

Université de la Manouba
Ecole Nationale des Sciences de l'Informatique



Projet de Fin d'Etudes

**Une approche dirigée par les modèles pour la
vérification formelle des systèmes embarqués**

LAAS-CNRS

Présenté par :

Mohamed Amine Aouadhi

Encadré par : Dr. Pierre-Emmanuel Hladik

Supervisé par : Pr. Henda Hajjami Ben Ghezala

Année universitaire : 2013 - 2014



Plan de la présentation

- ❖ Contexte Générale
- ❖ Etat de l'art
- ❖ Approche
- ❖ Analyse et Spécification des besoins
- ❖ Conception
- ❖ Réalisation
- ❖ Conclusion et Perspectives

Plan de la présentation

❖ **Contexte Générale**

❖ Etat de l'art

❖ Approche

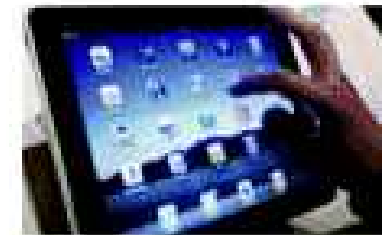
❖ Analyse et spécification des besoins

❖ Conception

❖ Réalisation

❖ Conclusion et Perspectives

Introduction



Méthodologies de modélisation des systèmes Embarqués



Vérification Formelle ???

Méthodologies de modélisation des systèmes Embarqués



Problématique

- ❖ Application des techniques de vérification formelle sur les modèles des systèmes embarqués



Plan de la présentation

- ❖ Contexte Générale

- ❖ **Etat de l'art**

- ❖ Approche

- ❖ Analyse et spécification des besoins

- ❖ Conception

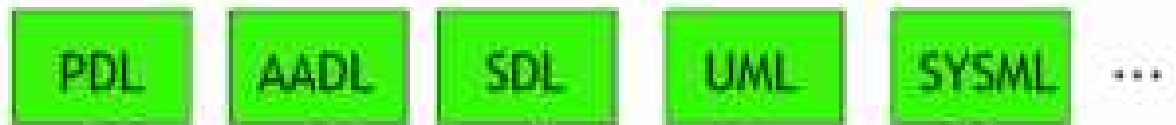
- ❖ Réalisation

- ❖ Conclusion et Perspectives

Etat de l'art

(Meta)-modeleur

Langages de
Modélisation



Editeurs

Transformation de modèles

Langages Formels

Moteurs de
transformation
(ATL, KERMETA, ...)

Compilation

Compilateurs



Outils de vérification

Etat de l'art

Nous citons quelques travaux :

- ❖ Transformation de AADL vers Fiacre
- ❖ Transformation de Uml vers Fiacre
- ❖ Transformation de AADL vers BIP
- ❖ Transformation de SystemC vers UPPAAL

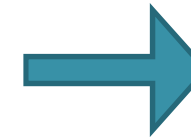
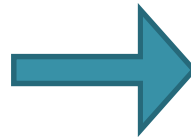
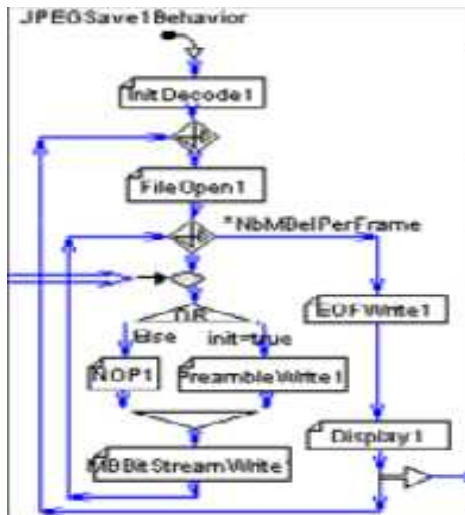
Etat de l'art

Notre travail se situe dans cette approche:

MCSE

Fiacre

TINA



Vérification
avec TINA

- ✓ Adoptée par Intel
- ✓ Description semi-formelle
- ✓ Expression des modèles

- ✓ Langage formel
- ✓ Description formelle

- ✓ Environnement logiciel
- ✓ Edition et analyse des réseaux de Petri
- ✓ Vérification formelle

Plan de la présentation

- ❖ Contexte Générale

- ❖ Etat de l'art

- ❖ **Approche**

- ❖ Analyse et spécification des besoins

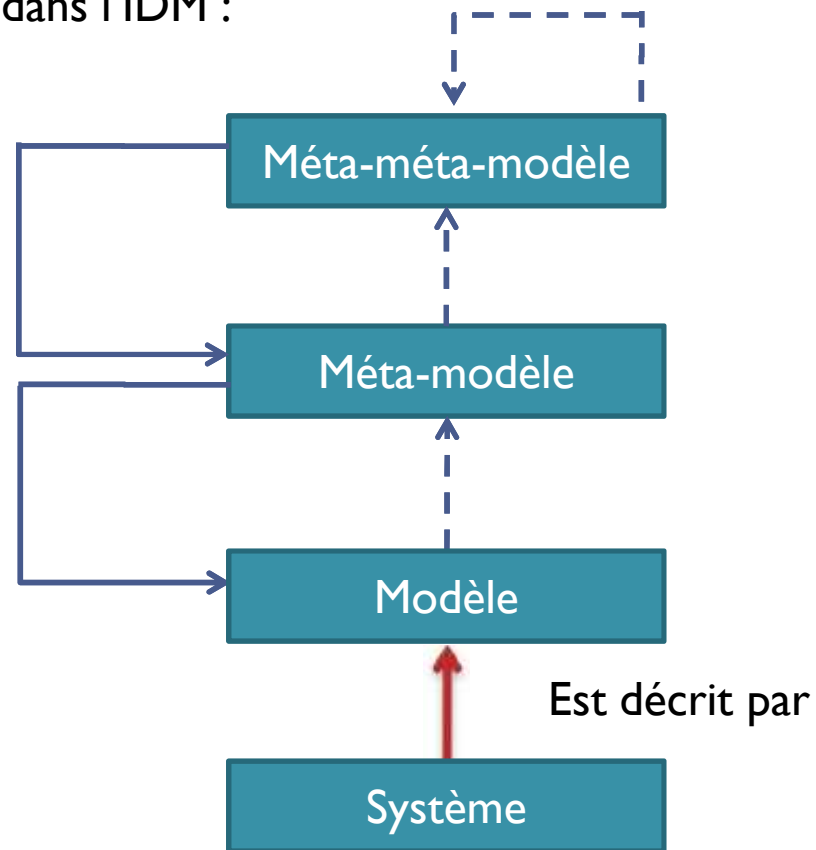
- ❖ Conception

- ❖ Réalisation

- ❖ Conclusion et Perspectives

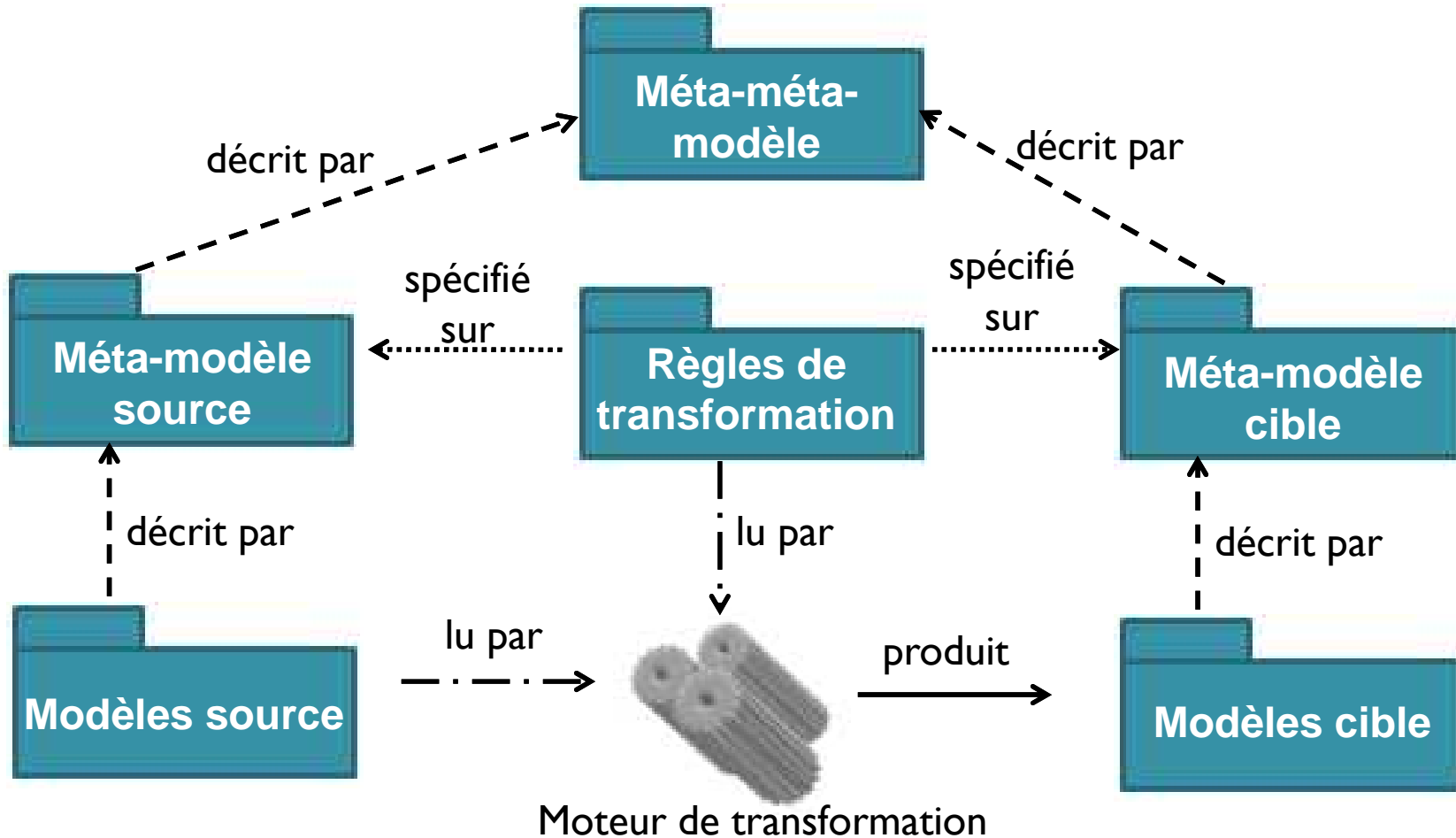
Ingénierie dirigée par les modèles

Niveaux d'abstraction dans l'IDM :



