Formalisation en Coq des systèmes de transitions modales

Stage de M2 Vérification logicielle

Étudiant : Joas Yannick Kinouani, Université de Bordeaux
Maîtres de stage : M. Jean-Paul Bodeveix, professeur d'université
M. Mamoun Filali, chargé de recherche au CNRS
Équipe : Assistance à la certification d'applications distribuées et embarquées
Laboratoire : Institut de recherche en informatique de Toulouse

Septembre 2015

université de BORDEAUX
UNIVERSITÉ TOULOUSE III PAUL SABATIER Université de Toulouse
IRIT CNRS - INPT - UPS - UT1 - UT2J

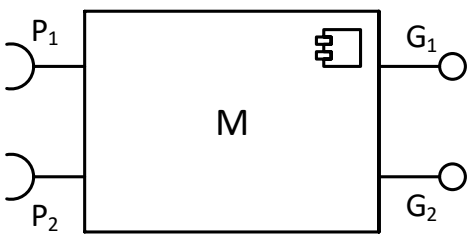
Plan

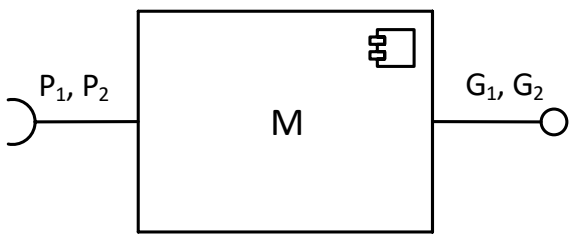
Plan

Plan

Développement à base de composants

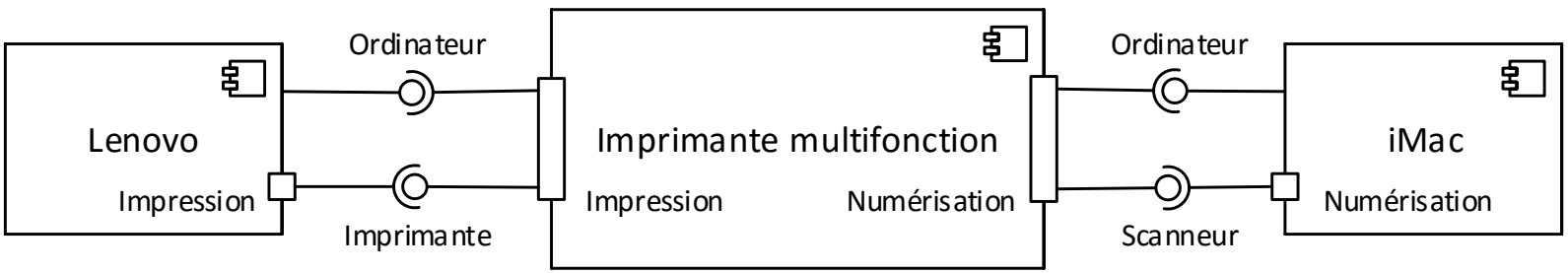
Composant : objet qui procure des services — ses garanties G_1, G_2, etc. — aussi longtemps que ses prérequis P_1, P_2, etc., sont respectés





