

Performance Analysis Repository for RTES Designs

Référentiel d'analyses pour la conception de système embarqué temps réel

Présentateur : Thanh-Dat NGUYEN

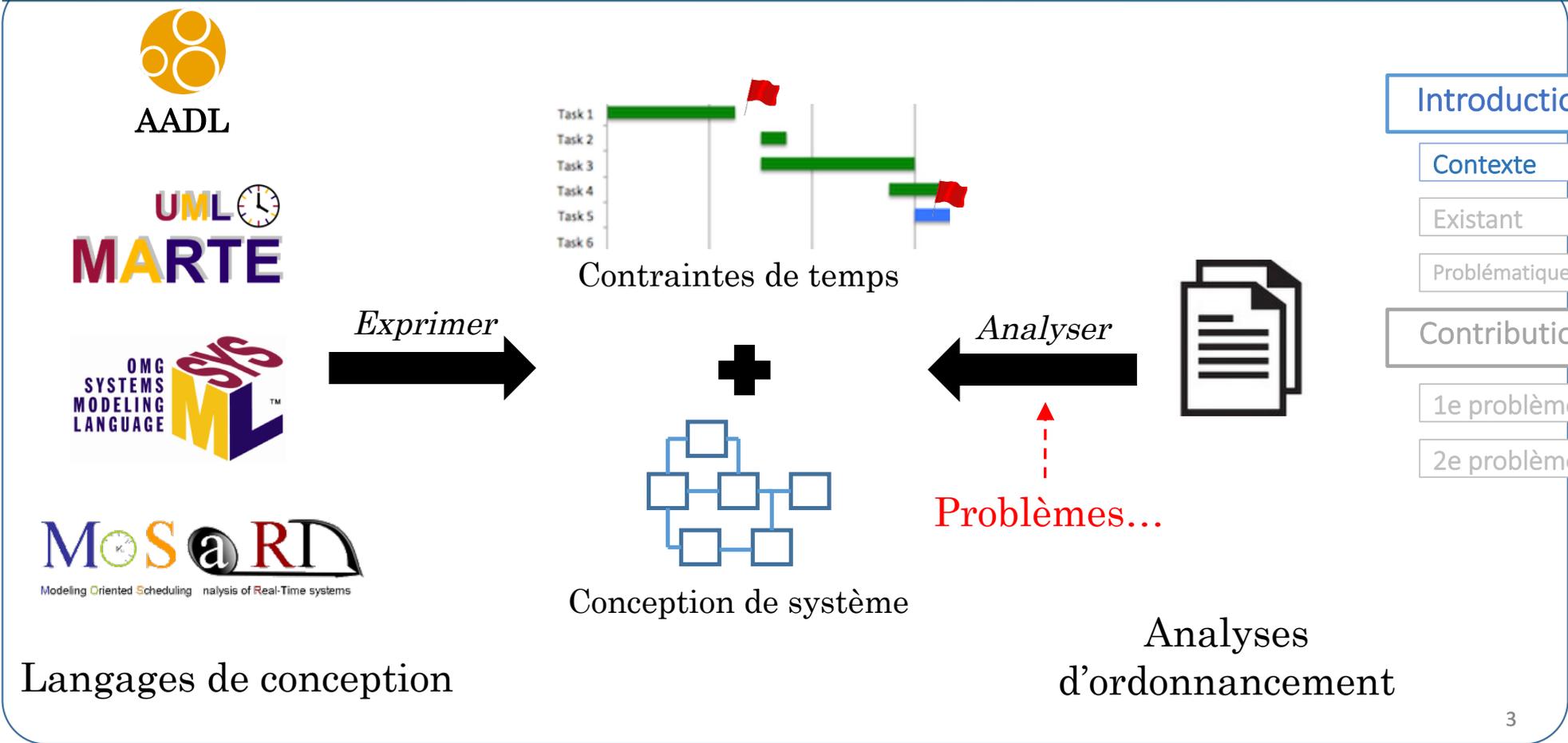
Octobre 2017

1

CONTENU

- Introduction
 - Contexte du travail*
 - Existant : MoSaRT Analysis Repository*
 - Problématiques*
- Contribution
 - Premier problème*
 - Deuxième problème*

