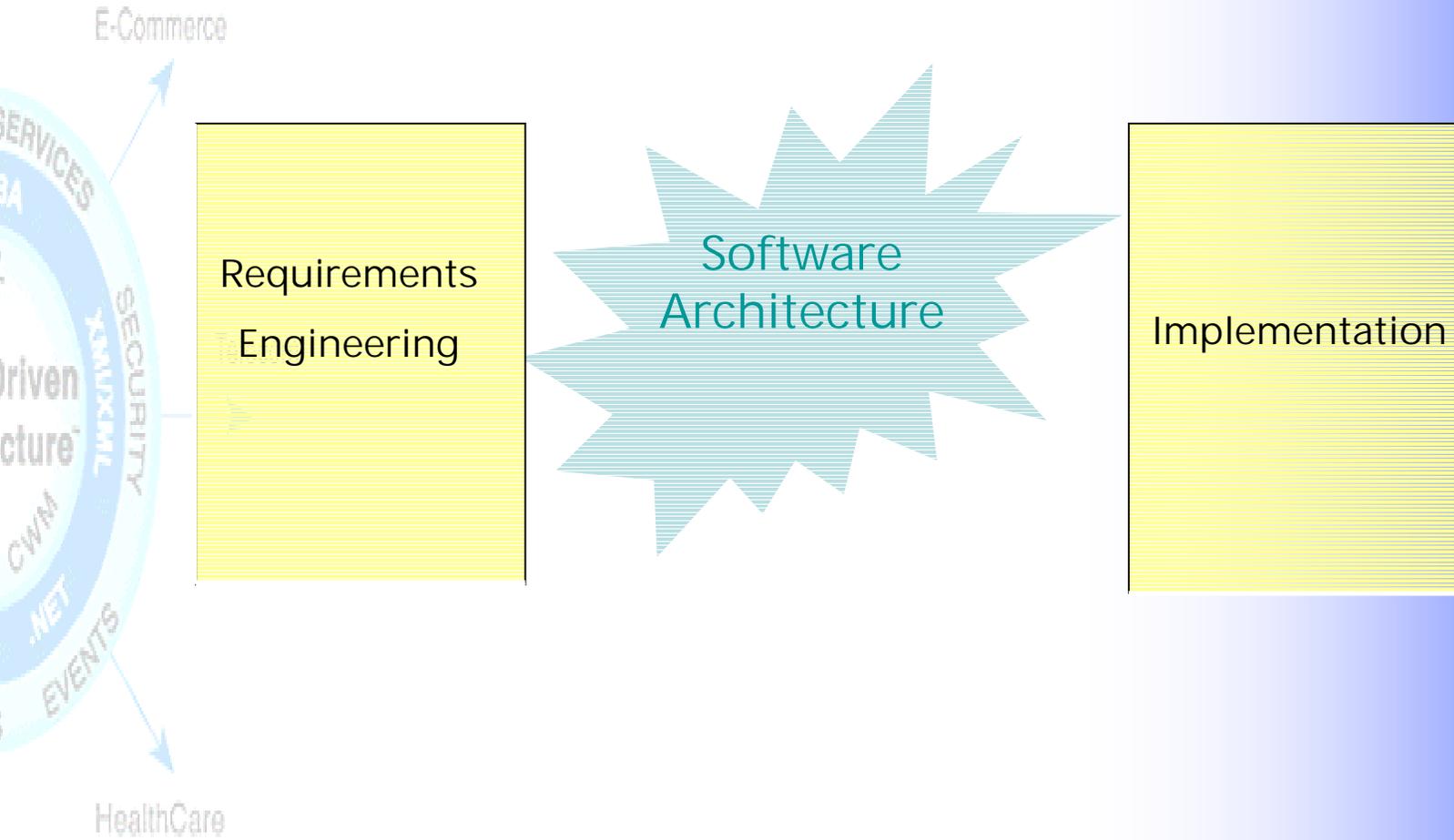*www.dimap.ufrn.br/~thais/*

TUPI

(Transformation one-to-one PIM-IDL)