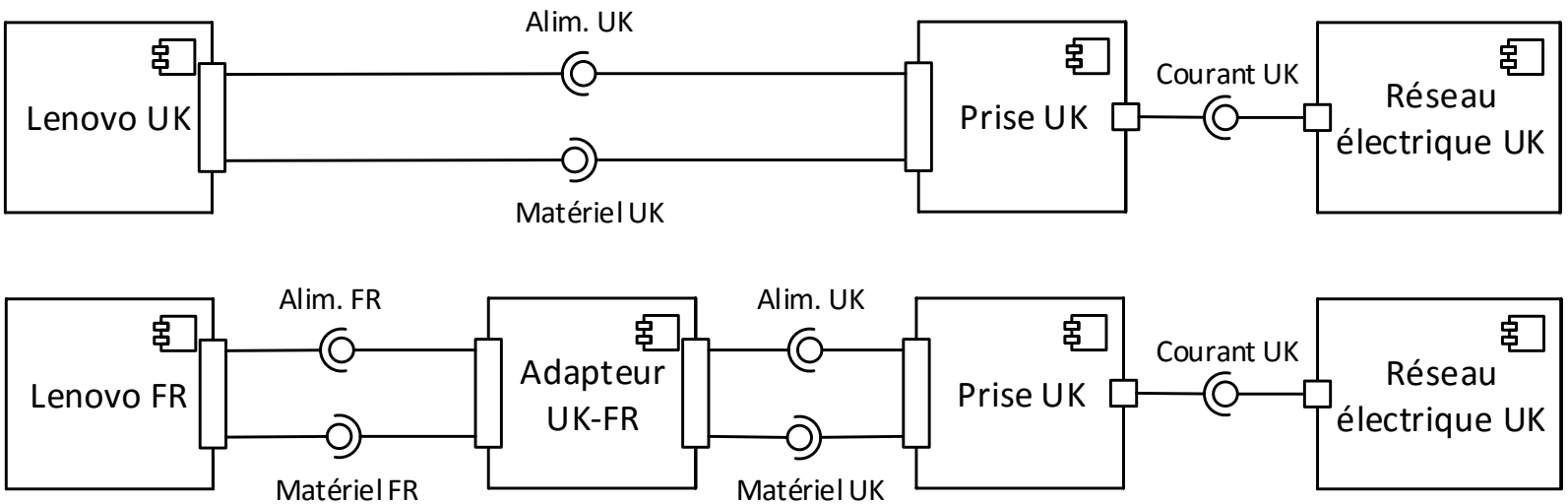
Assemblage de composants

- Un composant peut très bien proposer plusieurs services, chacun avec ses propres prérequis et garanties
- Exemple : imprimante multifonction



Remplacement de composants

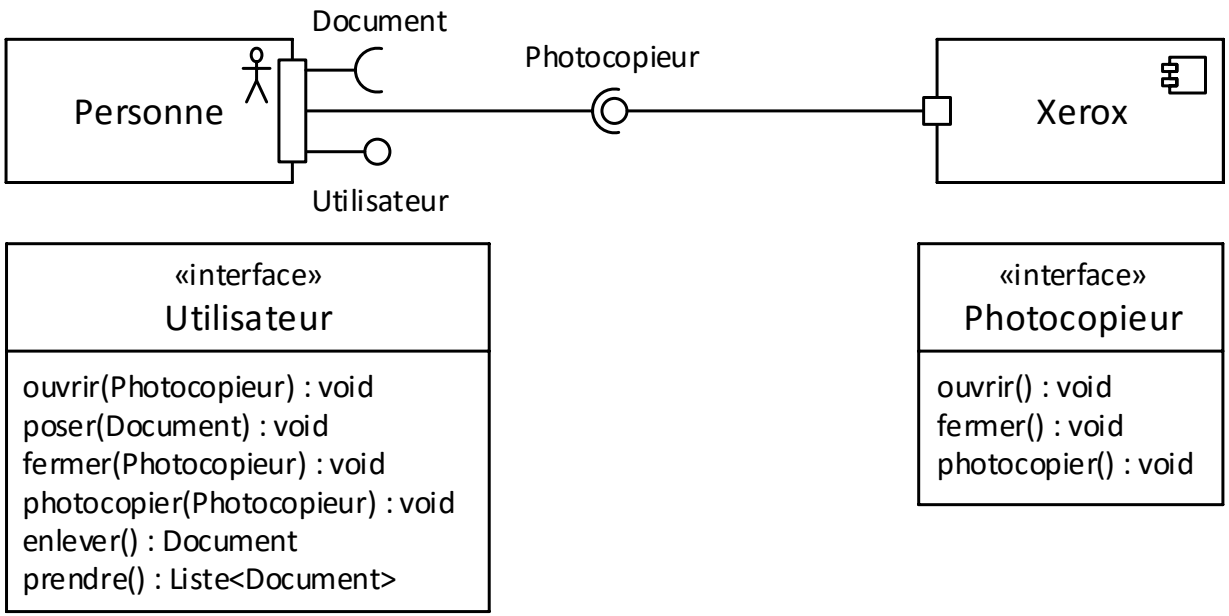
Exemple : remplacer, au Royaume-Uni (UK, United Kingdom), un ordinateur britannique par un ordinateur français (FR, France)



Plan

Spécification statique des prérequis/garanties

Spécification statique : liste des opérations + éventuelles pré/postconditions



Spécification dynamique des prérequis/garanties (1/4)

- Spécification dynamique : modélisation abstraite du comportement
- Outil standard : systèmes de transitions

Spécification dynamique des prérequis/garanties (2/4)

