

# Ingénierie des Modèles II

Frédéric Boulanger

frederic.boulanger@centralesupelec.fr



<https://wdi.centralesupelec.fr/mde>

**CentraleSupélec**

3<sup>e</sup> année mention Science du logiciel

# Rappels

---

## Modèle

représentation simplifiée (abstraction) d'un système pour un objectif donné

- modèle d'analyse (système pré-existant)
- modèle de conception (système à concevoir)
- modèle explicatif, prédictif, constructif (exécutable)

## Système

objet d'étude, structure + interactions entre ses composants

## Conception

suite de raffinements de modèles : spécification  $\Rightarrow$  implémentation

## MDE/IDM

approche de la conception dans laquelle les modèles capturent la majeure partie de l'effort de conception

# Rappels

## Relations entre modèles

$$M \xrightarrow{\mu} S$$

M représente (est un modèle de) S

$$M \xrightarrow{\varepsilon} L$$

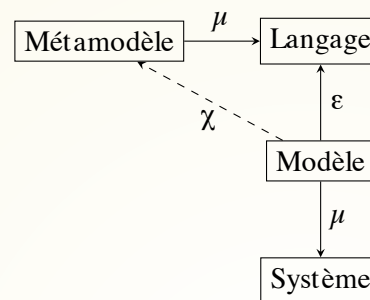
M appartient (est écrit en) langage L

$$M \xrightarrow{\delta} P$$

M se décompose en (contient) P

$$M \xrightarrow{\chi} MM$$

M est conforme au métamodèle MM



# Sémantique des modèles

# Sémantique des modèles



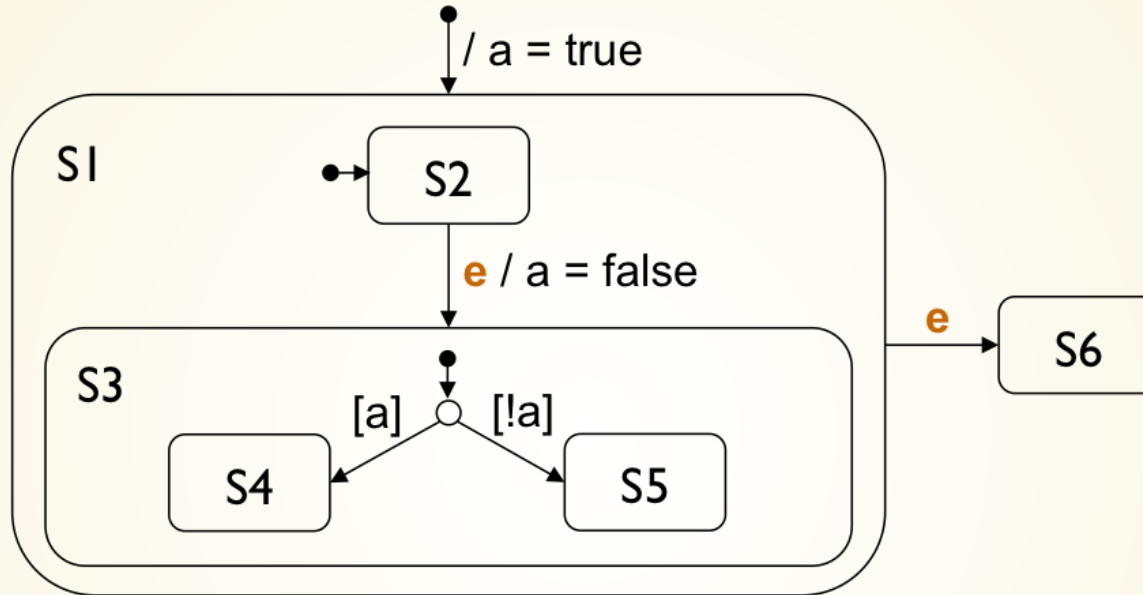
# Sémantique des modèles



Quel est le sens de « jaguar » ?