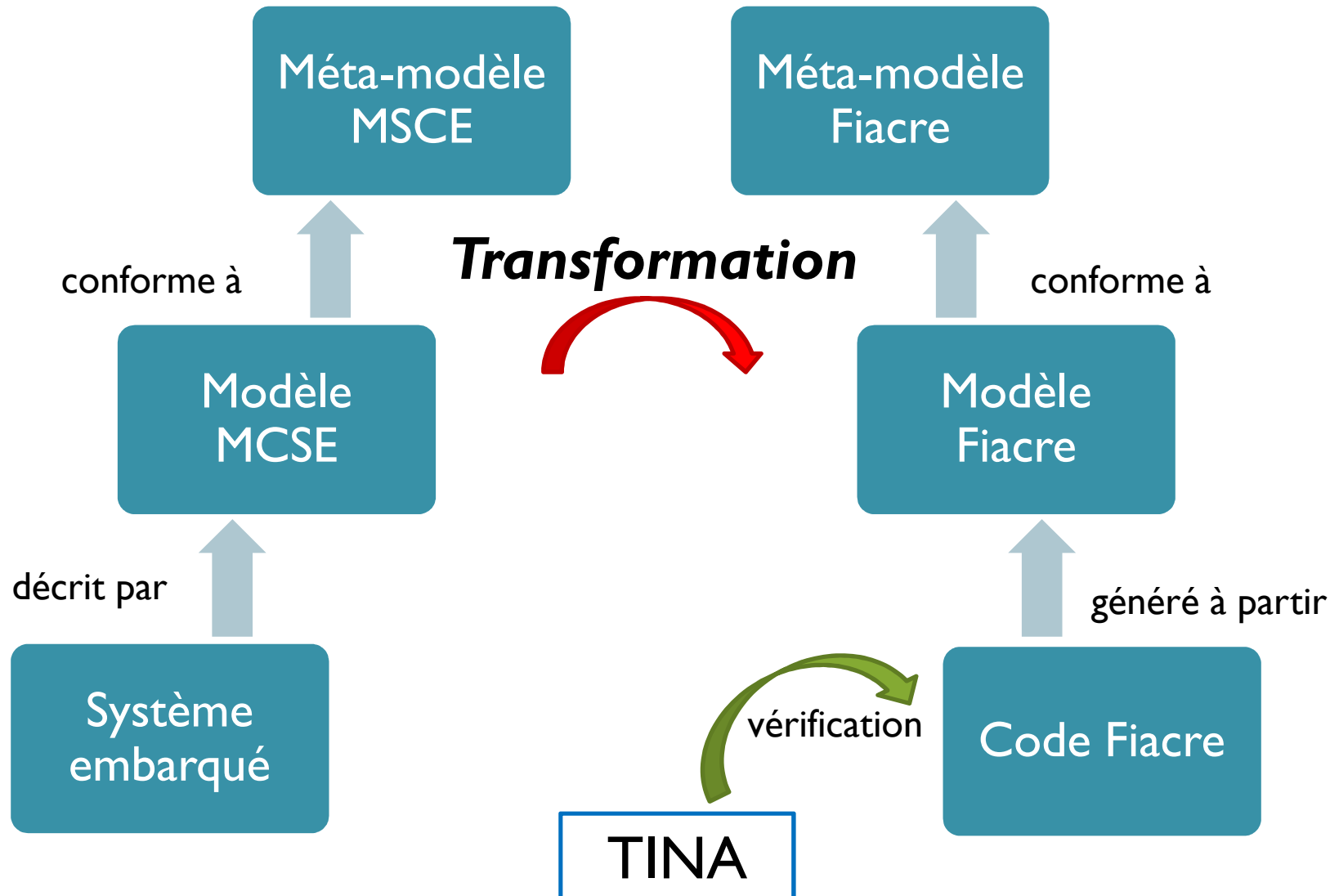
- - - - -> Est conforme à / est instance de
- > Est un langage de spécification

Ingénierie dirigée par les modèles



Principe d'une transformation de modèles

Approche





Plan de la présentation

- ❖ Contexte Générale
- ❖ Etat de l'art
- ❖ Approche
- ❖ **Analyse et Spécification des besoins**
- ❖ Conception
- ❖ Réalisation
- ❖ Conclusion et Perspectives

Objectif

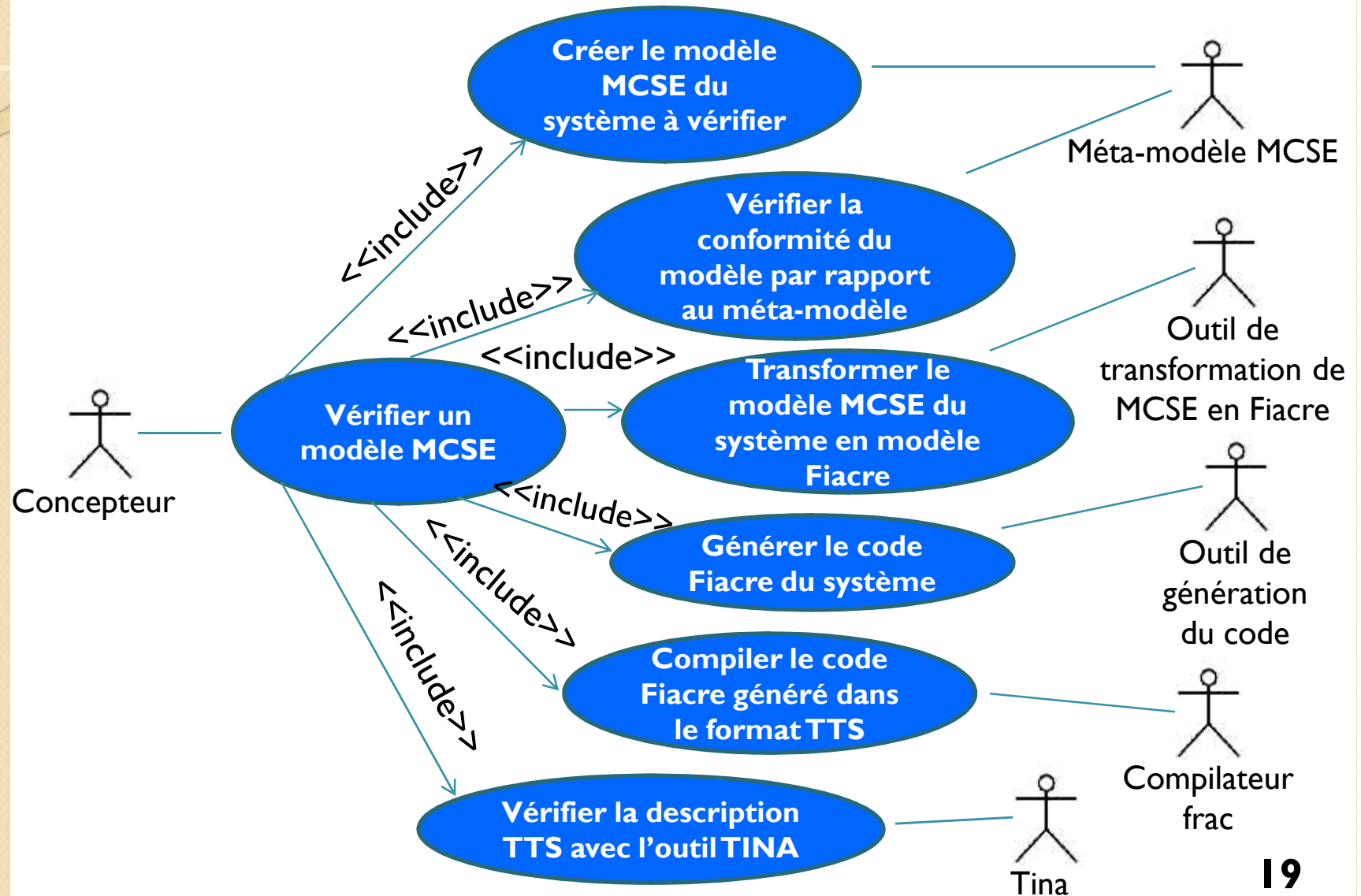
- ❖ Mettre en place une solution permettant l'application des techniques de vérification formelle sur les modèles MCSE des systèmes embarqués

Outils développés

Les outils développés sont:

- ❖ Un méta-modèle de MCSE
- ❖ Un méta-modèle de Fiacre
- ❖ Un outil de transformation de modèle MCSE en modèle Fiacre
- ❖ Un outil de génération du code Fiacre