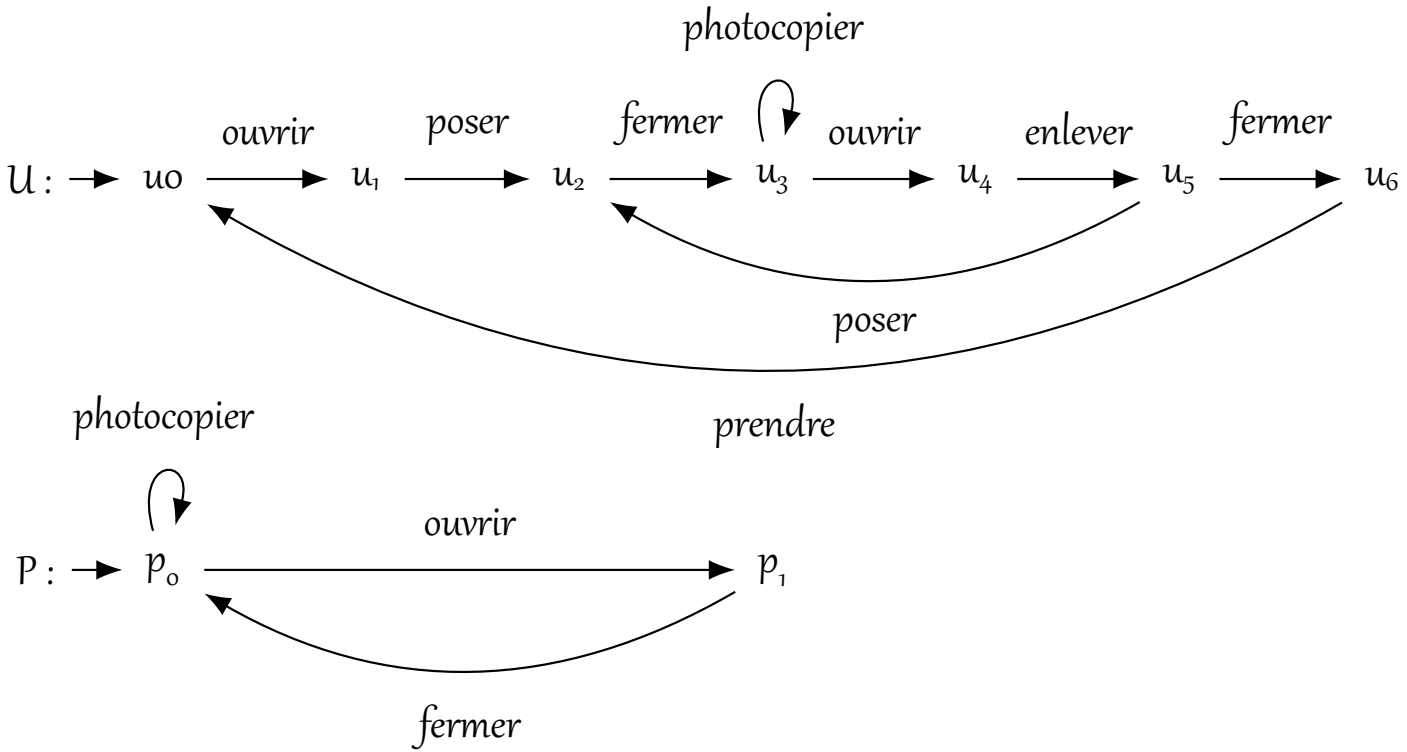
Définition (Système de transitions [Anorld, 1992])

Un système de transitions $\langle \mathcal{Q}, \mathcal{I}, \mathcal{E}, \mathcal{T}, \alpha, \beta, \lambda \rangle$ est un objet dont l'état est modifié par des évènements :

1. $\mathcal{Q}$ est un ensemble d'états
2. $\mathcal{I} \subseteq \mathcal{Q}$ est le sous-ensemble des états initiaux
3. $\mathcal{E}$ est un ensemble d'évènements
4. $\mathcal{T}$ est un ensemble de transitions
5. $\alpha, \beta \in \mathcal{T} \to \mathcal{Q}$ et $\lambda \in \mathcal{T} \to \mathcal{E}$ sont des fonctions totales qui associent à chaque transition $t \in \mathcal{T}$, une source $\alpha(t) \in \mathcal{Q}$, une cible $\beta(t) \in \mathcal{Q}$, et un évènement $\lambda(t) \in \mathcal{E}$ □

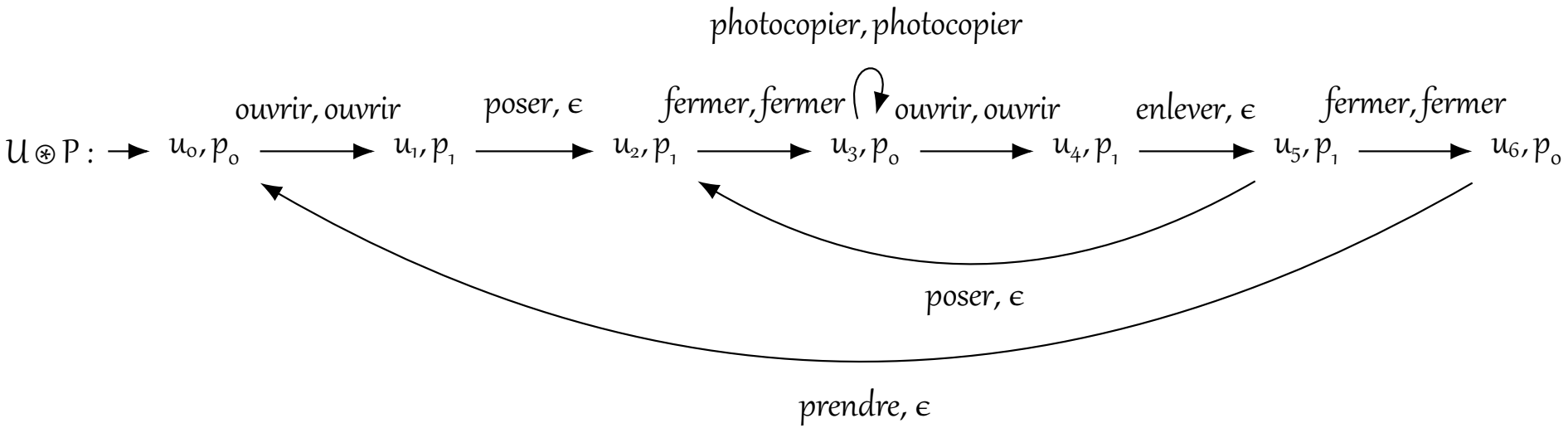
Spécification dynamique des prérequis/garanties (3/4)