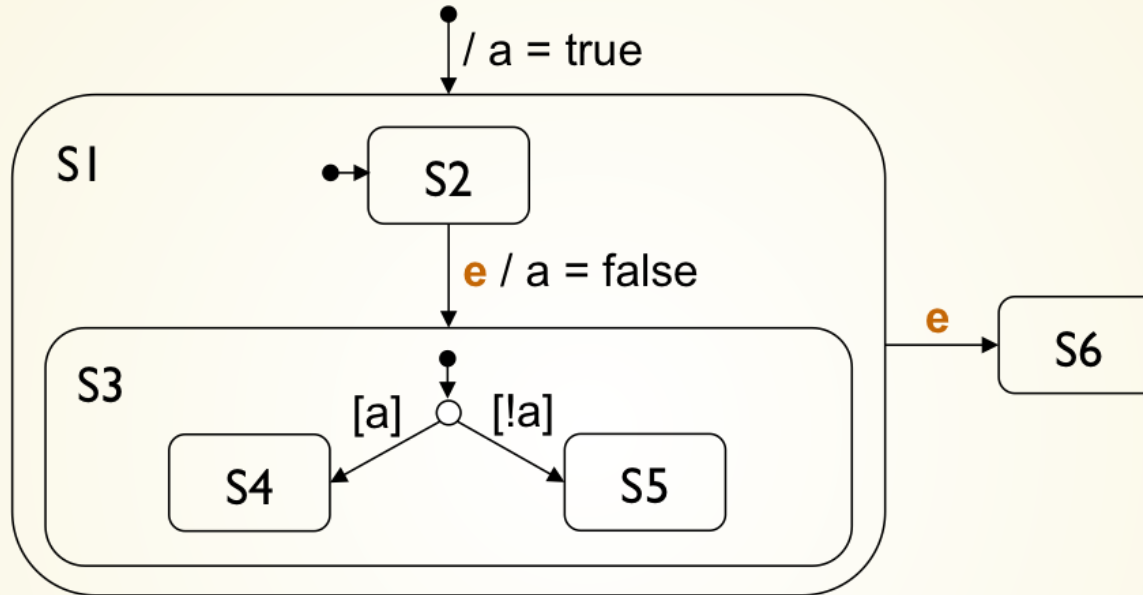
# Nécessité d'une sémantique précise

## Un diagramme StateCharts



# Nécessité d'une sémantique précise

## Un diagramme StateCharts

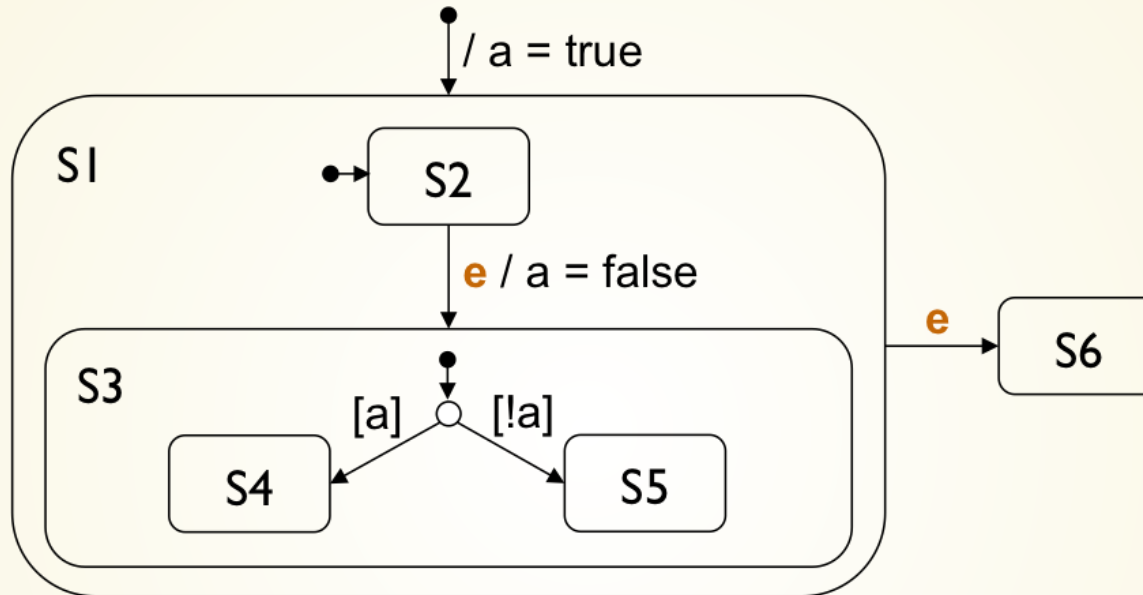


L'événement « e » peut mener à :