Diagramme des cas d'utilisation

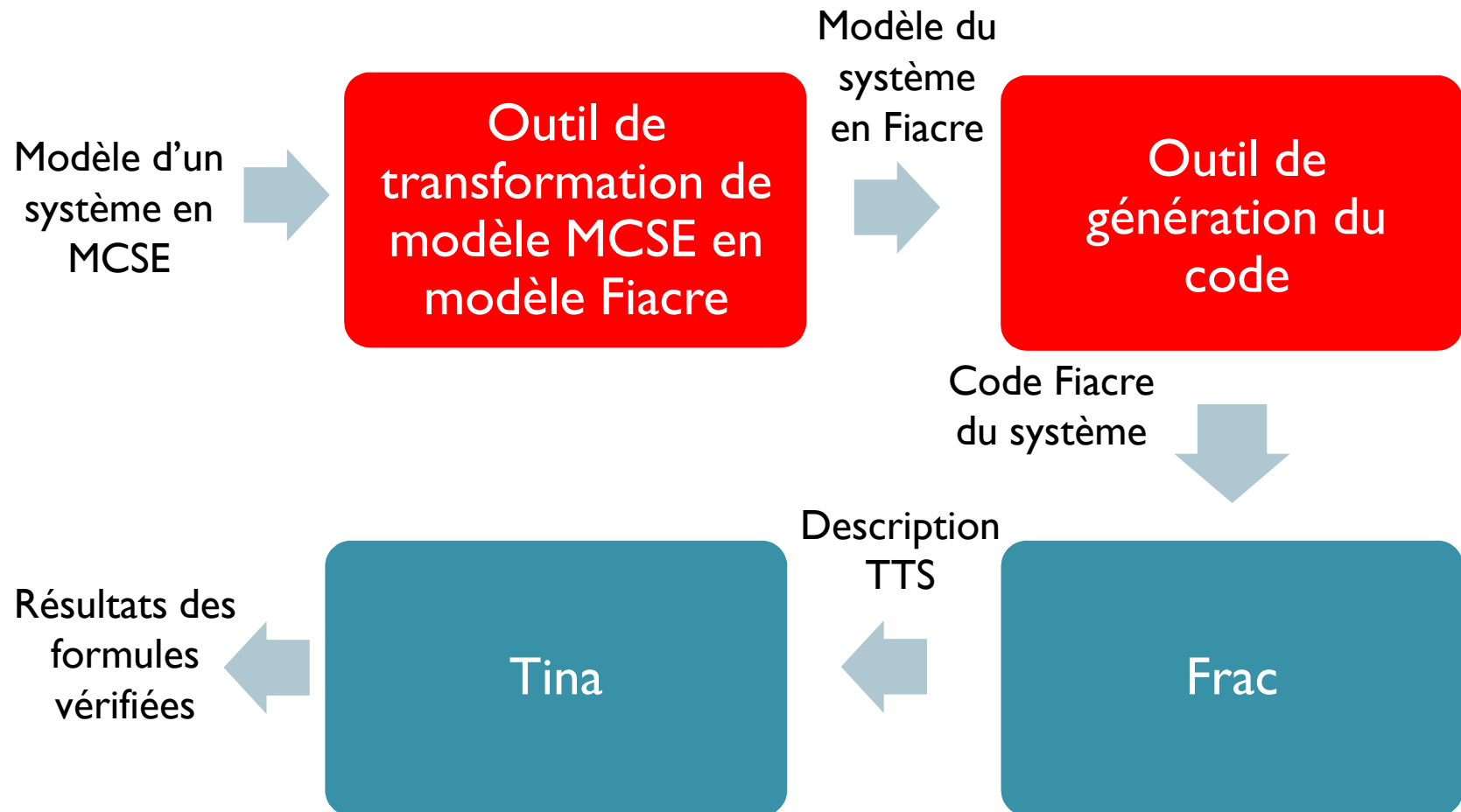




Plan de la présentation

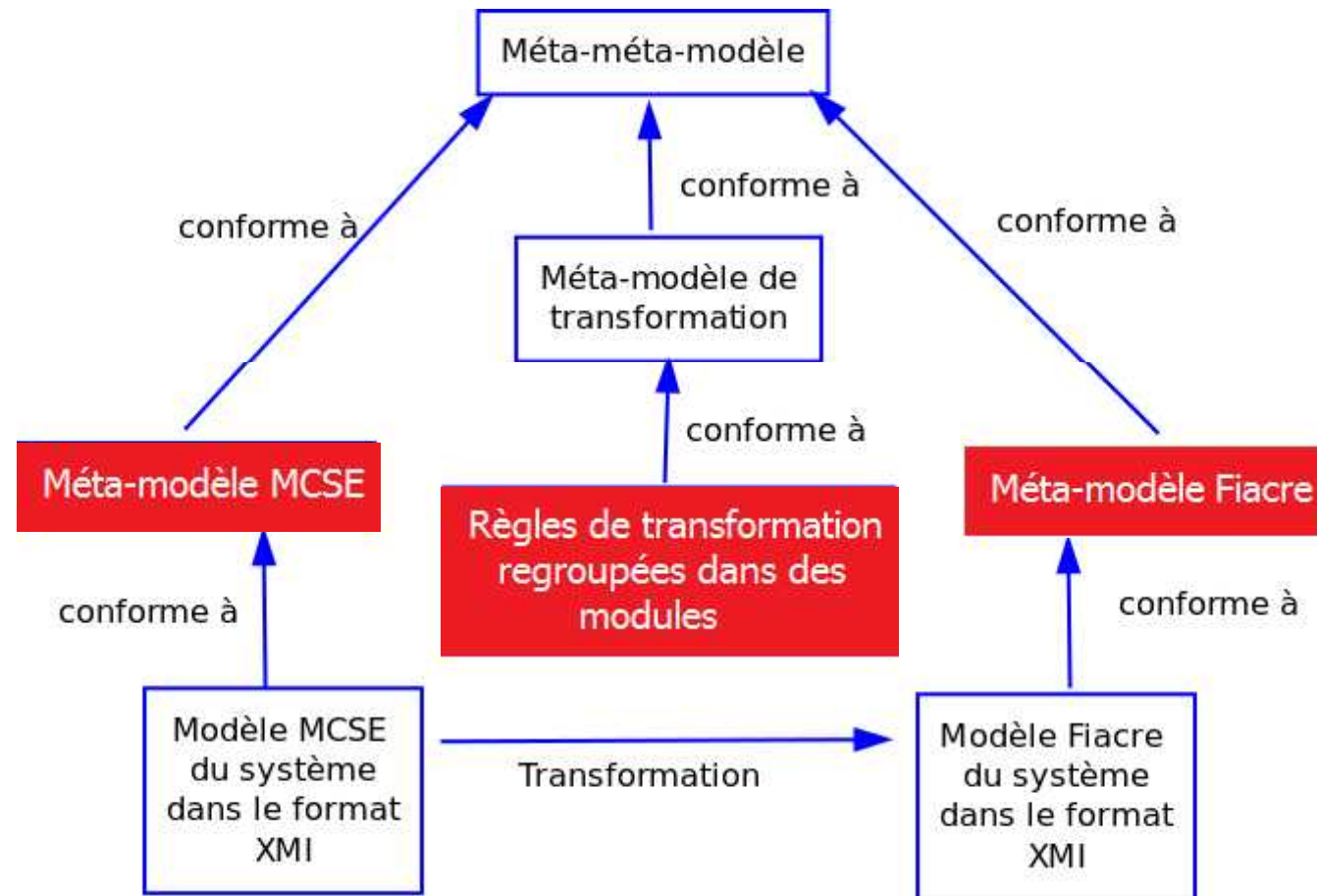
- ❖ Contexte Générale
- ❖ Etat de l'art
- ❖ Approche
- ❖ Analyse et Spécification des besoins
- ❖ **Conception**
- ❖ Réalisation
- ❖ Conclusion et Perspectives

Architecture générale

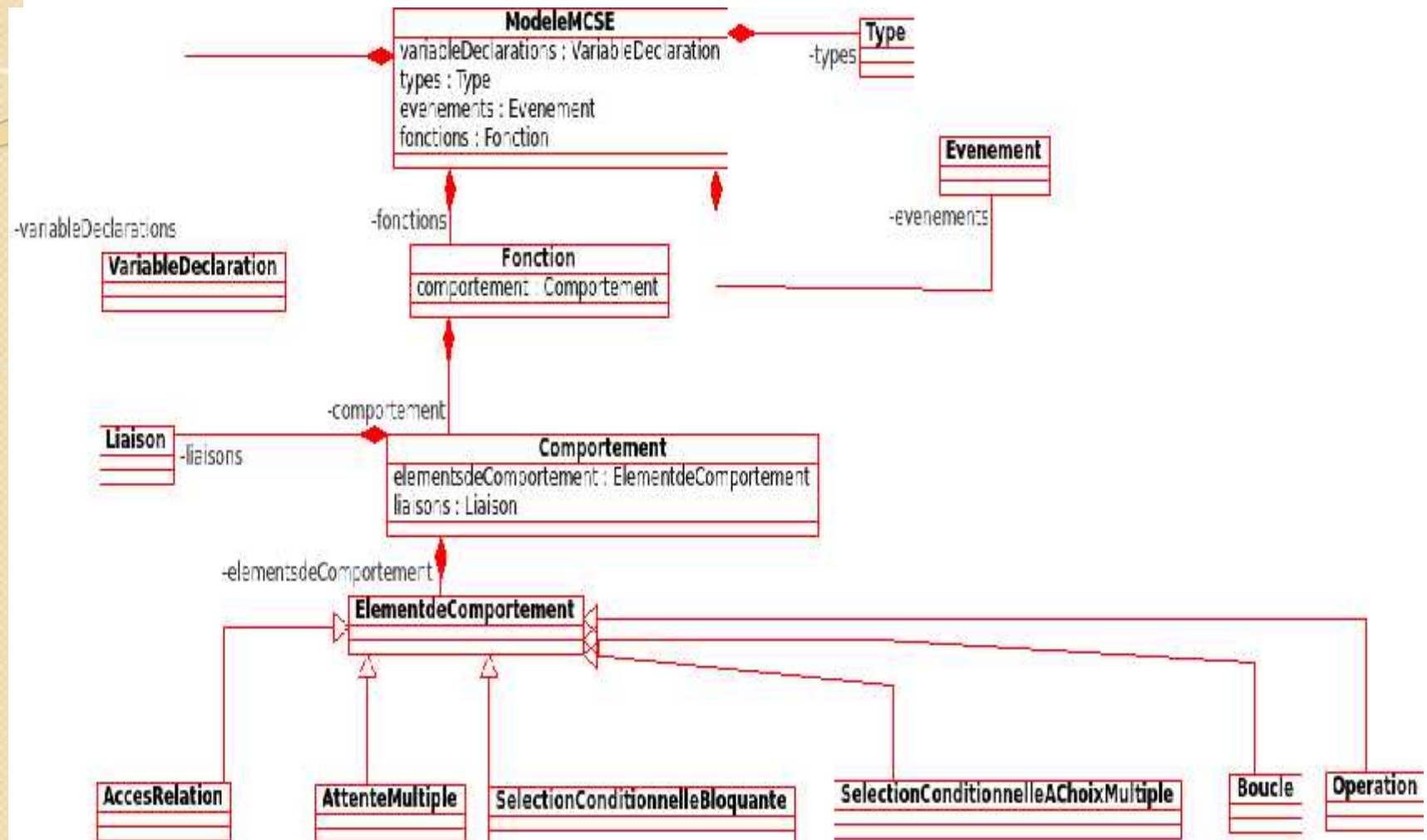


Architecture générale du processus de vérification

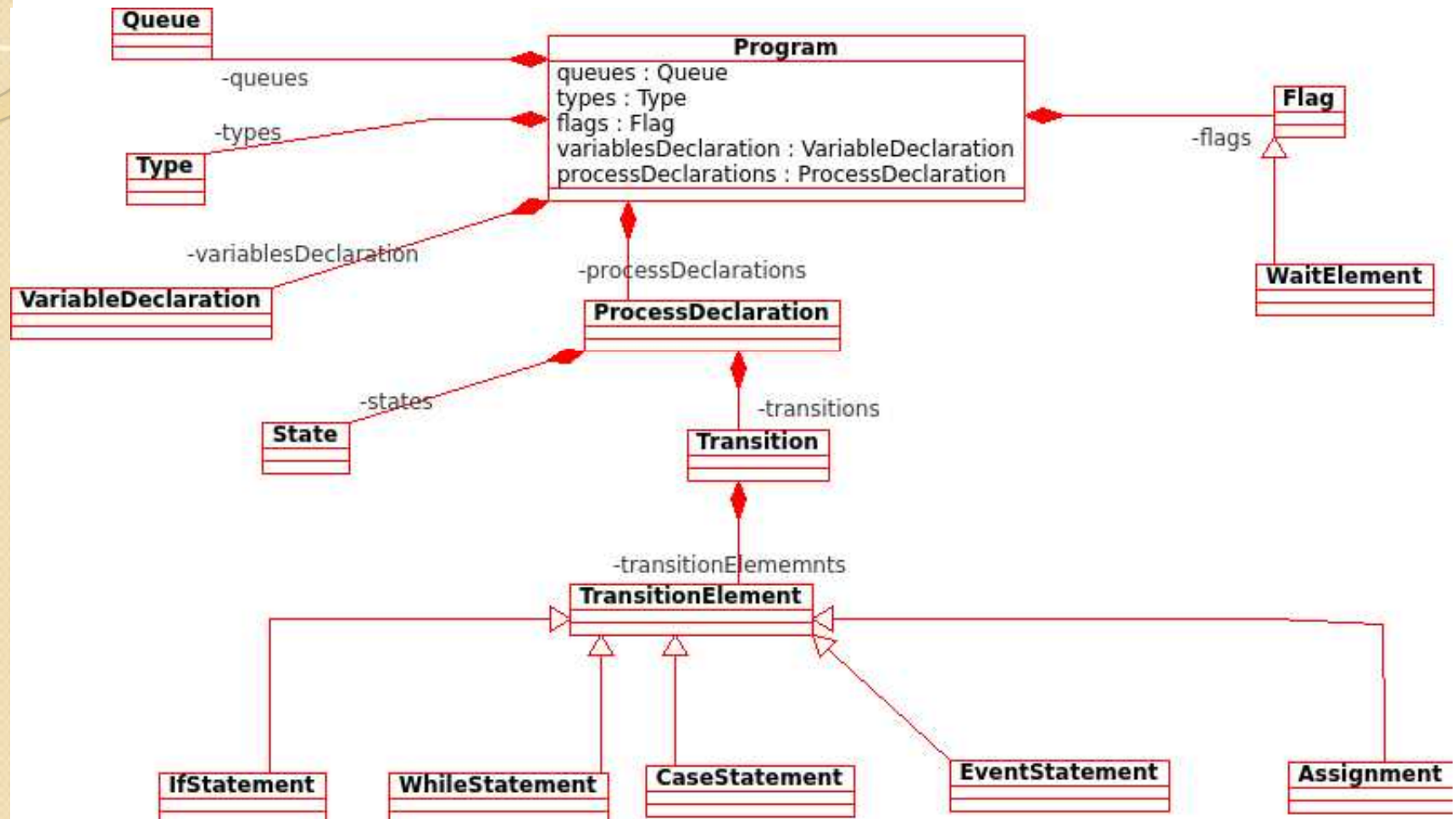
Principe de fonctionnement de l'outil de transformation de modèle MCSE en modèle Fiacre