Si l'utiliseur U dépose le document sous le capot du photocopieur P puis appuie sur le bouton, il sera photocopié



Spécification dynamique des prérequis/garanties (4/4)

- $U \circledast P$: comportement du photocopieur en présence de l'utilisateur
- Ici, $U \circledast P \equiv U$: l'utilisateur maîtrise le photocopieur



Plan

Plan

Théorie des contrats

Métathéorie des contrats [Benveniste, Damm, et al., 2012, Benveniste, Raclet, et al., 2012] : formalisation mathématique du développement à base de composants

Définition (Théorie des contrats [Benveniste, Raclet, et al., 2012])

Une théorie des contrats $\langle \mathcal{C}, \mathcal{R}, \equiv, \mathcal{L}, *, \vDash \rangle$ est la donnée :

1. d'un ensemble $\mathcal{C}$ de contrats
2. d'un ensemble partitionné ou setoïde $\langle \mathcal{R}, \equiv \rangle$ de réalisations
3. d'un sous-ensemble $\mathcal{L} \subseteq \mathcal{R}$ de réalisations dites légales
4. d'une opération de composition de réalisations $* \in \mathcal{R} \times \mathcal{R} \to \mathcal{R}$, partiel, $\equiv$-compatible, $\equiv$-commutatif, et $\equiv$-associatif
5. d'une relation de satisfaction $\vDash \subseteq \mathscr{R} \times \mathscr{C}$, $\equiv$-compatible □

Implémentation et consistance

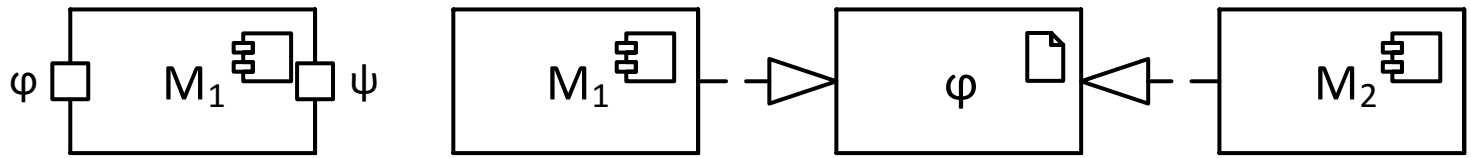
Définition (Implémentation, consistance, cohérence [Benveniste, Raclet, et al., 2012])

- Soit T une théorie des contrats
- On dit qu'une réalisation $M \in \mathcal{R}_\mathsf{T}$ est une implémentation du contrat $\varphi \in \mathcal{C}_\mathsf{T}$, ou que M implémente φ, si et seulement si $M \vDash_\mathsf{T} \varphi$
- L'ensemble

$$\mathrm{Imp}\ \varphi \triangleq \{\, M \in \mathcal{R}_\mathsf{T} \mid M \vDash_\mathsf{T} \varphi \,\}$$

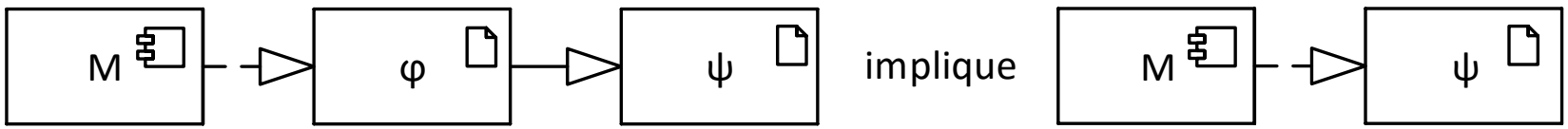
  est l'ensemble des implémentations de φ
- Un contrat $\varphi \in \mathcal{C}_\mathsf{T}$ est consistant ou cohérent si et seulement si $\mathrm{Imp}\ \varphi \neq \varnothing$, c.-à-d. si et seulement s'il est possible de réaliser une implémentation qui le satisfait
- Dans le cas contraire, le contrat est inconsistant ou incohérent □