Platform
Independent
Architectural
Description

Tupi

Platform
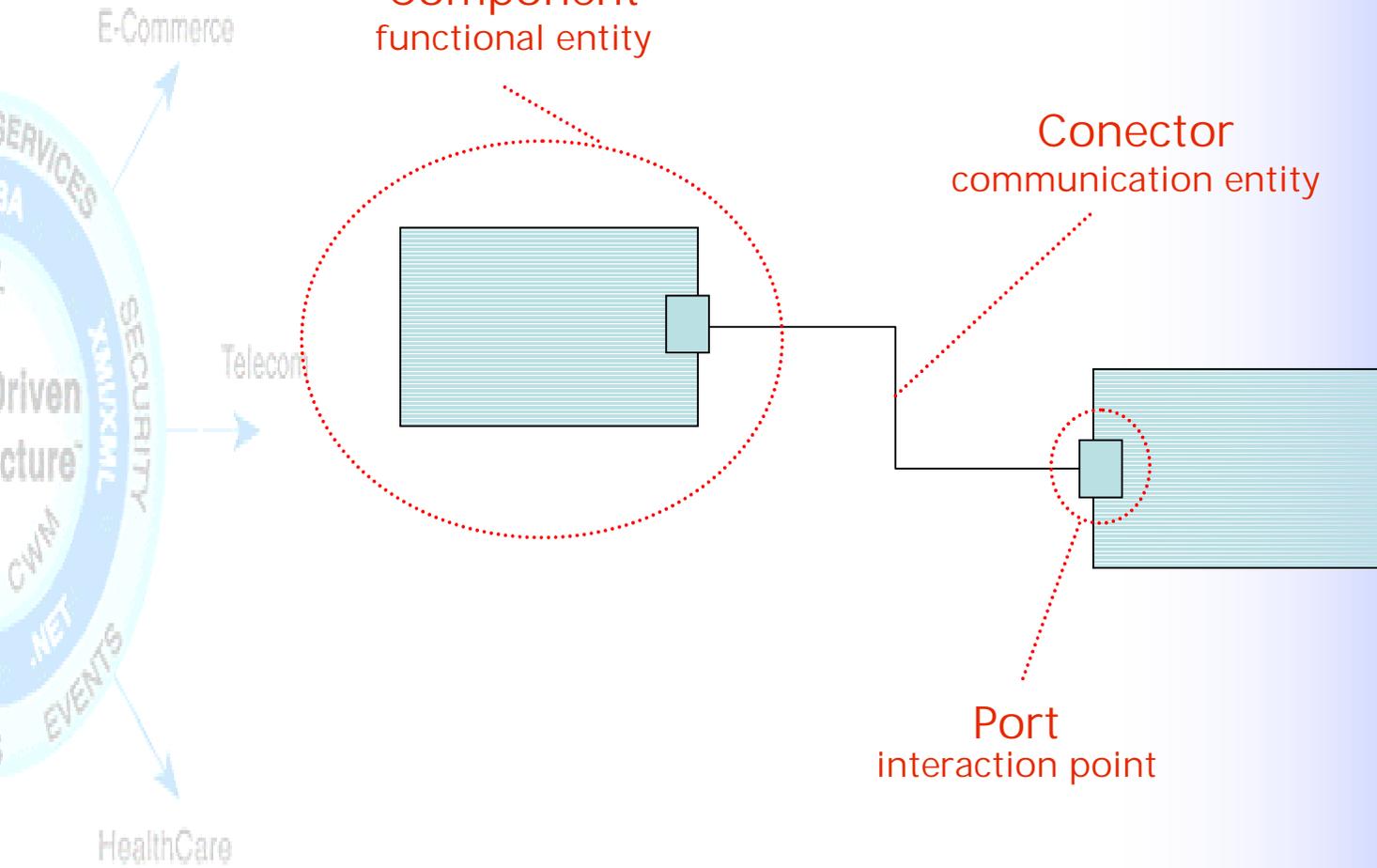Dependent
Architectural
Description

CORBA

Software Architecture[1]