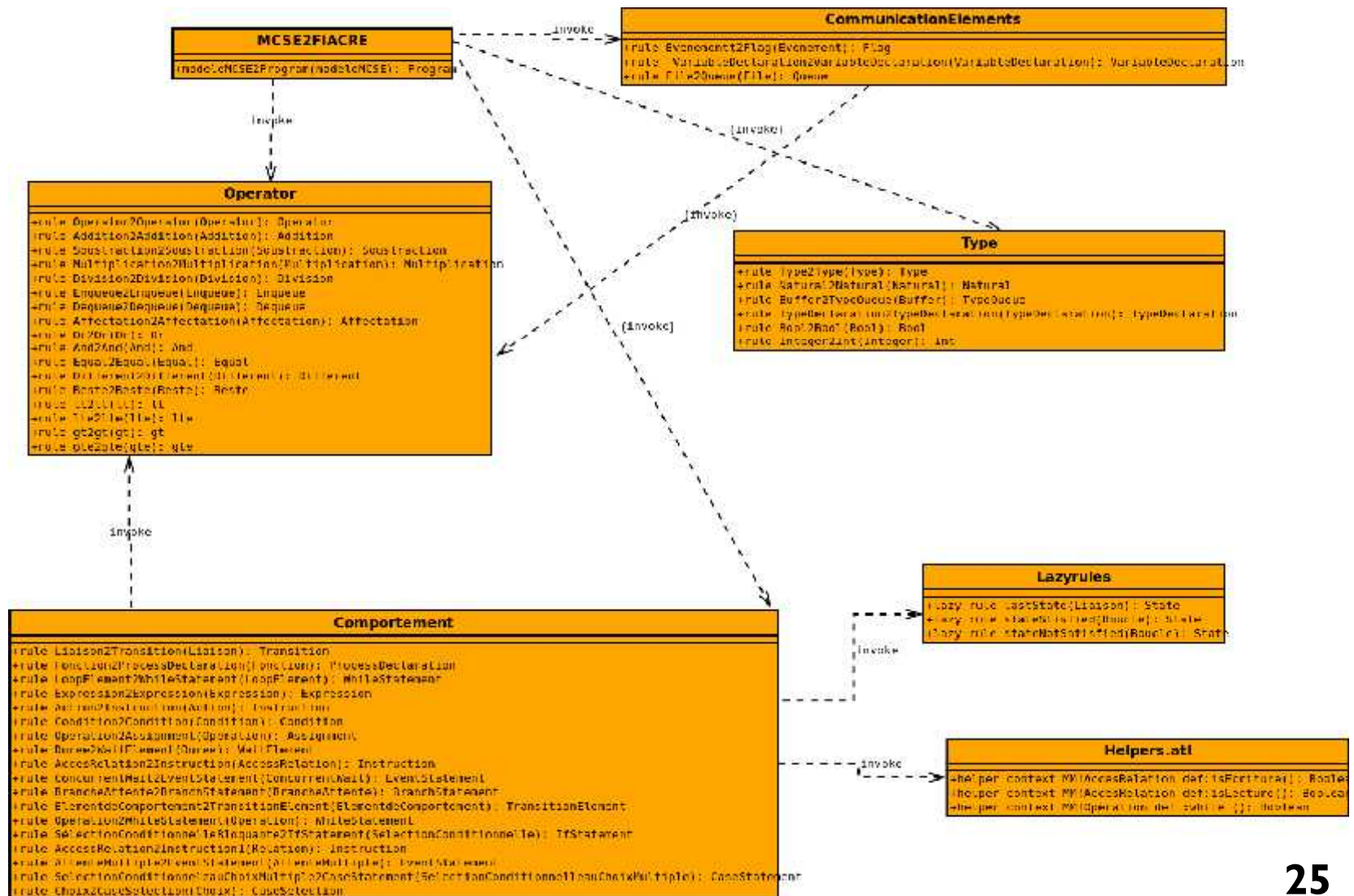
Méta-modèle de MCSE



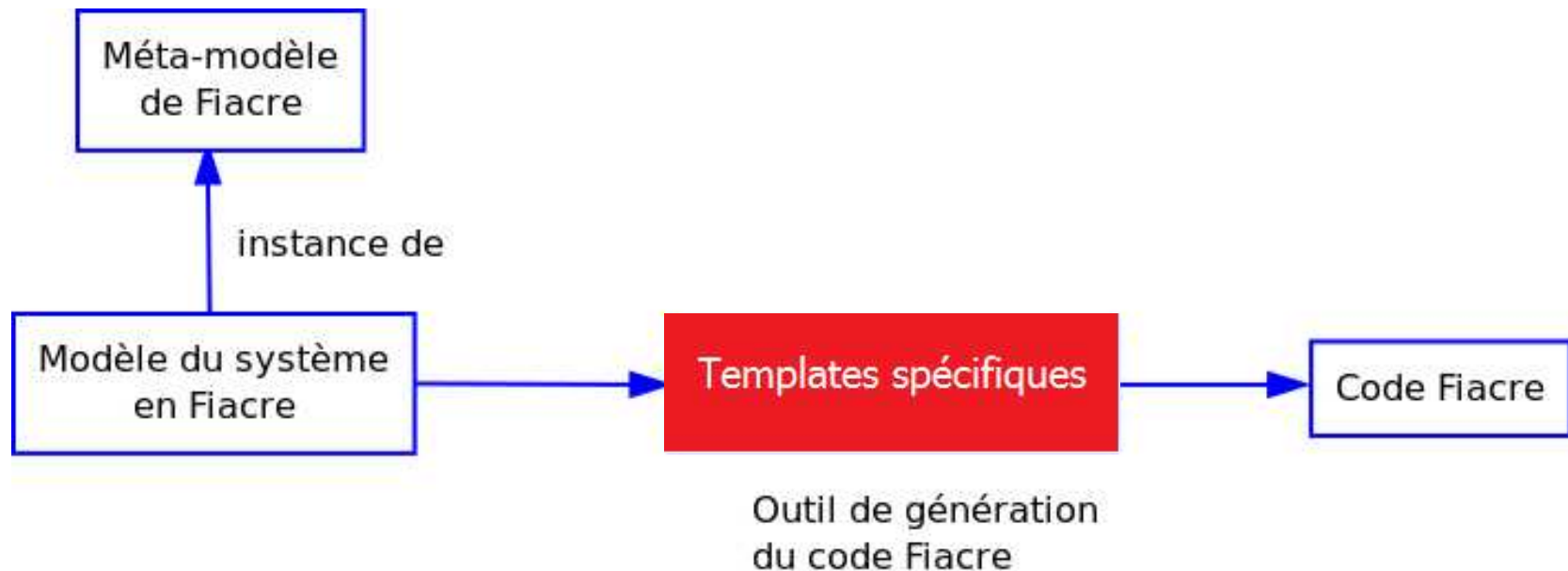
Méta-modèle de Fiacre



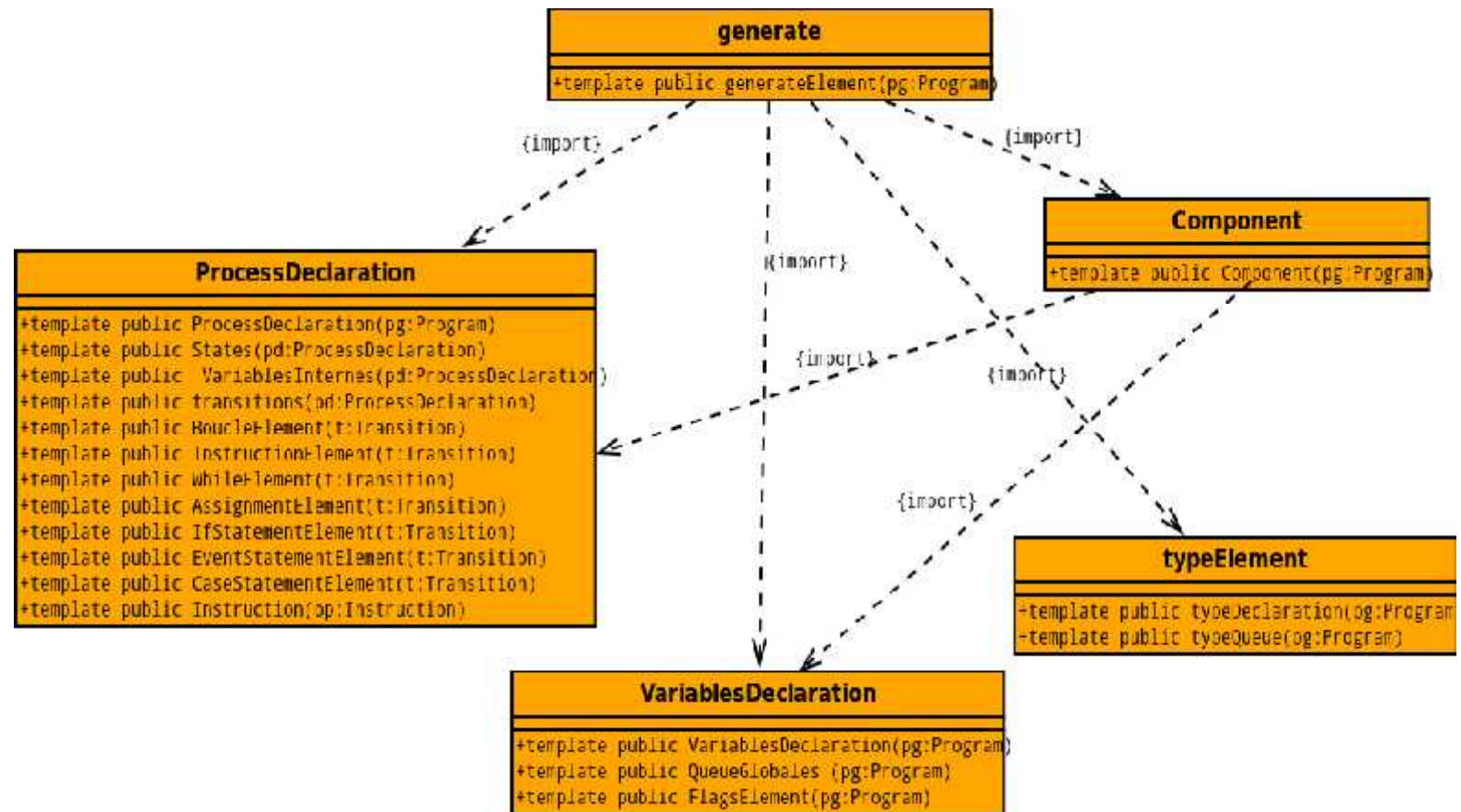
Architecture de l'outil de transformation de modèle MCSE en modèle Fiacre