Contrats et composants



- *Un composant est une réalisation, et ses services sont les contrats qu'il implémente*
- *À chaque contrat correspond un ensemble d'implémentations remplaçables les unes par les autres*
- *L'opération de composition ∗ formalise l'assemblage de composants*

Raffinement



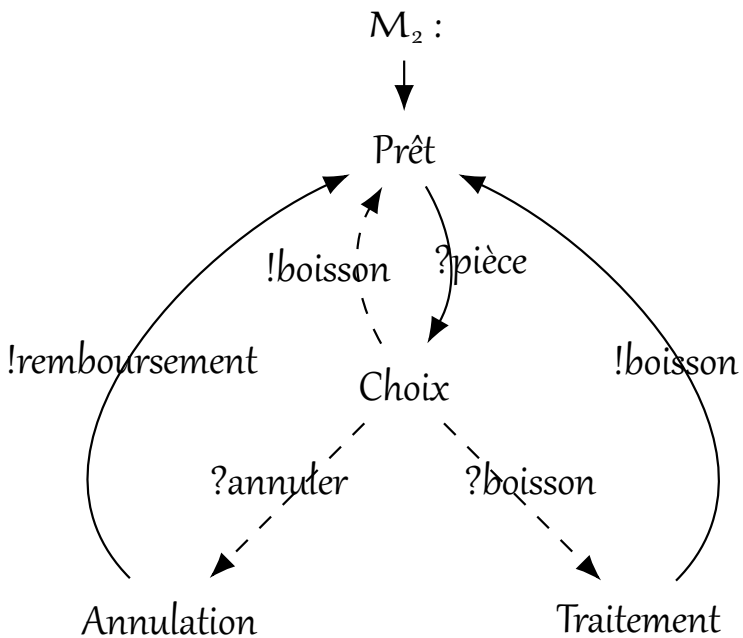
Définition (Raffinement [Benveniste, Raclet, et al., 2012])

- Soient T une théorie des contrats, et $\varphi, \psi \in \mathcal{C}_{\mathsf{T}}$ deux contrats
- On dit que φ est un raffinement de ψ, ou que φ raffine ψ, et on écrit "$\varphi \leq \psi$", si et seulement si $\mathrm{Imp}\,\varphi \subseteq \mathrm{Imp}\,\psi$
- Cela signifie que φ est un contrat plus précis ou plus restrictif que ψ, au sens où il élimine une partie des implémentations de ψ □

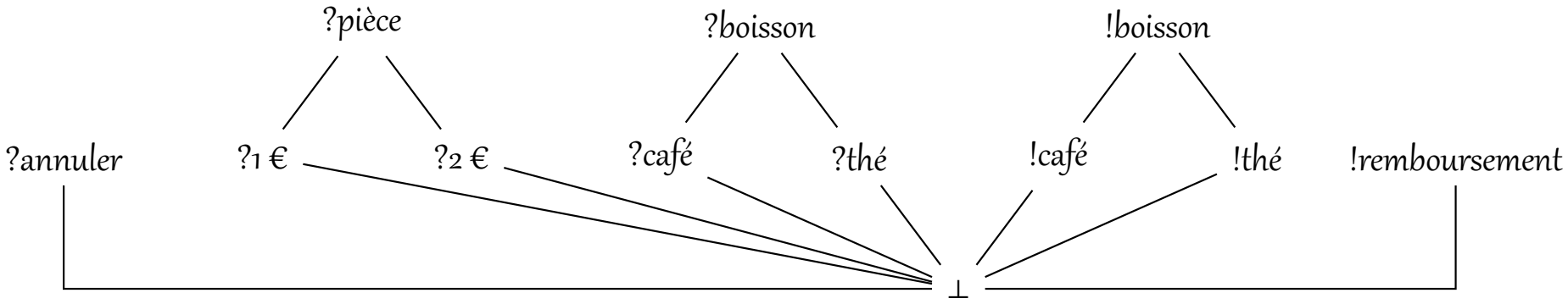
Plan

Contrats et systèmes de transitions modales

- Contrat = système de transitions + modalités (spécification dynamique)
- Modalité may (- -►) : la transition est permise : elle sera ou ne sera pas implémentée
- Modalité must (⟶) : la transition est obligatoire : elle sera implémentée
- ⟶ implique - -► : toute transition obligatoire est permise



Ensemble structuré d'étiquettes



Définition (Ensemble structuré d'étiquettes [Bauer, Juhl, et al., 2012])

Un ensemble structuré d'étiquettes $\langle \mathcal{E}, \preccurlyeq, \bot \rangle$ est la donnée :

1. d'un ensemble partiellement ordonné d'étiquettes $\langle \mathcal{E}, \preccurlyeq \rangle$
2. d'une plus petite étiquette $\bot \in \mathcal{E}$ □