# Nécessité d'une sémantique précise

## Un diagramme StateCharts



### L'événement « e » peut mener à :

- S4 avec UML : la transition vers S1 a priorité et met a à true
- S5 avec Rhapsody : la transition de S2 vers S3 a priorité
- S6 avec Stateflow : la transition vers S6 préempte S1

# Nécessité d'une sémantique précise

Les trois sémantiques sont correctes

Le problème est que la sémantique est définie *implicitement* par l'outil !

Que se passe-t-il si :

- le concepteur pense selon la sémantique UML
- le générateur de code interprète le modèle comme Rhapsody
- la vérification est faite selon la sémantique de StateFlow ?

# Nécessité d'une sémantique précise

Les trois sémantiques sont correctes

Le problème est que la sémantique est définie *implicitement* par l'outil !

Que se passe-t-il si :

- le concepteur pense selon la sémantique UML
- le générateur de code interprète le modèle comme Rhapsody
- la vérification est faite selon la sémantique de StateFlow ?

La sémantique d'un modèle doit être *bien définie* et *explicite* pour éliminer les doutes sur l'interprétation et permettre la vérification de propriétés.

# Nécessité d'une sémantique précise

## Les trois sémantiques sont correctes

Le problème est que la sémantique est définie *implicitement* par l'outil !

Que se passe-t-il si :

- le concepteur pense selon la sémantique UML
- le générateur de code interprète le modèle comme Rhapsody
- la vérification est faite selon la sémantique de StateFlow ?

La sémantique d'un modèle doit être *bien définie* et *explicite* pour éliminer les doutes sur l'interprétation et permettre la vérification de propriétés.

## Sémantique formelle

Sémantique définie de telle manière qu'un modèle puisse être traité de façon automatique et consistante par des programmes.

# Syntaxe et sémantique

## Modèle informatique

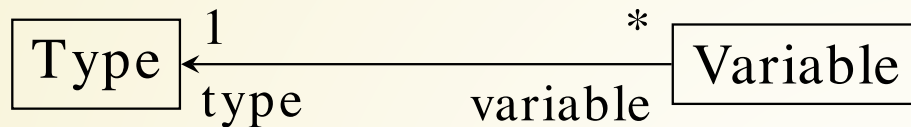
- un modèle est considéré en tant que système (transformation, validation)
- un modèle est dans un langage de modélisation (éditeurs, grammarware)
- un langage de modélisation est décrit par un métamodèle (modèle du langage)

## Métamodèle = syntaxe seule

- syntaxe abstraite (concepts et relations)
- syntaxe concrète (mots-clefs, éléments graphiques)

# Syntaxes abstraite et concrète

- syntaxe abstraite (concepts et relations)