Requirements Engineering

Software Architecture

Implementation

Software Architecture^2

Component
functional entity

Conector
communication entity

Port
interaction point

UFRN – CCET – DIMAp

MDA

Model-Driven Architecture

✓ Architectural Description Technique

   ✓ Based on UML

      ✓ Deals with Architectural Models

UML, XMI, MOF

OMG Standard

Middleware Platforms

Essential Services

UFRN – CCET – DIMAp
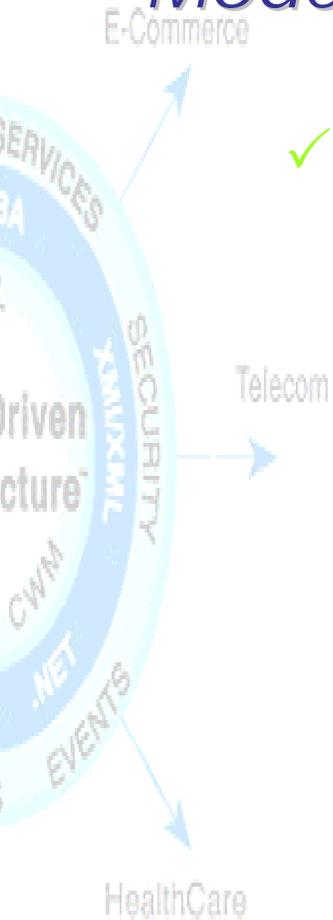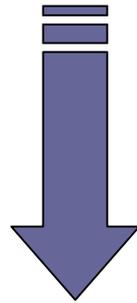
MDA Models

Platform Independent Model (PIM)
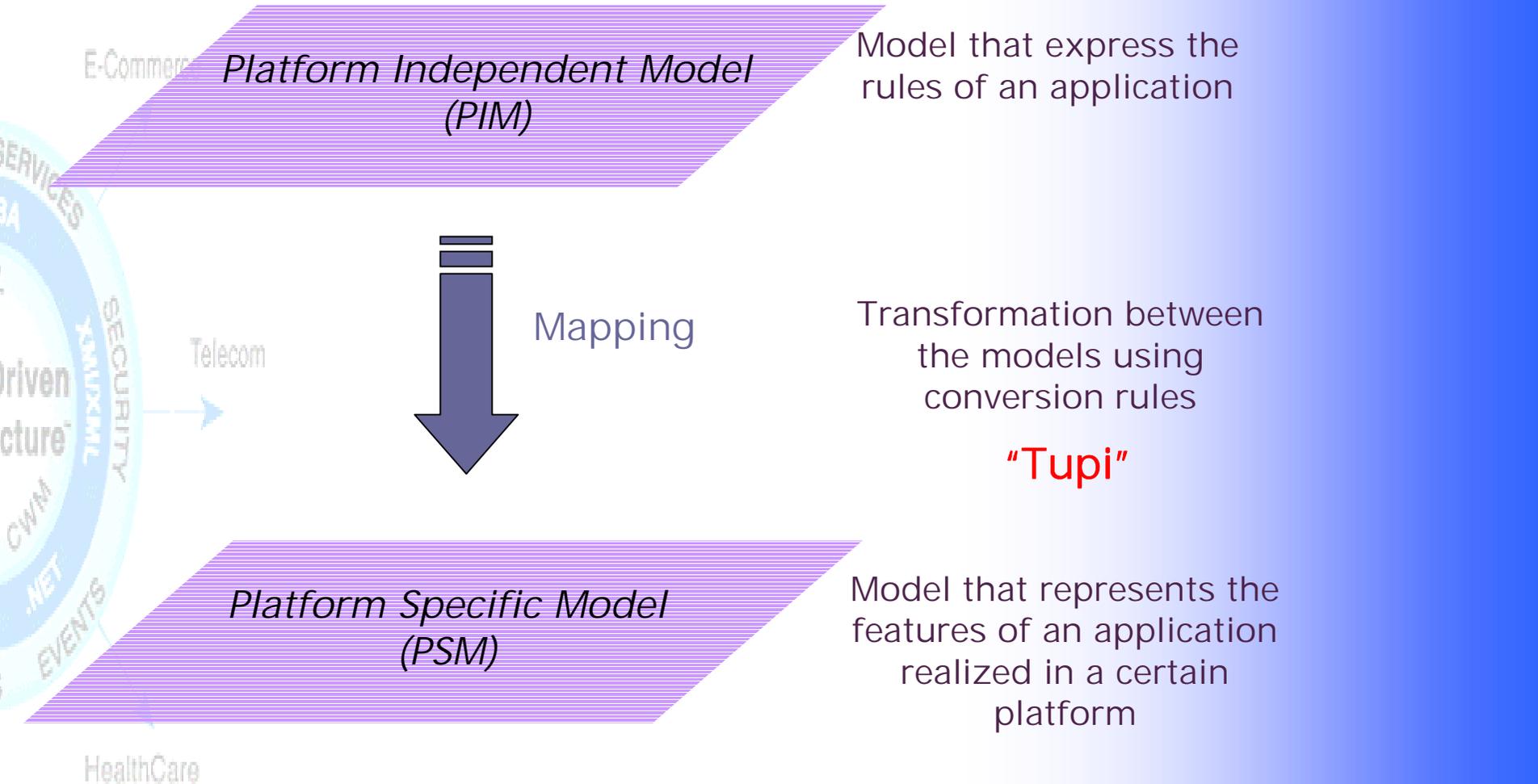
Model that express the rules of an application