INTRODUCTION



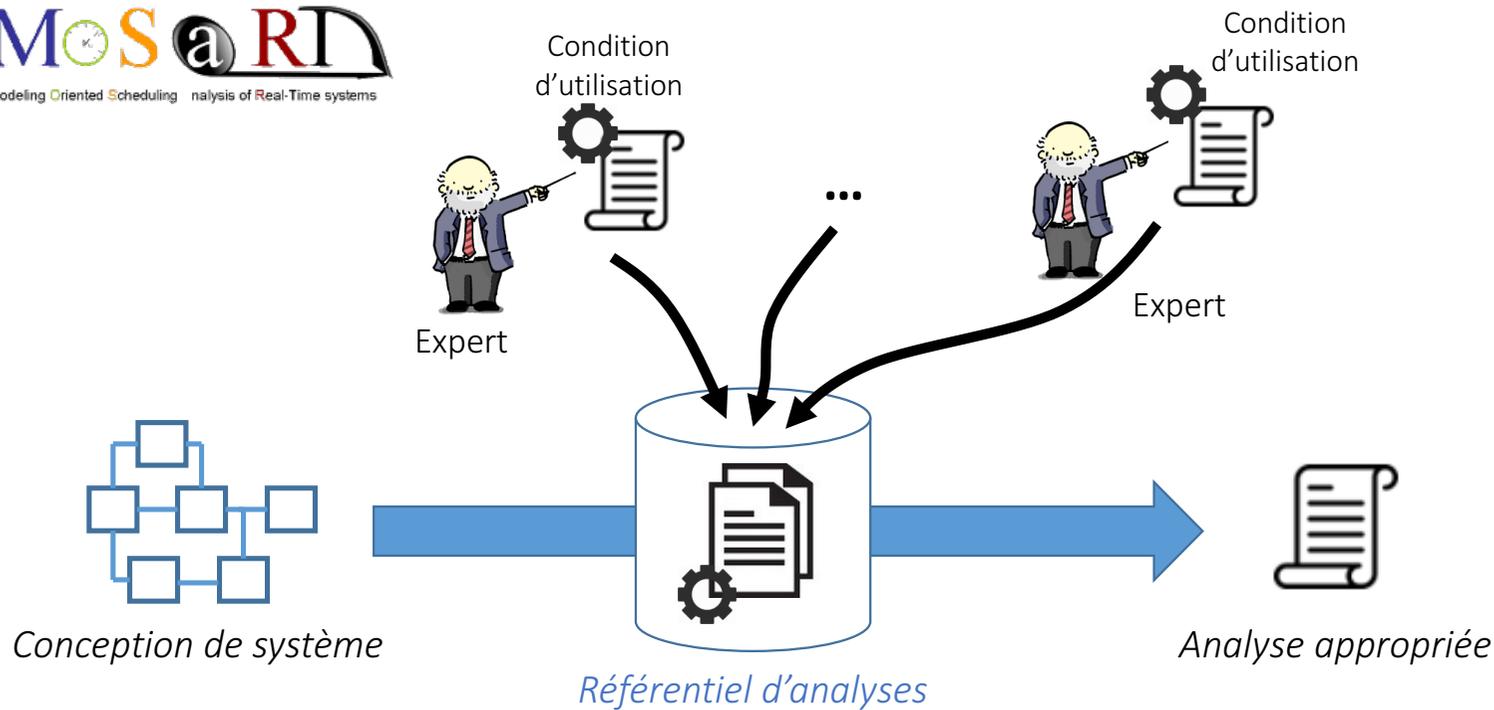
OBJECTIFS DU TRAVAIL

- Proposer une démarche qui aide à prendre de décisions en termes de choix d'analyse pour une conception de systèmes



- Introduction
- Contexte
- Existant
- Problématiques
- Contribution
- 1e problème
- 2e problème