Principe de fonctionnement de l'outil de génération du code



Architecture de l'outil de génération du code





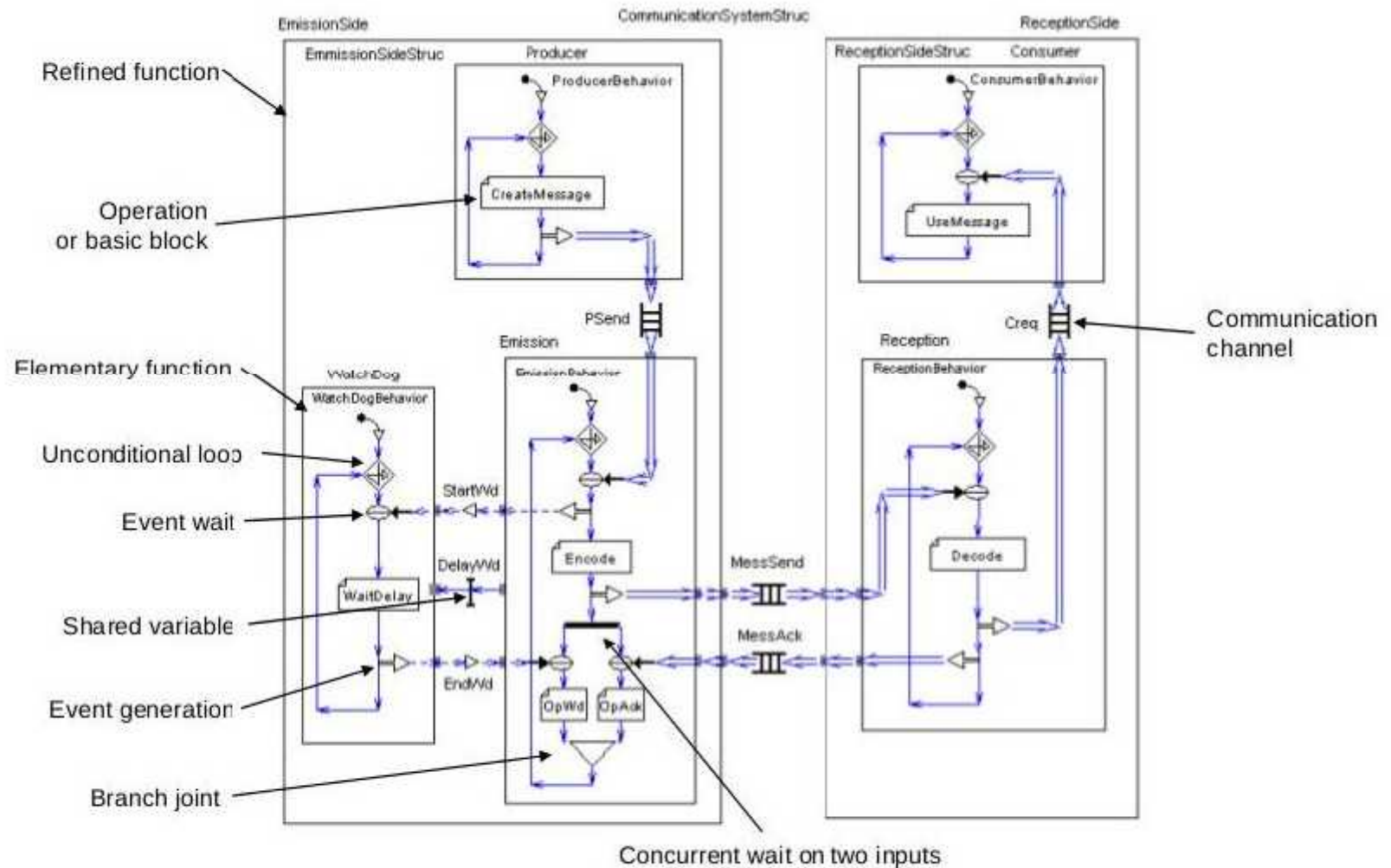
Plan de la présentation

- ❖ Contexte Générale
- ❖ Etat de l'art
- ❖ Approche
- ❖ Analyse et Spécification des besoins
- ❖ Conception
- ❖ **Réalisation**
- ❖ Conclusion et Perspectives

Choix techniques



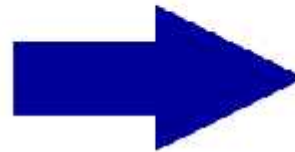
Modèle MCSE du système étudié



Transformation du modèle MCSE du système vers son modèle en Fiacre

▼ ♦ Modele MCSE CommunicationSystemStruc

- ♦ Int int
- ♦ Bool bool
- ♦ Natural nat
- ♦ Buffer bufferInt
- ▶ ♦ Fonction Producer
- ▶ ♦ Fonction Emission
- ▶ ♦ Fonction WatchDog
- ▶ ♦ Fonction Consumer
- ▶ ♦ Fonction Reception
- ♦ File PSEND
- ♦ File MessSend
- ♦ File MessAck
- ♦ File Creq
- ♦ Evenement StartWd
- ♦ Evenement EndWd
- ♦ Variable Partage DelayWd



▼ ♦ Program CommunicationSystemStruc

- ♦ Variable Declaration DelayWd
- ▶ ♦ Process Declaration Producer
- ▶ ♦ Process Declaration Emission
- ▶ ♦ Process Declaration WatchDog
- ▶ ♦ Process Declaration Consumer
- ▶ ♦ Process Declaration Reception
- ♦ Flag StartWd
- ♦ Flag EndWd
- ♦ Int int
- ♦ Bool bool
- ♦ Natural nat
- ♦ Type Queue bufferInt
- ♦ Queue PSEND
- ♦ Queue MessSend
- ♦ Queue MessAck
- ♦ Queue Creq

Code Fiacre généré du système

```
process Producer (&PSEND: bufferInt)

process Emission (&PSEND: bufferInt, &StartWd: bool, &DelayWd: int, &EndWd: bool)

process WatchDog (&StartWd: bool, &EndWd: bool, &DelayWd: int)

process Consumer (&Creq: bufferInt)

process Reception (&MessSend: bufferInt, &MessAck: bufferInt)

component CommunicationSystemStruc
  is
    var DelayWd: int := 0,
        PSEND: bufferInt := {},
        MessSend: bufferInt := {},
        MessAck: bufferInt := {},
        Creq: bufferInt := {},
        StartWd: bool := false,
        EndWd: bool := false
  par * in
    Producer (&PSEND)
    || Emission (&PSEND,&StartWd,&DelayWd,&EndWd)
    || WatchDog (&StartWd,&EndWd,&DelayWd)
    || Consumer (&Creq)
    || Reception (&MessSend,&MessAck)
  end

CommunicationSystemStruc
```


Informations sur le système

```
mohamed@mohamed-Inspiron-N5110:~/Bureau/PFE-LAAS/rapport et articles$ tina communicationSystem.tts communicationSystem.ktz
# net process_Reception_1_Consumer_1_WatchDog_1_Emission_1_Producer_1, 15 places, 16 transitions#
# bounded, not live, not reversible #
# abstraction      count      props      psets      dead      live #
#   states         1664       15         1664       0         768 #
#   transitions    6560       16         16         2         14 #
```

Vérification de l'inter-blocage

```
mohaned@mohamed-Inspiron-N5110:~/Bureau/PFE-LAAS/rapport et articles$ selt communicationSystem.ktz -f "-" - dead"  
Selt version 3.2.0 -- 02/22/14 -- LAAS/CNRS  
ktz loaded, 2816 states, 11176 transitions  
0.012s  
TRUE  
0.000s
```

Vérification de la divergence temporelle

```
nohamed@nohamed-Inspiron-N5110:~/Bureau/PFE-LAAS/rapport et articles$ selt communicationSystem.ktz -f " - div"  
Selt version 3.2.0 -- 02/22/14 -- LAAS/CNRS  
ktz loaded, 2816 states, 11176 transitions  
0.020s  
TRUE  
0.000s
```

Résultat de la vérification d'une formule de type LTL

```
nohamed@nohared-Inspiron-N5110:~/Bureau/PFE-LAAS/rapport et articles$ selt communicationSystem.ktz -f " [] WatchDog_1_sbEventOnStartkdWait => WatchDog_1_sbWaitDelay"
Selt version 3.2.8 -- 02/22/14 -- LAAS/CNRS
ktz loaded, 1664 states, 6560 transitions
0.012s
FALSE
state 0: Consumer_1_sbLectureCreq Emission_1_sbDequeuePSEND Producer_1_sbCreateMessage Reception_1_sbLectureMessSEND WatchDog_1_sbEventOnStartkdWait WatchDog_1_vbvar3
-Producer_1_t0 ... (preserving WatchDog_1_sbWaitDelay /\ WatchDog_1_sbEventOnStartkdWait)->
state 1: Consumer_1_sbLectureCreq Emission_1_sbDequeuePSEND Producer_1_sbEnqueuePSEND Reception_1_sbLectureMessSEND WatchDog_1_sbEventOnStartkdWait WatchDog_1_vbvar3
-Producer_1_t1 ... (preserving WatchDog_1_sbEventOnStartkdWait)->
* [accepting] state 4: Consumer_1_sbLectureCreq Emission_1_sbEncode Producer_1_sbCreateMessage Reception_1_sbLectureMessSEND WatchDog_1_sbEventOnStartkdWait WatchDog_1_vbvar3 CommunicationSystemStruc_1_vStartWd
-Reception_1_t0 ... (preserving WatchDog_1_sbEventOnStartkdWait)->
state 4: Consumer_1_sbLectureCreq Emission_1_sbEncode Producer_1_sbCreateMessage Reception_1_sbLectureMessSEND WatchDog_1_sbEventOnStartkdWait WatchDog_1_vbvar3 CommunicationSystemStruc_1_vStartWd
0.008s
```



Plan de la présentation

- ❖ Contexte Générale
- ❖ Etat de l'art
- ❖ Approche
- ❖ Analyse et Spécification des besoins
- ❖ Conception
- ❖ Réalisation
- ❖ **Conclusion et Perspectives**

Conclusion

Ce travail tourne au tour :

- ❖ Une étude sur la méthodologie MCSE
- ❖ Une étude sur les transformations des modèles
- ❖ Une étude sur la vérification formelle
- ❖ La mise en place d'une solution permettant l'application des techniques de vérification formelle en utilisant le concept de transformation des modèles