- syntaxe concrète (notation à l'aide de symboles)

```
int i;
```

- autre syntaxe concrète textuelle

```
i: int;
```

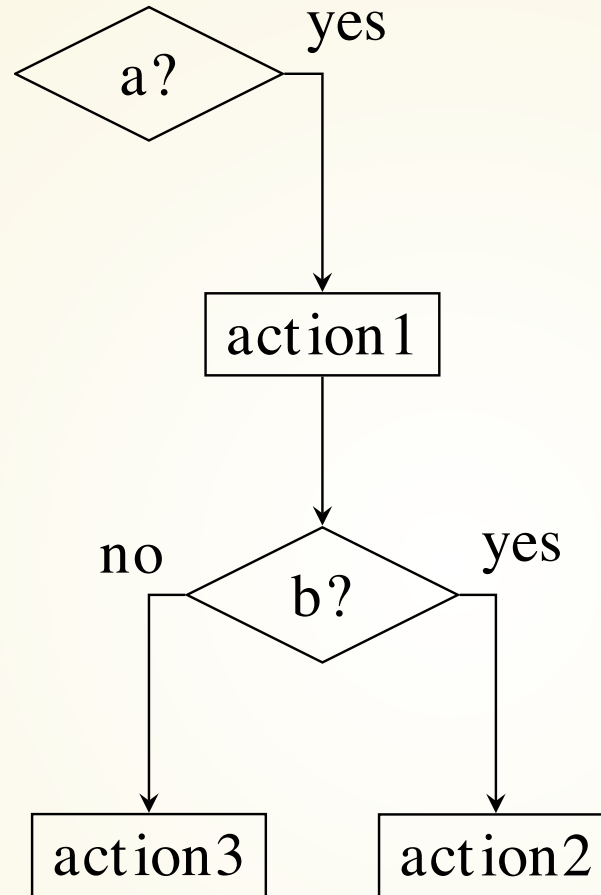
- syntaxe concrète graphique



# Nécessité d'une syntaxe non ambiguë

```
if (a) then  
do action1  
if (b) then  
do action2  
else  
do action3
```

# Nécessité d'une syntaxe non ambiguë

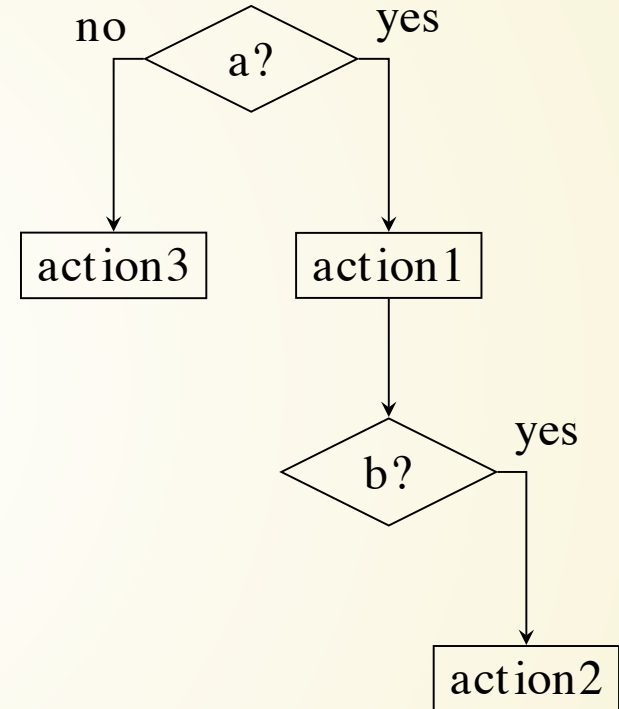
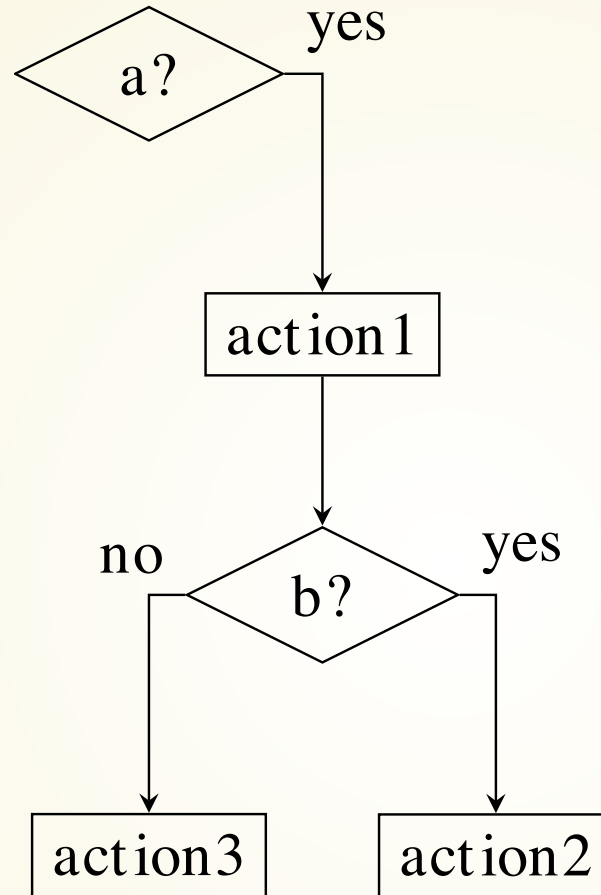


```
if (a) then  
do action1  
if (b) then  
do action2  
else  
do action3
```