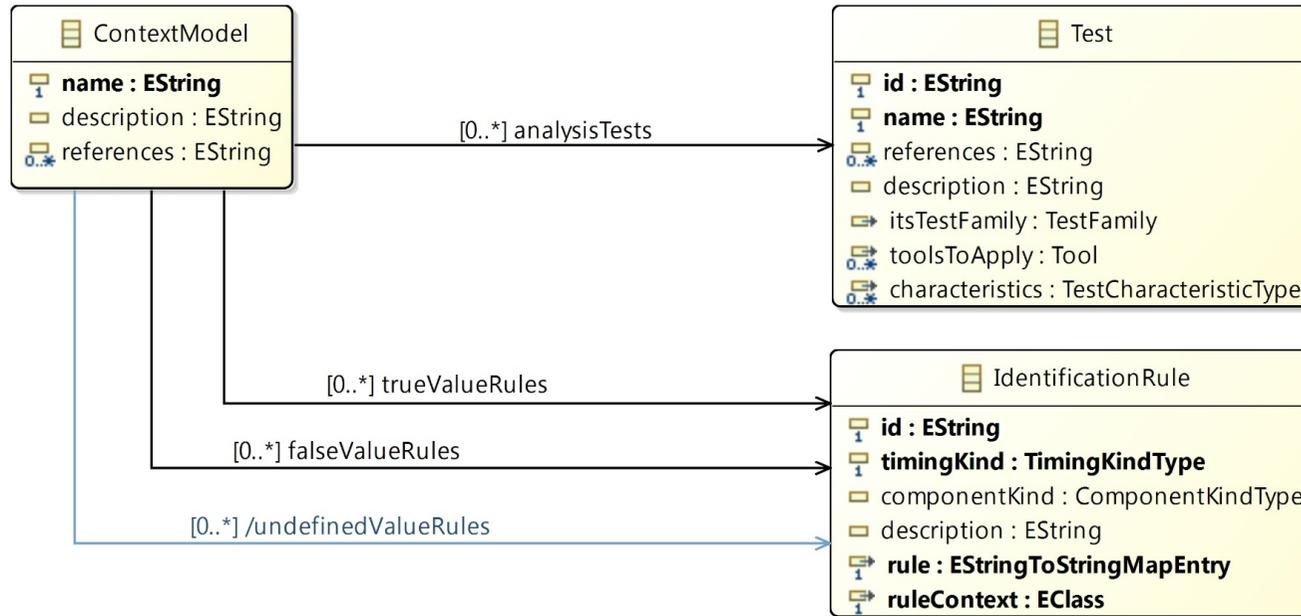
APPROCHE EXISTANTE



- Introduction
- Contexte
- Existant
- Problématique
- 5
- Contribution
- 1e problème
- 2e problème

➔ LIAS : MoSaRT Analysis Repository [Thèse, Yassine OUHAMMOU]