Conclusion

- ❖ Ce travail a été conclu par la rédaction d'un article scientifique

Perspectives

Ce travail peut être enrichi par :

❖ L'automatisation du Processus



MERCI POUR VOTRE ATTENTION

Analyse et spécification des besoins

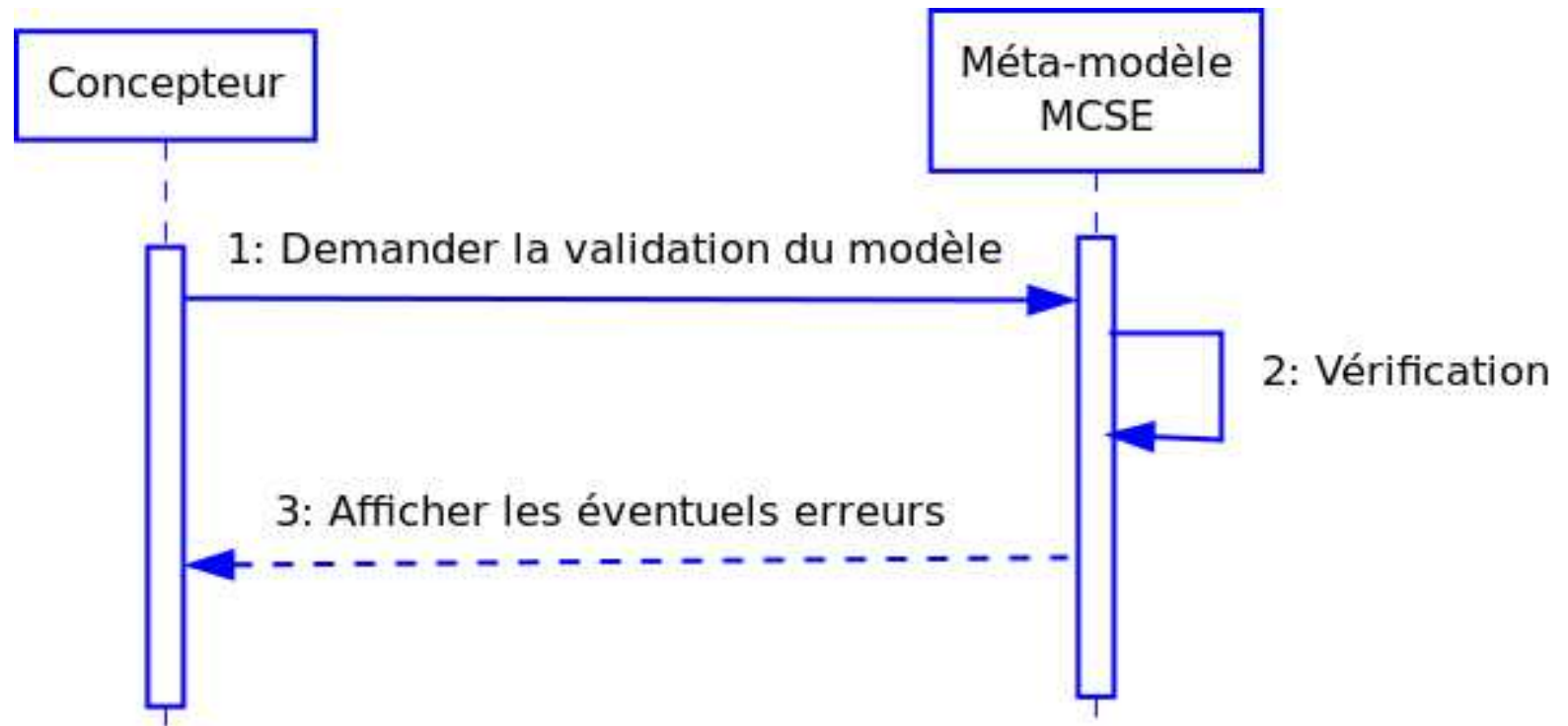


Diagramme de séquence du cas d'utilisation
« Valider le modèle créé »

Analyse et spécification des besoins

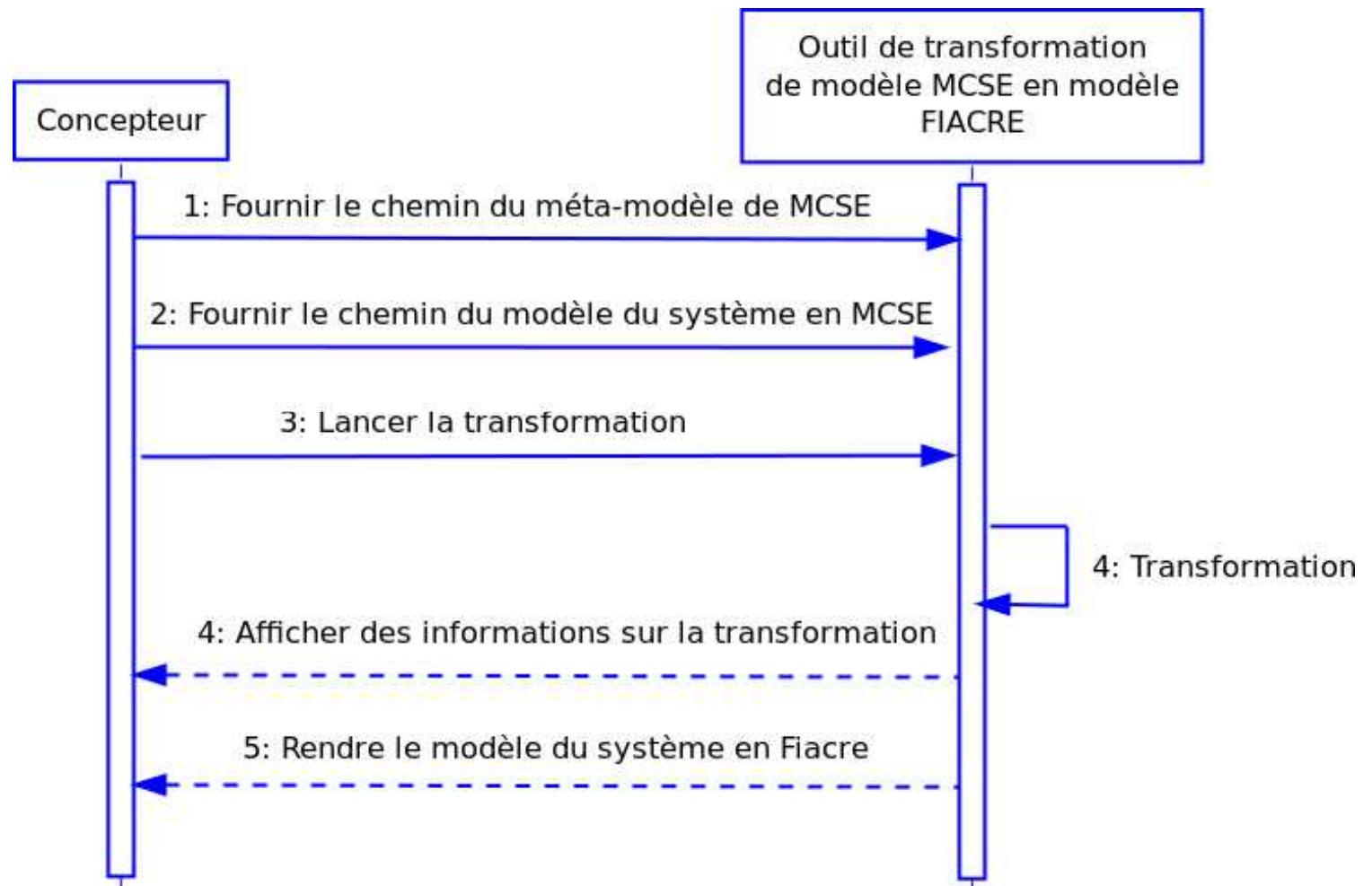


Diagramme de séquence du cas d'utilisation
« Transformer le modèle MCSE en modèle Fiacre »

Analyse et spécification des besoins

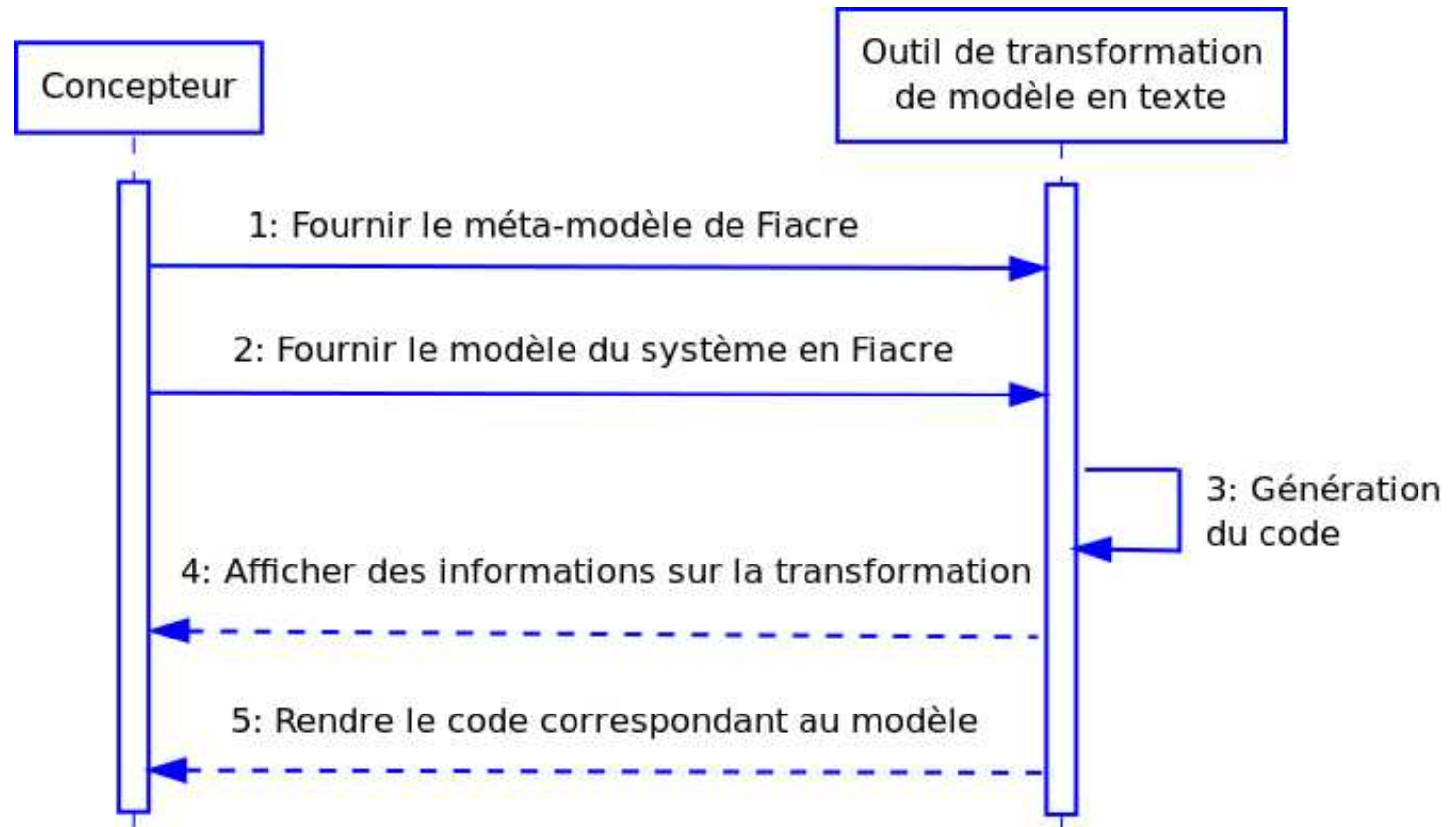


Diagramme de séquence du cas d'utilisation
« Générer le code Fiacre du système »