Mapping

Transformation between the models using conversion rules

"Tupi"

Platform Specific Model (PSM)

Model that represents the features of an application realized in a certain platform

PIM

UML *profile* for EDOC

- Component Integration
- Description of process, application structure, component composition

|		Descrição UML
ProcessComponent	component	class
Property Definition	component properties	class related with the component
Port	port	----
FlowPort	port that sends only one flow	class related with the class of the component
MultiPort	port that glue a set of Flow Port	classe related with the class of the component with Flow Port
OperationPort	port that represents a call/return of an operation	operation in the class of the component.
ProtocolPort	port connected to a protocol	classe related with the class of the component
Protocol	communication protocol	class related with the class of the protocol ports
InitiatingRole / RespondingRole	role of a component in a protocol	class

UFRN – CCET – DIMAp

PSM

CORBA

- Interoperability in distributed and heterogeneous environments

- Interaction between objects via ORB

- Separation between object interface and object implementation (IDL – *Interface Definition Language*)

interface	object description
attribute	attribute
operation	operation
module	set of elements
struct, enum	data type
typedef	named typed
exception	exception

PIM-PSM Mapping
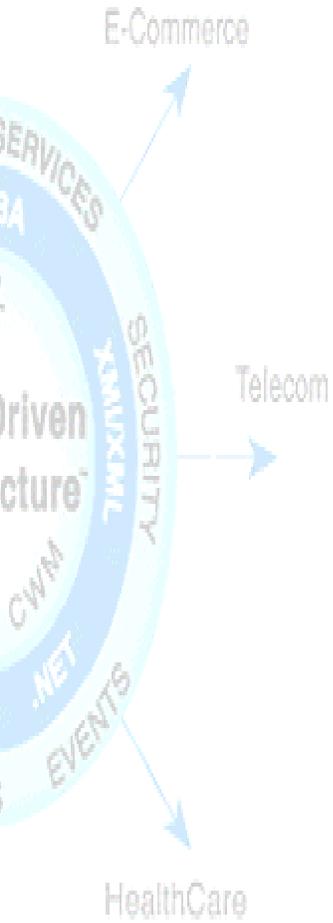
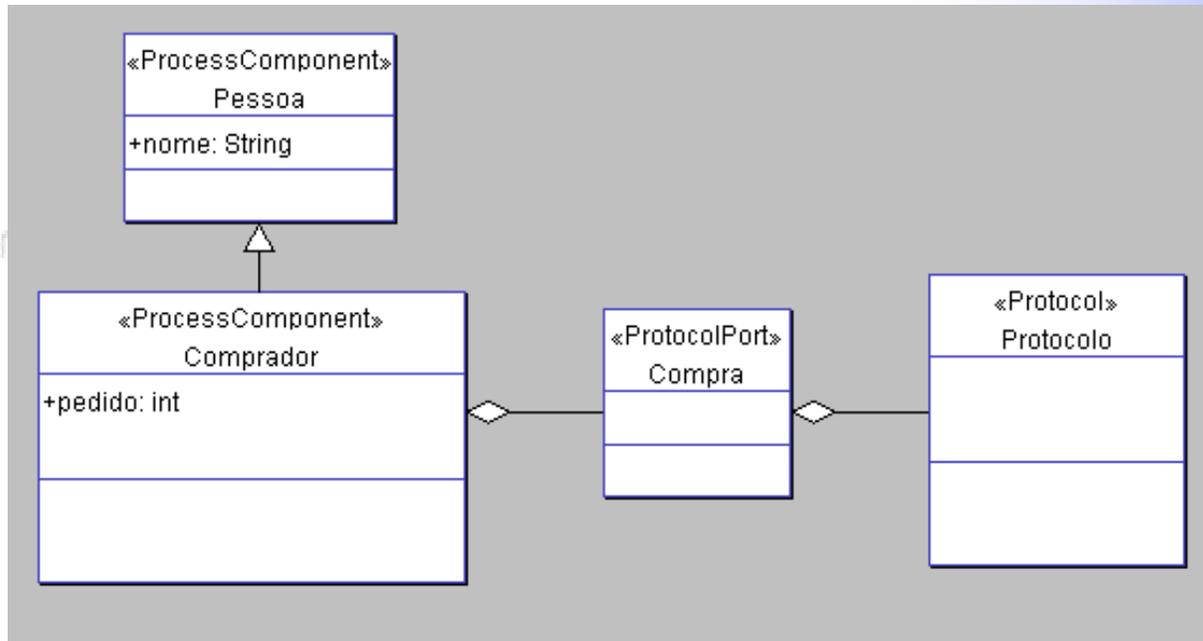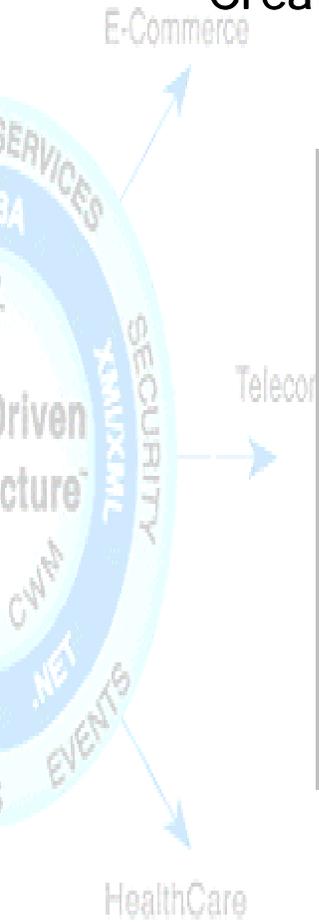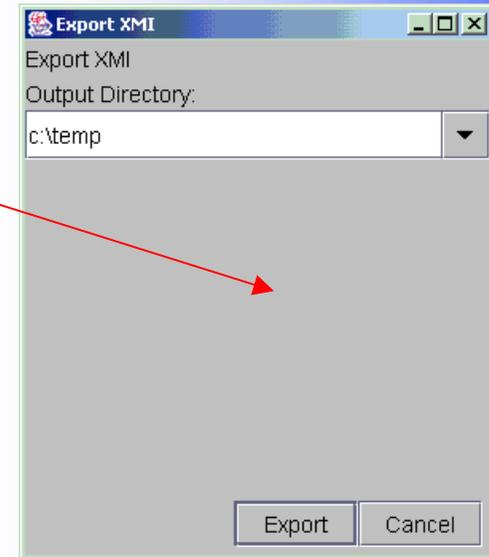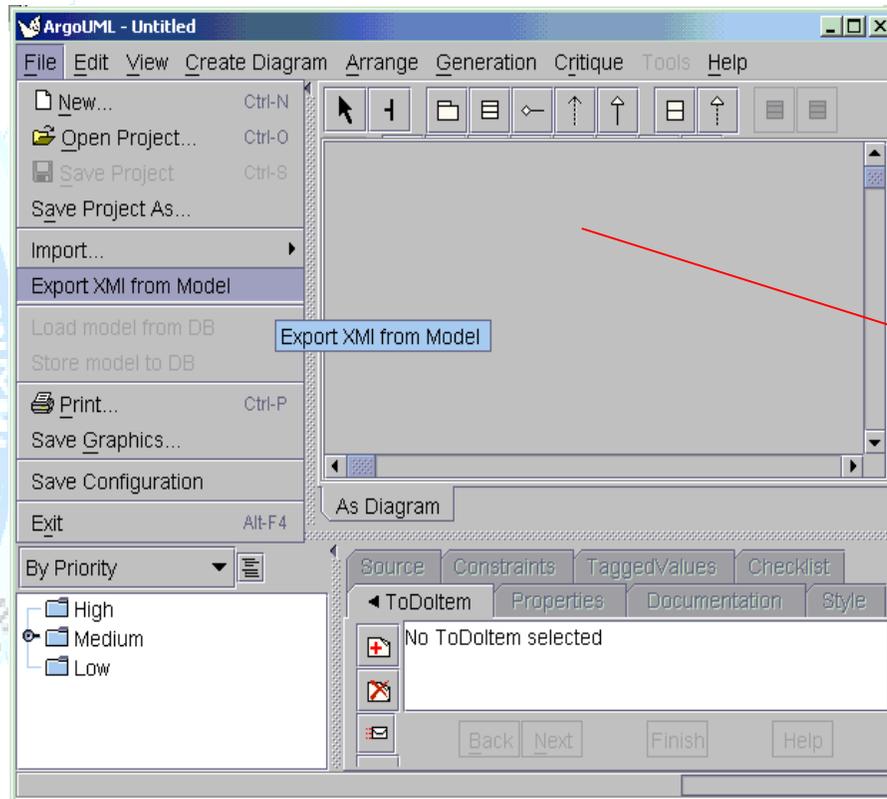|	
ProcessComponent	*interface with an* attribute for each property and for each port
Port	----
FlowPort	*interface* with a method that indicates the direction of the port
MultiPort	*interface* with an attribute for each subport
OperationPort	operation in the interface of the component
Protocol	*interface*
ProtocolPort	*interface* with an attribute that represents the protocol
InitiatingRole / RespondingRole	class with an attribute to each protocol port