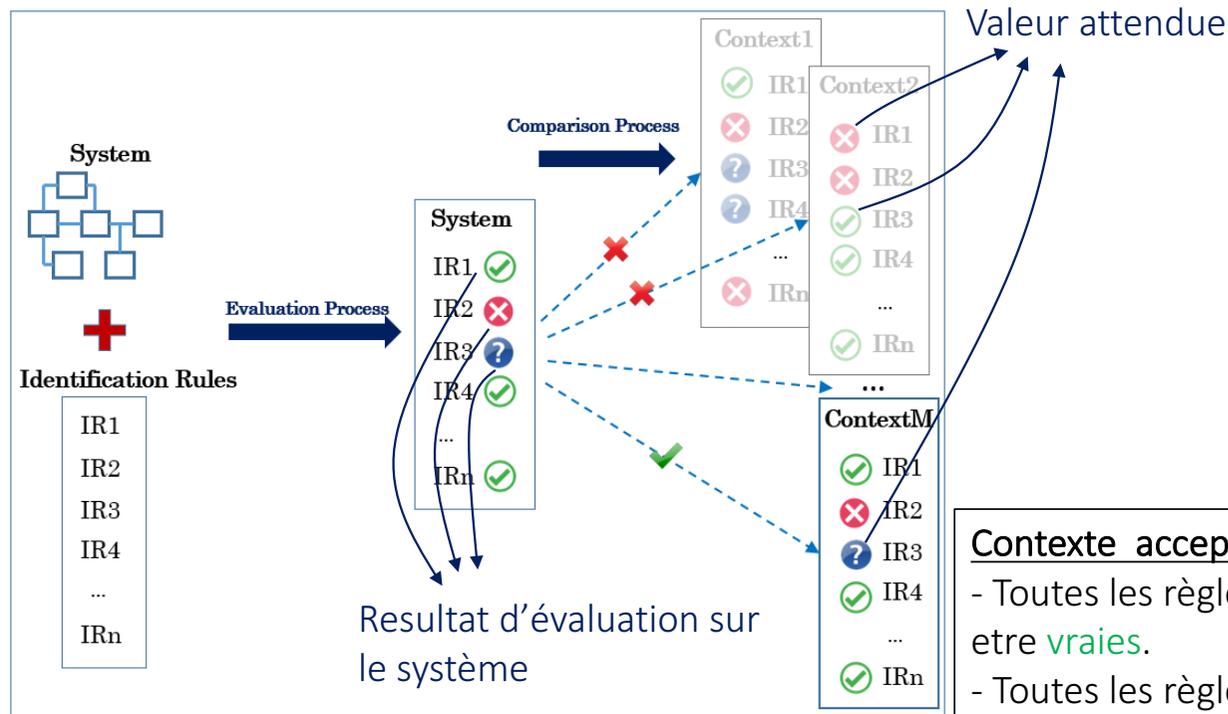
ARCHITECTURE GENERALE DE MoSaRT



- Introduction
- Contexte
- Existant
- Problématique
- Contribution
- 1e problème
- 2e problème

➔ LIAS : MoSaRT Analysis Repository...

PROCESSUS D'IDENTIFICATION



- Introduction
- Contexte
- Existant
- Problématique
- Contribution
- 1e problème
- 2e problème

Contexte accepté:

- Toutes les règles prévue vraies doivent être vraies.
- Toutes les règles prévue fausses doivent être fausses.

➔ LIAS : MoSaRT Analysis Repository...

PROBLEMATIQUES

PROBLEME 1 :

Toutes les tâches sont périodiques

Concepts de *MoSaRT*
Design Language

Regles exprimees en OCL

```
SoftwarePlatform::SoftwareBehavior:SbTaskActivity allInstances()
->forAll(taskActivity|taskActivity.trigger.ocIsTypeOf(SoftwarePlatform::SoftwareBehavior:SbTimeTrigger)
and taskActivity.trigger.ocIsType(SoftwarePlatform::SoftwareBehavior:SbTimeTrigger).periodicity
.ocIsTypeOf(RealTimeProperties::RtpTypes:RtpPeriodicType)
```

(1). Règle d'identification dépend du langage de conception et le référentiel d'analyses *dépend* donc du langage de conception.

→ Limiter l'utilisabilité du référentiel d'analyses.