Pour les systèmes de transitions, les étiquettes sont les évènements, et l'ordre partiel la relation de raffinement

Ensemble bien structuré d'étiquettes

Définition (Étiquette d'implémentation [Bauer, Juhl, et al., 2012])

- Soit E un ensemble structuré d'étiquettes
- Toute étiquette « juste au-dessus » de $\bot_{\mathsf{E}}$ est une étiquette d'implémentation □

Définition (Ensemble bien structuré d'étiquettes [Bauer, Juhl, et al., 2012])

Un ensemble bien structuré d'étiquettes est un ensemble structuré d'étiquettes E où toute étiquette distincte de $\bot_{\mathsf{E}}$ est précédée par une étiquette d'implémentation □

Contre-exemple : $\mathbb{R}^+$ avec $\bot = 0$.

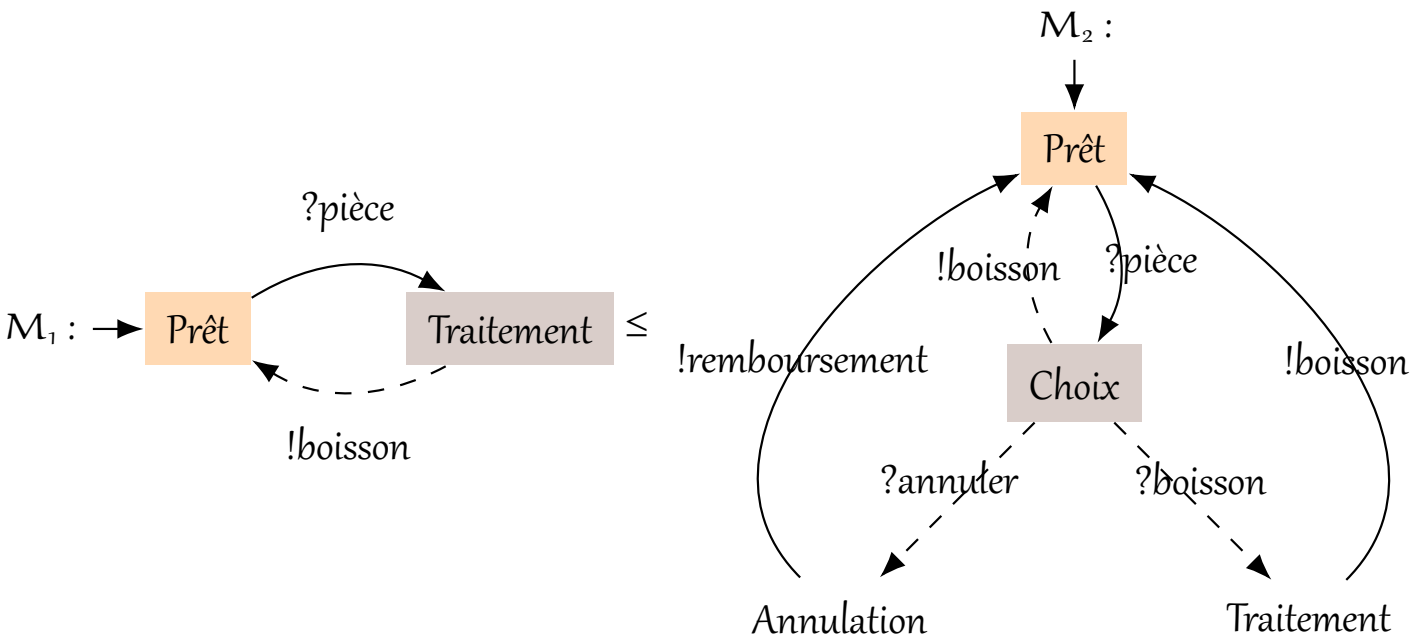
Systèmes de transitions modales

Définition (Systèmes de transitions modales à évènements structurés [Bauer, Juhl, et al., 2012])

Un système de transitions modales à évènements structurés $\langle Q, i, \mathcal{E}, \preccurlyeq, \bot, \mathcal{T}, \alpha, \beta, \lambda, \mu \rangle$ est la donnée :

1. d'un système de transitions $\langle Q, \{i\}, \mathcal{E}, \mathcal{T}, \alpha, \beta, \lambda \rangle$
2. d'un ensemble bien structuré d'étiquettes $\langle \mathcal{E}, \preccurlyeq, \bot \rangle$
3. d'une fonction totale $\mu \in \mathcal{T} \to \wp\{may, must\}$ qui associe à chaque transition $t \in \mathcal{T}$ un ensemble de modalités $\mu(t) \subseteq \{may, must\}$ tel que :
   - $\mu(t)$ est non vide
   - $must \in \mu(t)$ implique $may \in \mu(t)$ □