Analyse et spécification des besoins

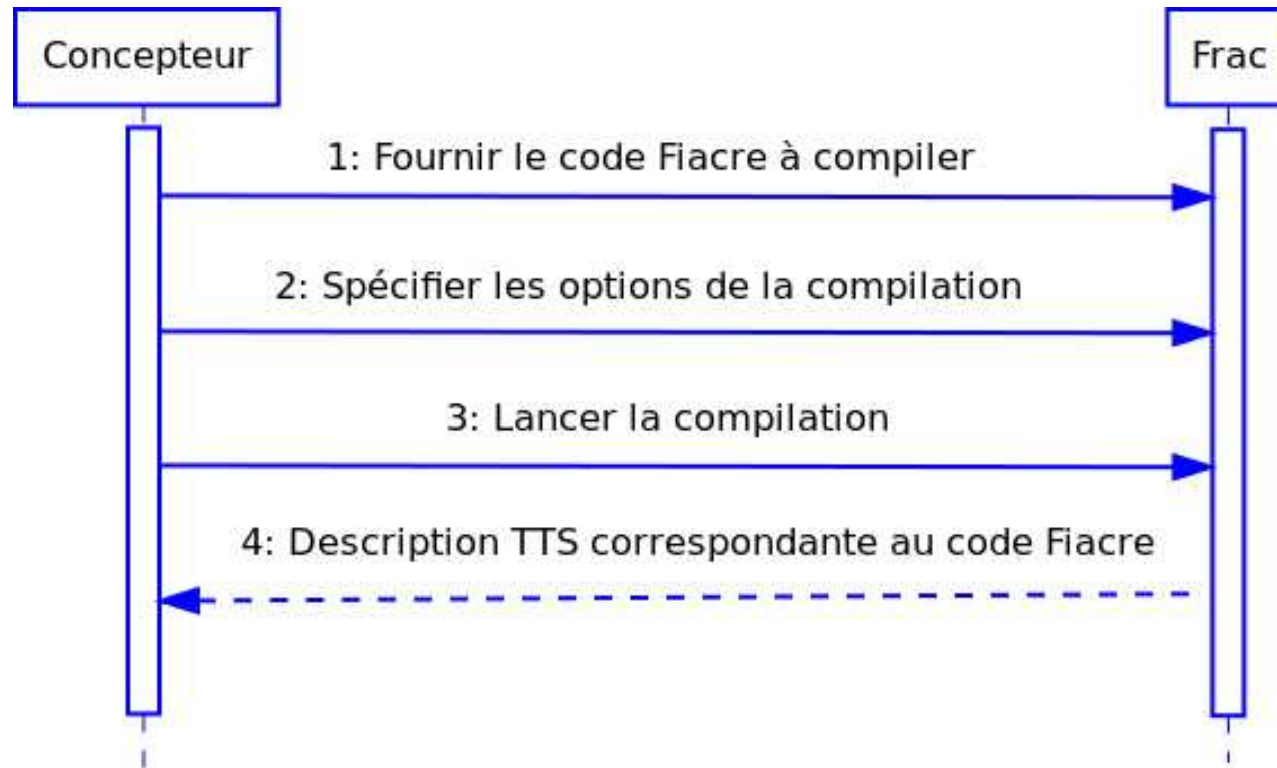


Diagramme de séquence du cas d'utilisation
« Compiler le code Fiacre »

Analyse et spécification des besoins

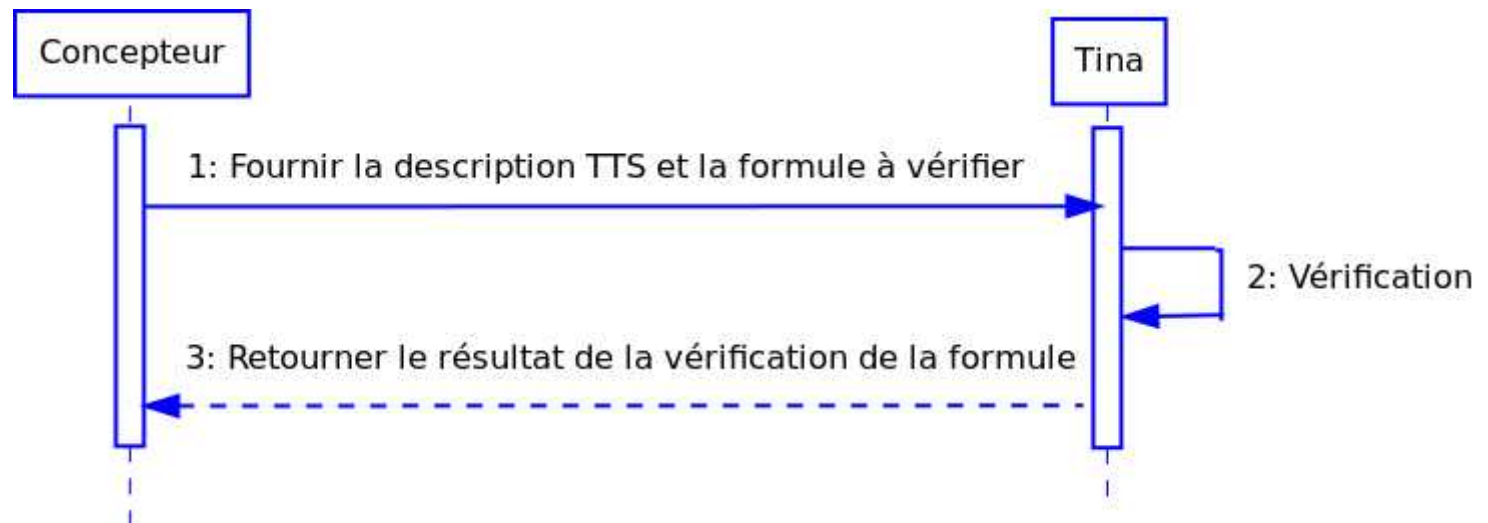
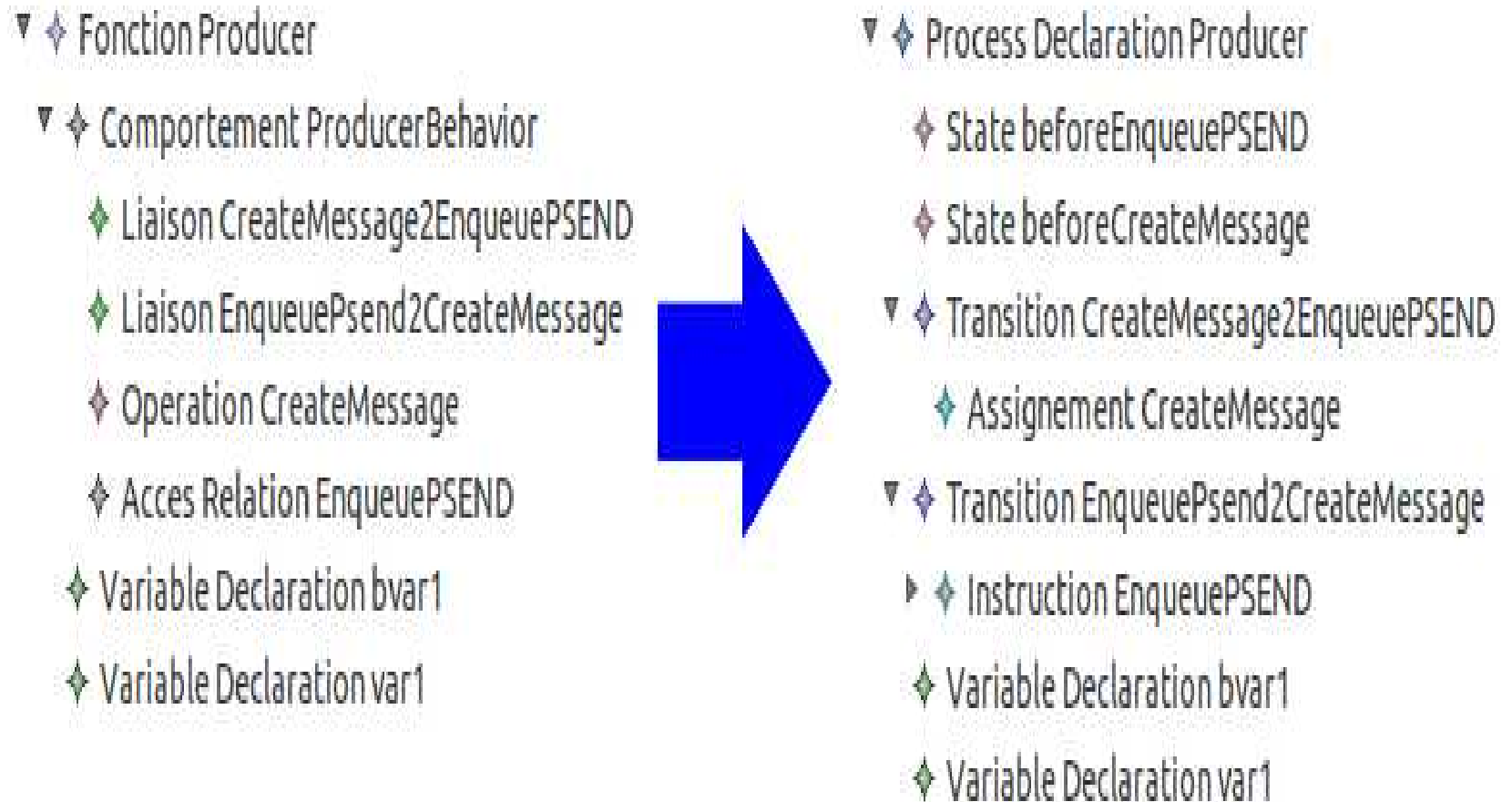


Diagramme de séquence du cas d'utilisation
« Vérifier une formule avec l'outil Tina »

Réalisation



Transformation de la fonction Producer vers le processus Fiacre Producer

Réalisation

▼ ♦ Fonction Emission

▼ ♦ Comportement EmissionBehaviour

- ♦ Liaison DequeuePSEND2EnableStartWd
- ♦ Liaison EnableStartWd2Encode
- ♦ Liaison Encode2EnqueueMessSEND
- ♦ Liaison EnqueueMessSEND2ConcurrentWait
- ♦ Liaison ConcurrentWait2DequeuePSEND