Introduction

Contexte

Existant

Problématique

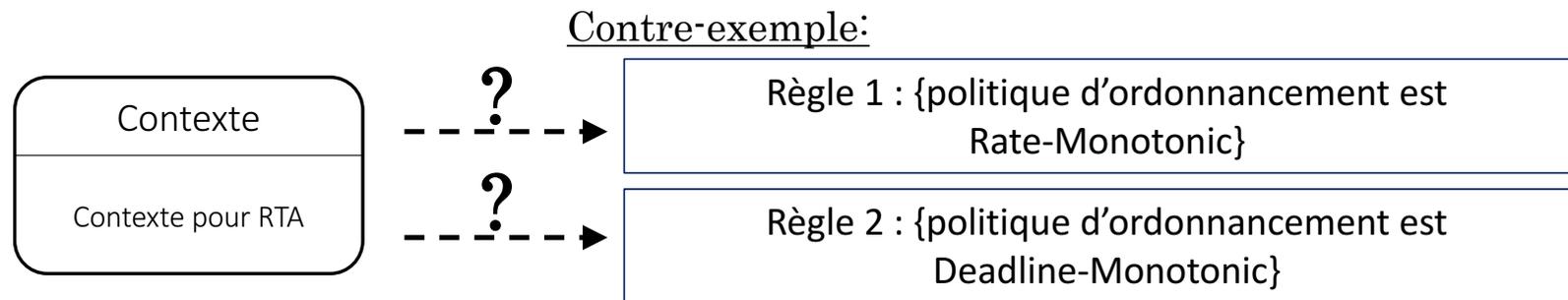
Contribution

1e problème

2e problème

PROBLEMATIQUES

PROBLEME 2 : Expressivité des règles d'identification par rapport au contexte



Pour le test RTA, les deux politiques d'ordonnancement (RM, DM) sont acceptables.
 Cependant, *on ne peut pas mettre la valeur attendue de 2 règles est vraie*

Introduction

Contexte

Existant

Problématique

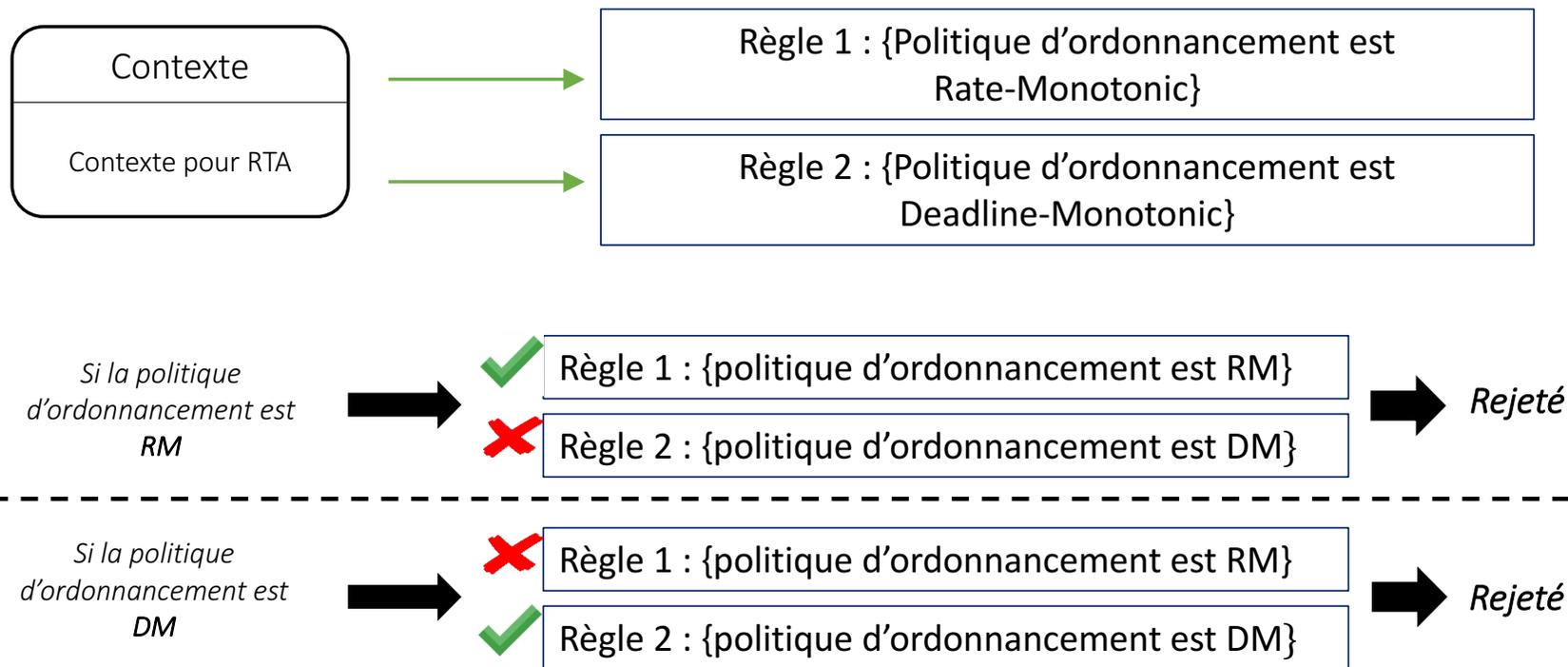
^s Contribution

1e problème

2e problème

PROBLEMATIQUES

PROBLEME 2 : Expressivité des règles d'identification par rapport au contexte



Dans les 2 cas, le contexte devrait être accepté, cependant l'exemple montre qu'il n'est pas accepté