# Nécessité d'une syntaxe non ambiguë

```
if (a) then
do action1
if (b) then
do action2
else
do action3
```



# Sémantique

**Sémantique** : sens donné aux éléments syntaxiques

Se définit par une association entre les éléments syntaxiques (signifiants) et les éléments du domaine sémantique (signifiés)

## Domaine sémantique

- autre langage (traduction)
- théorie mathématique

## Définition de la sémantique

1. définition formelle de la syntaxe abstraite
2. choix d'un domaine sémantique
3. association entre syntaxe abstraite et domaine sémantique

# Points clefs pour la sémantique des modèles

## Sémantique

- définition non ambiguë de la syntaxe
- choix d'un domaine sémantique
- définition d'une correspondance entre domaines syntaxique et sémantique

*Souvent donnée par une transformation de modèle*

# Points clefs pour la sémantique des modèles

## Sémantique

- définition non ambiguë de la syntaxe
- choix d'un domaine sémantique
- définition d'une correspondance entre domaines syntaxique et sémantique

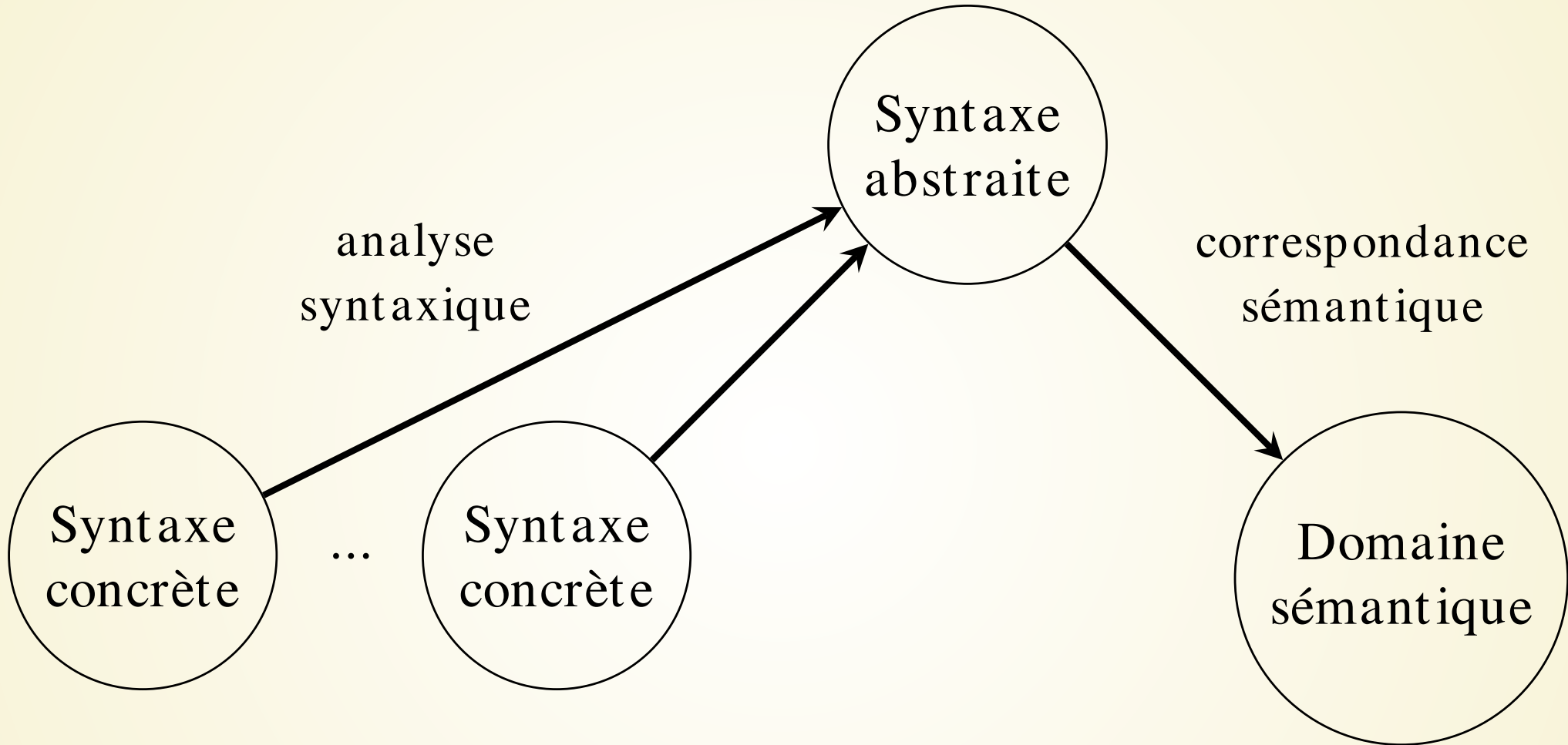
*Souvent donnée par une transformation de modèle*