- ▶ ♦ Acces Relation DequeuePSEND
- ▶ ♦ Acces Relation EnableStartWd
- ♦ Operation Encode
- ▶ ♦ Acces Relation EnqueueMessSEND

▼ ♦ Attente Multiple ConcurrentWait

▼ ♦ Branche Attente

- ♦ Operation OpWd
- ♦ Acces Relation LectureEndWd

▼ ♦ Branche Attente

- ♦ Operation OpAck
- ♦ Acces Relation LectureMessAck

- ♦ Variable Declaration bvar2

- ♦ Variable Declaration var2

▼ ♦ Process Declaration Emission

- ♦ State beforeEnableStartWd
- ♦ State beforeEncode
- ♦ State beforeEnqueueMessSEND
- ♦ State beforeConcurrentWait
- ♦ State beforeDequeuePSEND

▼ ♦ Transition DequeuePSEND2EnableStartWd

- ▶ ♦ Instruction DequeuePSEND

▼ ♦ Transition EnableStartWd2Encode

- ▶ ♦ Instruction EnableStartWd

▼ ♦ Transition Encode2EnqueueMessSEND

- ♦ Assignment Encode

▼ ♦ Transition EnqueueMessSEND2ConcurrentWait

- ▶ ♦ Instruction EnqueueMessSEND

▼ ♦ Transition ConcurrentWait2DequeuePSEND

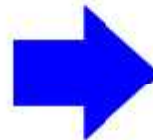
▼ ♦ Event Statement ConcurrentWait

▼ ♦ Branche Statement

- ♦ Assignment OpWd
- ♦ Instruction LectureEndWd

▼ ♦ Branche Statement

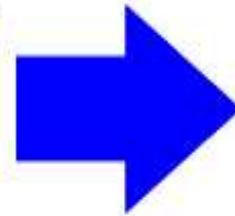
- ♦ Assignment OpAck
- ♦ Instruction LectureMessAck



Transformation de la fonction Emission vers le processus Fiacre Emission

Réalisation

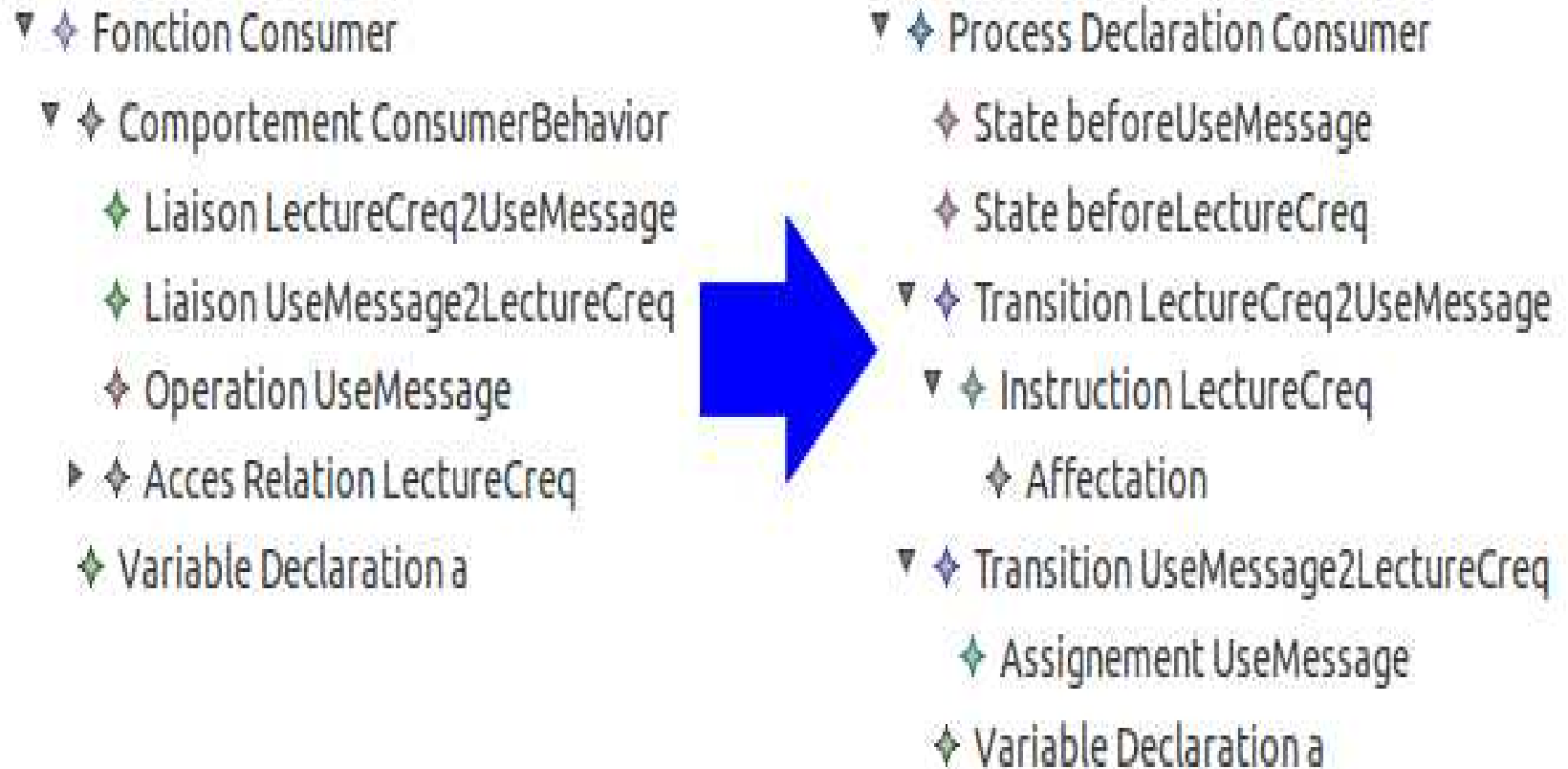
- ▼ ◆ Fonction WatchDog
 - ▼ ◆ Comportement WatchDogBehaviour
 - ◆ Liaison EventOnStartWd2WaitDelay
 - ◆ Liaison WaitDelay2EnableEndWd
 - ◆ Liaison EnableEndWd2WaitingOnStartWd
 - ▶ ◆ Acces Relation EventOnStartWdWait
 - ◆ Operation WaitDelay
 - ▶ ◆ Acces Relation EnableEndWd
 - ◆ Variable Declaration var3
 - ◆ Variable Declaration bvar3



- ▼ ◆ Process Declaration WatchDog
 - ◆ State beforeWaitDelay
 - ◆ State beforeEnableEndWd
 - ◆ State beforeEventOnStartWdWait
- ▼ ◆ Transition EventOnStartWd2WaitDelay
 - ▼ ◆ Instruction EventOnStartWdWait
 - ◆ Affectation
- ▼ ◆ Transition WaitDelay2EnableEndWd
 - ◆ Assignement WaitDelay
- ▼ ◆ Transition EnableEndWd2WaitingOnStartWd
 - ▼ ◆ Instruction EnableEndWd
 - ◆ Affectation
 - ◆ Variable Declaration var3
 - ◆ Variable Declaration bvar3

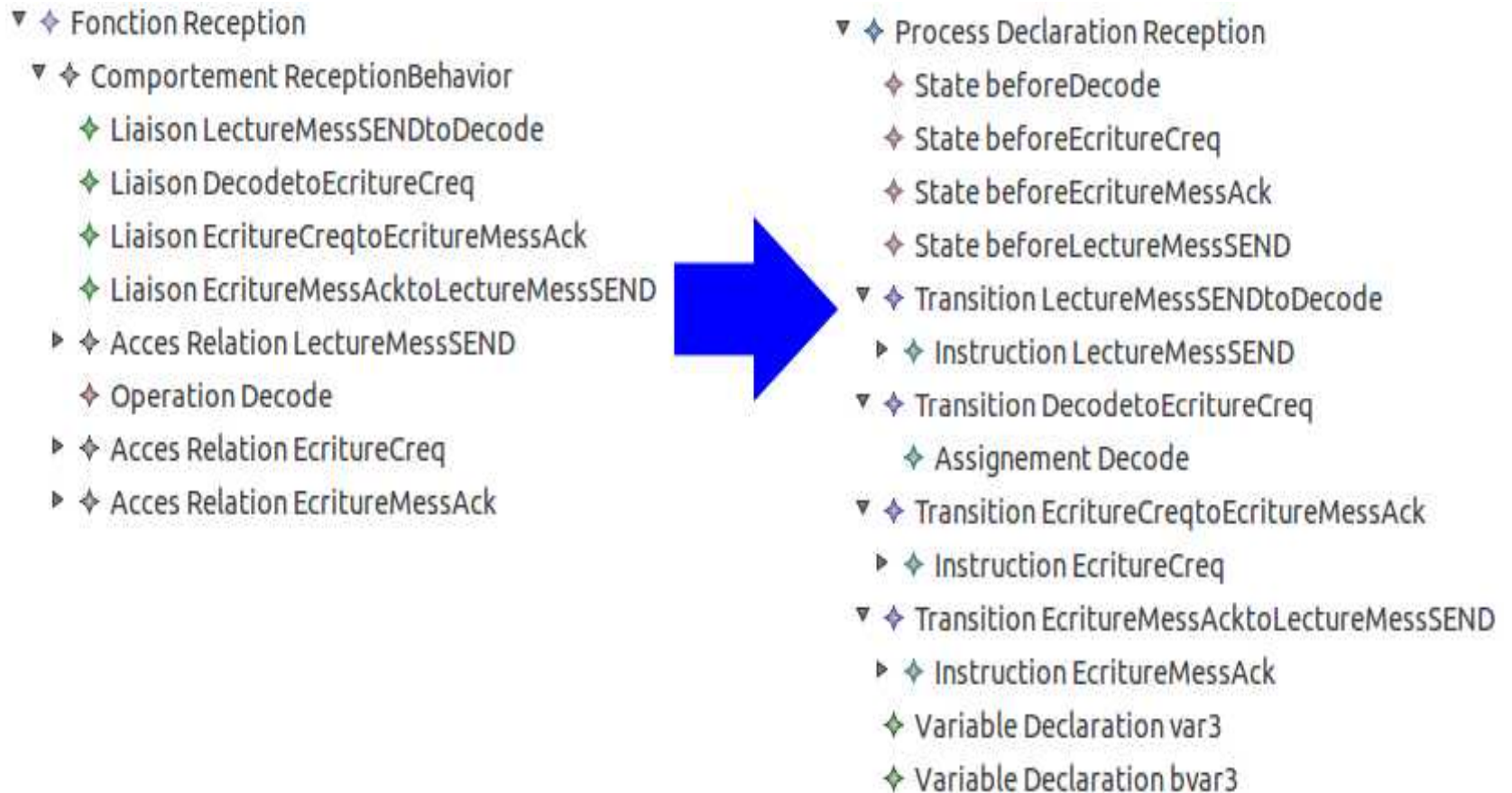
Transformation de la fonction Emission vers le processus Fiacre Emission

Réalisation



Transformation de la fonction Consumer vers le processus Fiacre Consumer

Réalisation



Transformation de la fonction Reception vers le processus Fiacre Reception