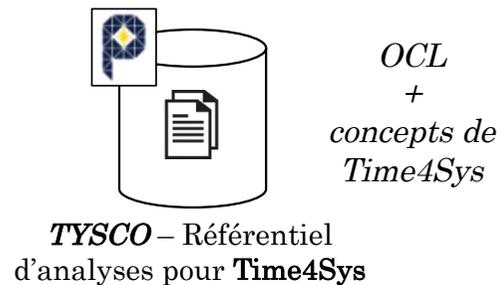
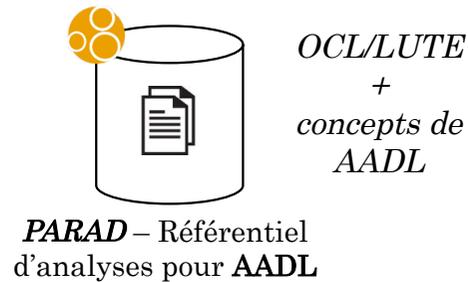
- Introduction
- Contexte
- Existant
- Problématique
- Contribution
- 1e problème
- 2e problème

→ trueValueRule
→ falseValueRule

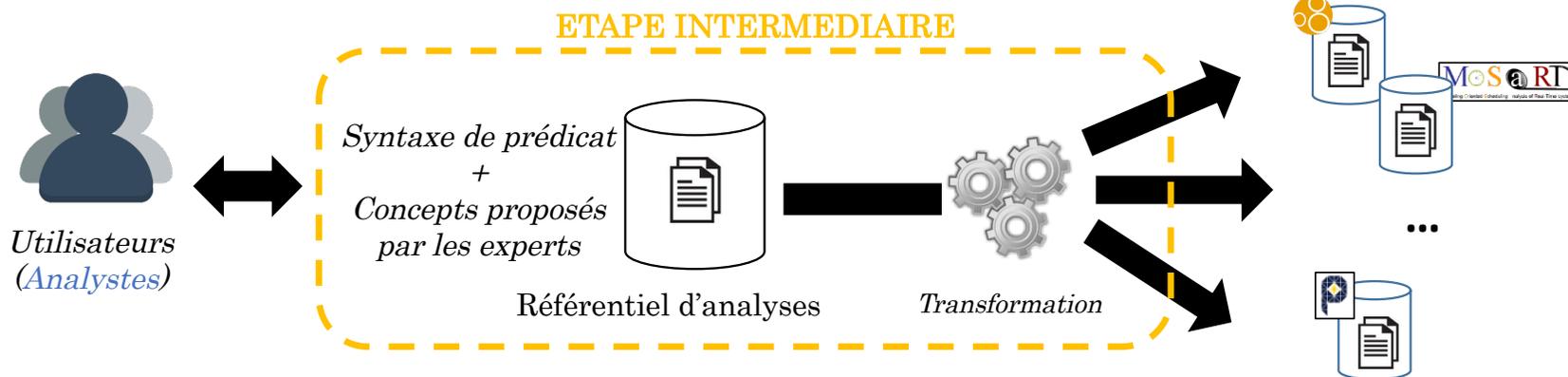
CONTRIBUTION

PROBLEME 1 : Dépendance du référentiel d'analyses par rapport au langage de conception

- Première approche : Construire pour chaque langage de conception un référentiel d'analyses



- Deuxième approche : Langage intermédiaire



- Introduction
- Contexte
- Existant
- Problématique
- Contribution**
- 1e problème
- 2e problème

CONTRIBUTION

PROBLEME 1 : Dépendance du référentiel d'analyses par rapport au langage de conception

- **Première approche** : Construire pour chaque langage de conception un référentiel d'analyses
- **Deuxième approche** : Langage intermédiaire