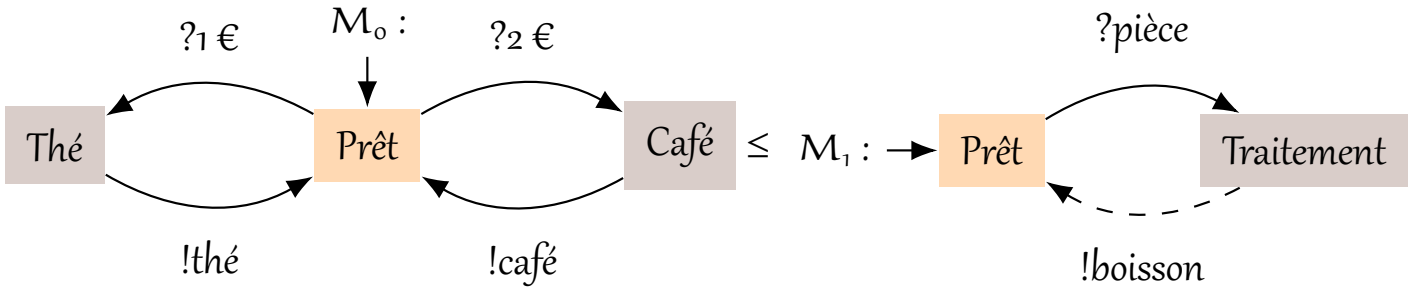
Raffinement



$M_1 \leq M_2$ si et seulement s'il existe une relation entre leurs états telle que :

- Tout $\dashrightarrow$ (donc $\longrightarrow$) de M_1 raffine un $\dashrightarrow$ de M_2 : pas de nouvelles transitions
- Tout $\longrightarrow$ de M_2 est raffiné par un $\longrightarrow$ de M_1 : respect des obligations

Implémentation



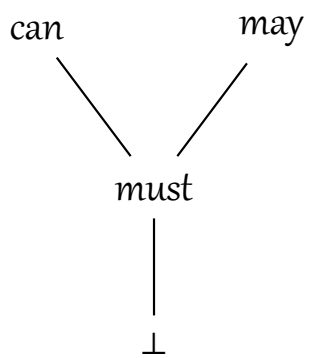
M_0 est une implémentation :

- Toute ses transitions sont obligatoires
- Tous ses évènements sont des évènements d'implémentation

Plan

Plan

De deux modalités à un ensemble de modalités (1/2)



- Modalité may : la transition est permise : elle sera ou ne sera pas implémentée
- Modalité must : la transition est obligatoire : elle sera implémentée
- Modalité can : la transition est possible : il y a moyen de l'implémenter
- must $\preccurlyeq$ may : must implique may : toute transition obligatoire est permise
- must $\preccurlyeq$ can : must implique can : toute transition obligatoire est implémentable
- must : modalité d'implémentation

De deux modalités à un ensemble de modalités (2/2)

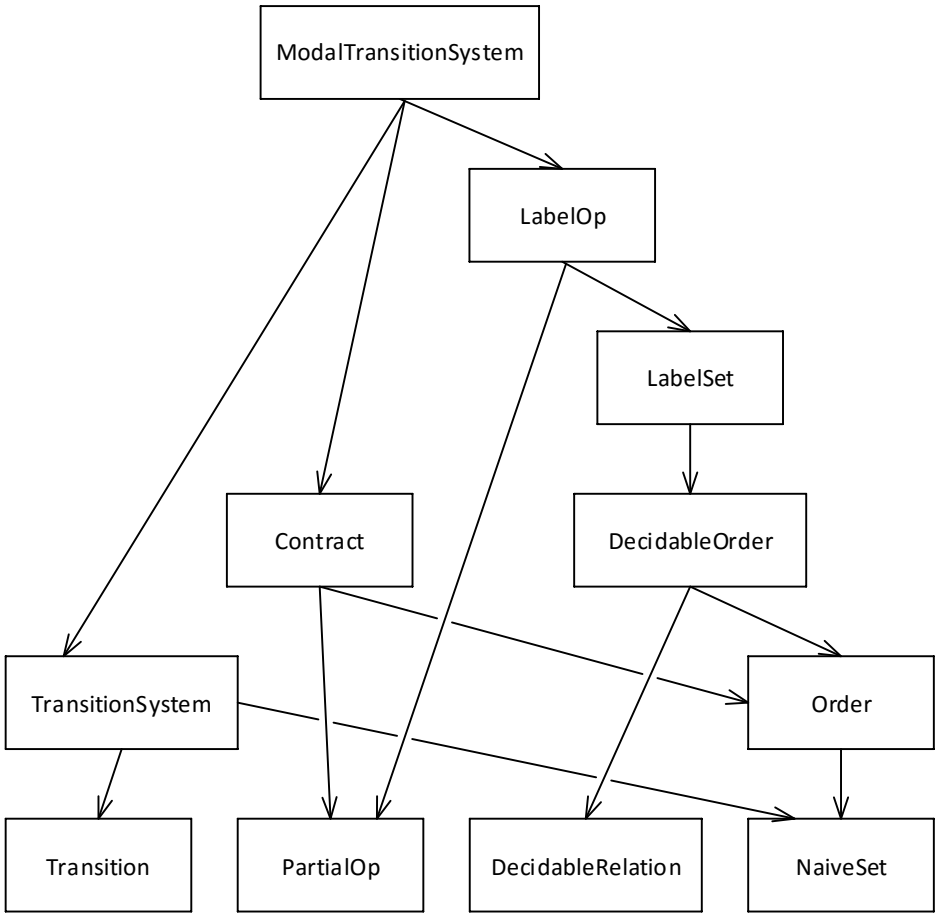
Définition (Systèmes de transitions modales à évènements structurés)