## Sémantique statique

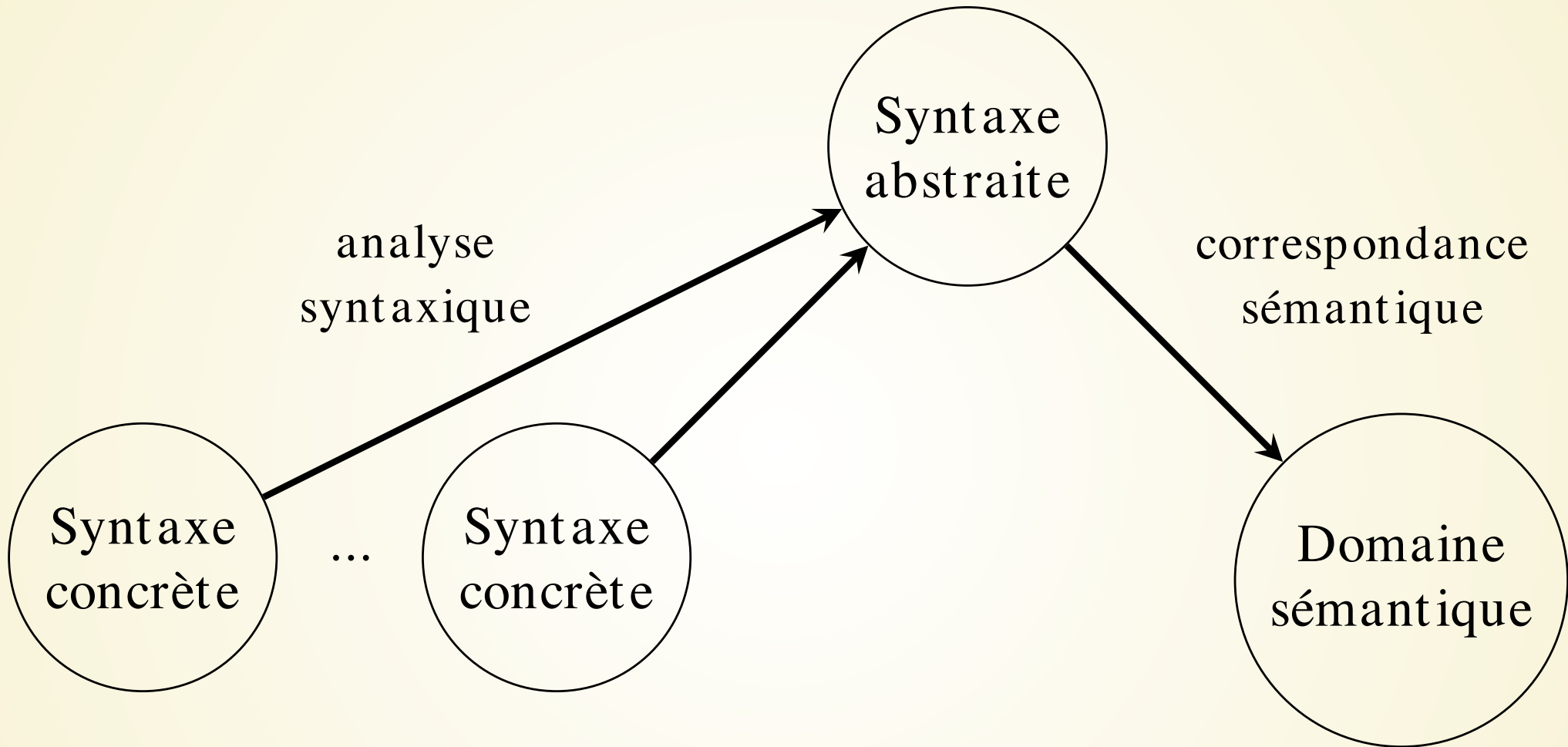
- exprime des contraintes sur les modèles
- n'est pas capturée par la syntaxe
- exemple : une variable est déclarée avant d'être utilisée