Syntaxe déclarative

- Simple
- Chercher à décrire quel est le problème

Exemple :

$$J = \{2k+1 \mid k \in \mathbb{N} \wedge 0 \leq k \leq 10\}$$

- Moins expressive

VS

Syntaxe impérative

- Plus complexe
- Chercher à décrire comment résoudre le problème

Exemple :

```
J = {}
k := 0
Tantque k ≤ 10
    J := J U {2k+1}
    k := k + 1
Fin tantque
```

- Plus expressive
- Exemple: OCL

Introduction

Contexte

Existant

Problématique

Contribution

1e problème

2e problème

➔ *la syntaxe déclarative est meilleure pour exprimer les règles d'identification*

CONTRIBUTION : Langage de predicat

```

example.ir
RulesSet
Theory : "liu&layland"
{
  IdentificationRule "Periodic tasks":
  {
    CONTENT: All "liu&layland.task" t | (GetProperty(t, "liu&layland.periodicity" = "liu&layland.periodicity.periodic));
  }
  IdentificationRule "Harmonic tasks":
  {
    CONTENT: All "liu&layland.task" t1, Some "liu&layland.task" t2, Some k in NAT |
      (GetProperty(t1, "liu&layland.period" = (k * GetProperty(t2, "liu&layland.period"))));
  }
}

liu_layland.theory
Theory Concepts Package 'liu&layland' is
Component Types:
- task
- system
- processor
- process
Property Types:
- periodicity applied for (task), concrete values: (periodic,aperiodic,sporadic)
- wcet applied for (task), is numeric value
- deadline applied for (task), is numeric value
- offset applied for (task), is numeric value
- period applied for (task), is numeric value
- scheduling_policy applied for (processor), concrete values: (priority_fixed,EDF,DM,RM)
        
```

Règles d'identification

Concepts

- Introduction
- Contexte
- Existant
- Problématique
- Contribution**
- 1e problème
- 2e problème

CONTRIBUTION : Transformation

→ Langage de prédicat :

Transformation

Limitation courante : Transformation de règles dont le quantificateur est ***pour tous*** (\forall)