Un système de transitions modales à évènements structurés $\langle Q, i, \mathcal{E}, \preccurlyeq_{\mathcal{E}}, \perp_{\mathcal{E}}, \mathcal{T}, \alpha, \beta, \lambda, \mathcal{M}, \preccurlyeq_{\mathcal{M}}, \perp_{\mathcal{M}}, \mu \rangle$ est la donnée :

1. d'un système de transitions $\langle Q, \{ i \}, \mathcal{E}, \mathcal{T}, \alpha, \beta, \lambda \rangle$
2. d'un ensemble bien structuré d'évènements $\langle \mathcal{E}, \preccurlyeq_{\mathcal{E}}, \perp_{\mathcal{E}} \rangle$
3. d'un ensemble structuré de modalités $\langle \mathcal{M}, \preccurlyeq_{\mathcal{M}}, \perp_{\mathcal{M}} \rangle$
4. d'une fonction totale $\mu \in \mathcal{T} \to \wp(\mathcal{M})$ qui associe à chaque transition $t \in \mathcal{T}$ un ensemble de modalités $\mu(t) \subseteq \mathcal{M}$ tel que :
   - $m \preccurlyeq_{\mathcal{M}} m'$ et $m \in \mu(t)$ impliquent $m' \in \mu(t)$ □

Plan

Architecture



LSMTS
- + Classes
  - DecidableOrder
  - DecidableRelation
  - Order
  - PartialOp
- + Contracts
  - Contract
- + Sets
  - NaiveSet
- + TransitionSystems
  - LabelOp
  - LabelSet
  - ModalTransitionSystem
  - Transition
  - TransitionSystem

Exemple

```
Add LoadPath ".." as LSMTS.
Require Import Coq.Relations.Relation_Definitions.
Require Import LSMTS.Sets.NaiveSet.
Require Import LSMTS.Classes.Order.

Class LSet_LabelSet {label: Type} (precedes: relation label) (bottom: label): Prop :=
    LSet_buildLabelSet {
        LSet_QuasiOrder: QO_QuasiOrder precedes;
        LSet_eq: relation label := QO_eq LSet_QuasiOrder;
        LSet_Equivalence: Equivalence LSet_eq := QO_Equivalence LSet_QuasiOrder;
        LSet_PartialOrder: PO_PartialOrder LSet_eq precedes :=
            QO_PartialOrder LSet_QuasiOrder;
        LSet_bottom_least: PO_least LSet_PartialOrder (Nv_full label) bottom
    }.
```

Plan

Conclusion

- Le quart de [Bauer, Juhl, et al., 2012] a été revu, généralisé, et mécanisé en Coq
- La comparaison entre les définitions de [Bauer, Juhl, et al., 2012] et leur mécanisation en Coq est très intéressante
- Les fameux « il est facile de voir que... » ne sont pas toujours si faciles !

Merci


André Arnold. Systèmes de transitions finis et sémantique des processus communicants. Études et recherches en informatique. Masson 1992.


Albert Benveniste, Werner Damm, Alberto Sangiovanni-Vincentelli, Dejan Nickovic, Roberto Passerone, et Philipp Reinkemeier. Contracts for the Design of Embedded Systems. Part I : Methodology and Use Cases. HAL 2012.


Albert Benveniste, Jean-Baptiste Raclet, Benoît Caillaud, Dejan Nickovic, Roberto Passerone, Alberto Sangiovanni-Vincentelli, Tom Henzinger, et Kim Larsen. Contracts for the Design of Embedded Systems. Part II : Theory. HAL 2012.


Sebastian S. Bauer, Line Juhl, Kim G. Larsen, Axel Legay, et Jirí Srba. Extending Modal Transition Systems with Structured Labels. Dans : Mathematical Structures in Computer Science 22.4 (2012), p. 581-617.


Yves Bertot et Pierre Castéran. Interactive Theorem Proving and Program Development. Coq'Art : The Calculus of Inductive Constructions. Texts in Theoretical Computer Science : An EATCS Series. Springer Berlin Heidelberg 2004.