*Peut être exprimée sous forme d'invariants OCL par exemple*

# Définition de la sémantique d'un langage



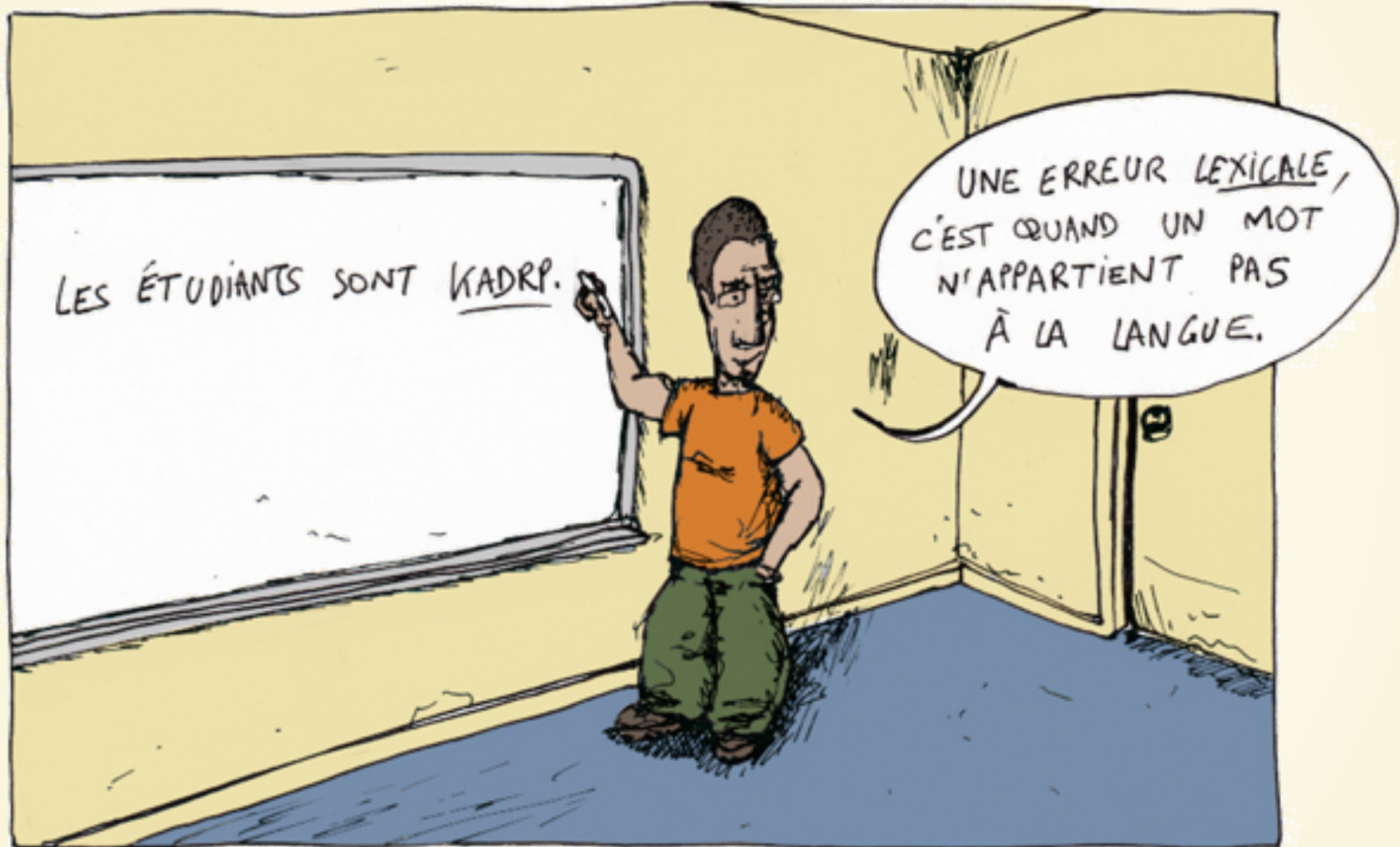
# Définition de la sémantique d'un langage