Passage à l'échelle

- Automatiquement détecter des doublons

- Ensemble de concepts extensible

Introduction

Contexte

Existant

Problématique

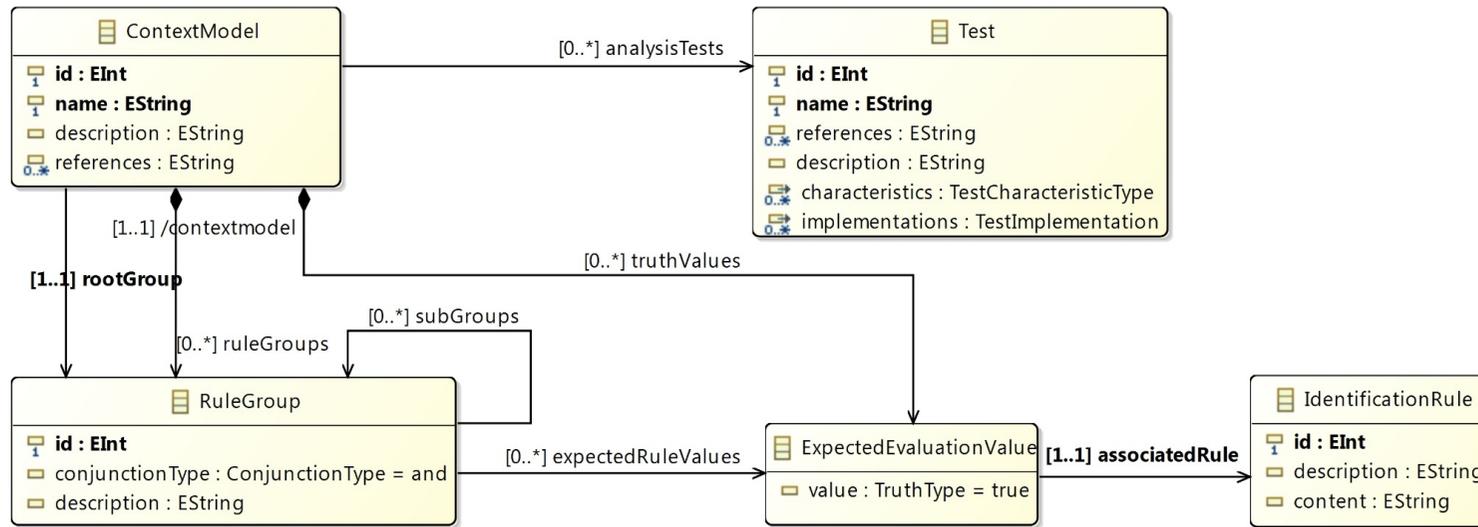
Contribution

1e problème

2e problème

CONTRIBUTION : Groupe de règles d'identification

➔ **Problème** : Expressivité du référentiel d'analyses



Valeur attendue est attachée directement à chaque règle d'identification

➔ Extrait du méta-modèle du *Référentiel d'Analyses*

- Introduction
- Contexte
- Existant
- Problématique
- S
Contribution
- 1e problème
- 2e problème

CONTRIBUTION : Groupe de règles d'identification

→ Problème : Expressivité du référentiel d'analyses

Résoudre le problème du contre-exemple



Introduction

Contexte

Existant

Problématique

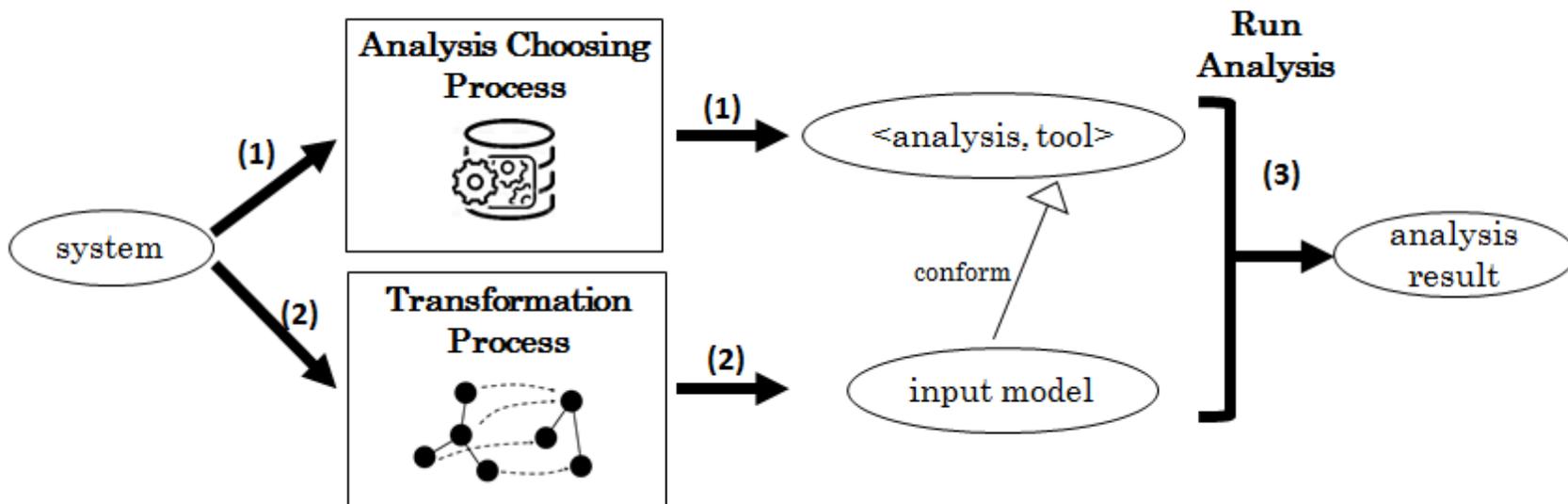
Contribution

1e problème

2e problème

DEMO

Processus global





École Nationale Supérieure de Mécanique et d'Aérotechnique / Université de Poitiers

Laboratoire d'Informatique et d'Automatique pour les Systèmes (<http://www.lias-lab.fr>)

ENSMA - Téléport 2 - 1 Avenue Clément Ader - BP 40109 - 86961 FUTUROSCOPE CHASSENEUIL Cedex - France



Merci pour votre présence !