Tupi

Creates PIM → **Metamodel** → **Converstion into IDL**

- ArgoUML
- Syntax follows the UML *profile* for EDOC

- ArgoUML
- Produces the XMI file

- Applying the conversion rules
- Produces the IDL interfaces

Conversion PIM-IDL

• Uses an JAXP API (allows to process the XML using DOM, SAX and XSLT)

Conversion

PSM Model (IDL File)

PIM Metamodel (in XMI)

•XML Document (well-formed)

•Represents a textual form of an UML model

Conversion Rules (in XSLT)

•XML Document (well-formed)

•Represents actions to be executed when a certain entity is found in the metamodel

UFRN – CCET – DIMAp

Example[1]

Creating the PIM model

Example²

Producing the PIM Metamodel

UFRN – CCET – DIMAp

Example[3]

PIM Metamodel

```
<Foundation.Core.Class xmi.id="xmi.9" xmi.uuid="10-3--128--9-78efd8:f2ae1e7096:-
7ffc">

    <Foundation.Core.ModelElement.name>Comprador

    </Foundation.Core.ModelElement.name>


    <Foundation.Core.Attribute xmi.id="xmi.14">

        <Foundation.Core.ModelElement.name>pedido

        </Foundation.Core.ModelElement.name>

        <Foundation.Core.StructuralFeature.type>

        <Foundation.Core.Classifier xmi.idref="xmi.16"/>

        </Foundation.Core.StructuralFeature.type>
```

Example[4]

Conversion of the metamodel into IDL

Conclusions

Difficulties:

- References about MDA.

- The lack of tools that implements MDA.

- The *gap* between PIM syntax and IDL syntax

Future Works

- An extension to TUPI to convert PSM in IDL of CORBA CCM *(CORBA Component Model)*