Plus de détails dans le cours « sémantique des langages et des modèles »

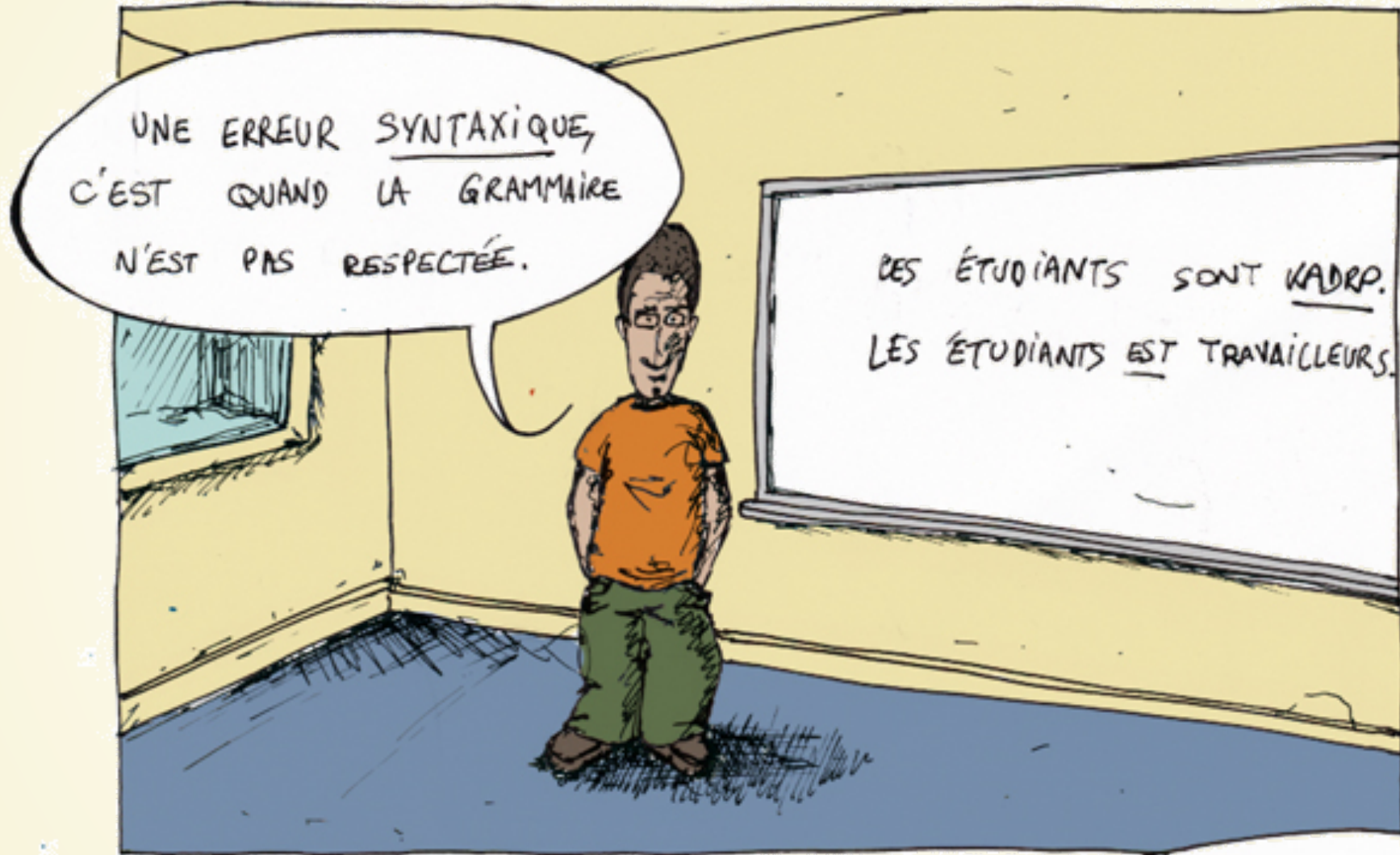
# Lexique, syntaxe et sémantique

# Lexique, syntaxe et sémantique





# Lexique, syntaxe et sémantique



# Lexique, syntaxe et sémantique

