Thème Sûreté du développement du logiciel (et des systèmes)

Responsable: Mamoun Filali
Présentation : Marc Pantel, Ralph Matthes

- ACADIE : Assistance à la Certification d'Applications Distribuées et Embarquées
  - 15 permanents (13 EC, 2C)
  - 12 doctorants et post-doctorants
  - 2 sites: INPT/N7, UPS

- MACAO : Modèles, Aspects et Composants pour des Architectures à Objets
  - 14 permanents (13 EC, 1IR)
  - 11 doctorants et post-doctorants
  - 3 sites: UPS, UTM

Objectif scientifique

Simplifier la construction de systèmes sûrs

- Modélisation (spécification) formelle des systèmes
- Ingénierie des modèles (MDE)
  - Langages dédiés
  - Transformations entre langages dédiés
- Sémantique, validation et vérification formelles des
  - Modèles de systèmes
  - Transformations des modèles
  - Outils de validation et vérification
- Fondements mathématiques
- Processus de développement adapté

Compétences

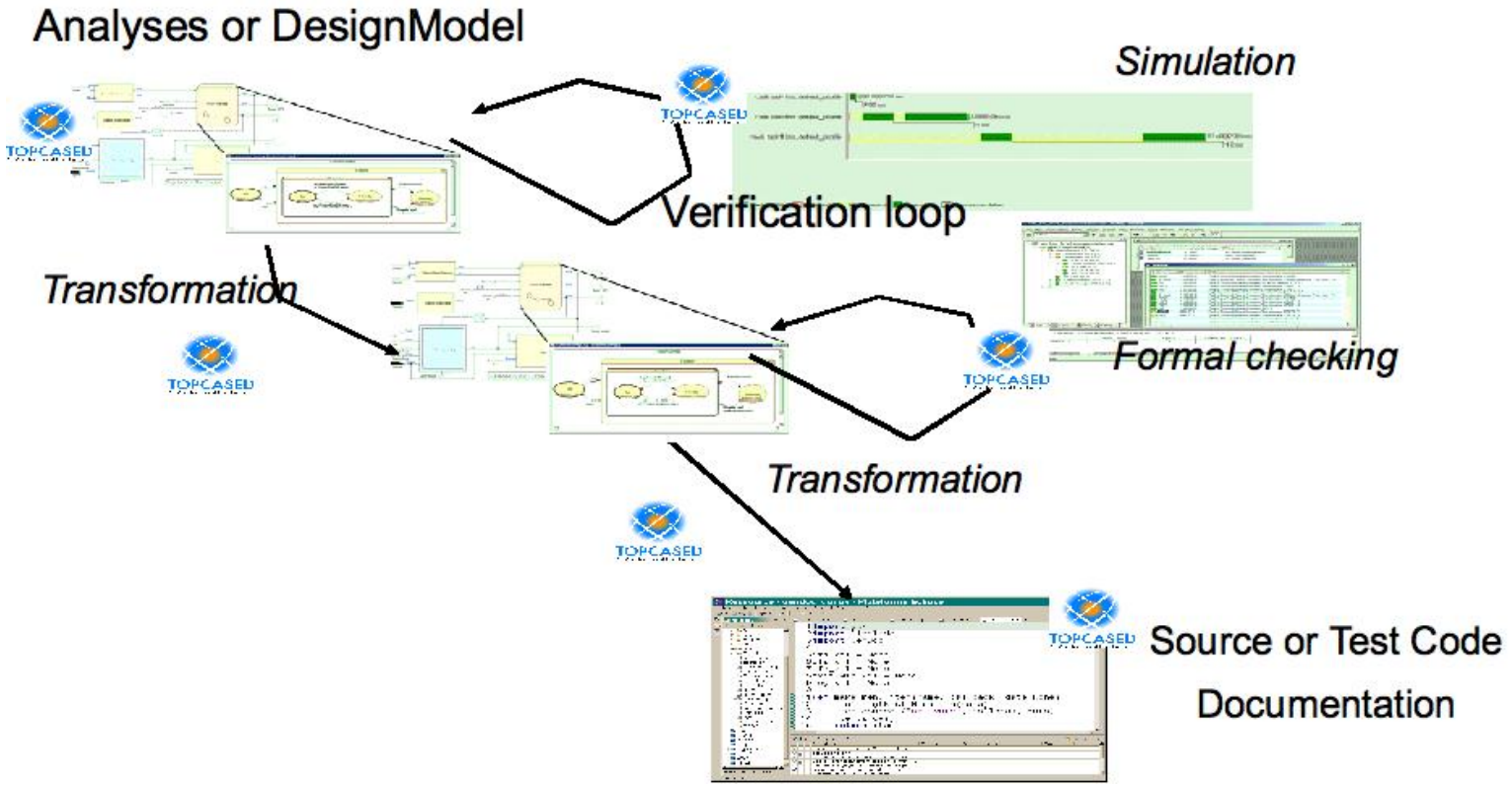
- ACADIE
  - Théorie de la démonstration et du calcul.
  - Méthodes de spécification et vérification formelles.
  - Sémantique des langages et modèles.
  - Analyse statique par résolution de contraintes et interprétation abstraite.
  - Algorithmique répartie
  - Langages d'architecture et Systèmes embarqués.
- MACAO
  - Transformation de modèles.
  - Modèles et langages dédiés.
  - Processus de développement à base de modèles.

Compétences

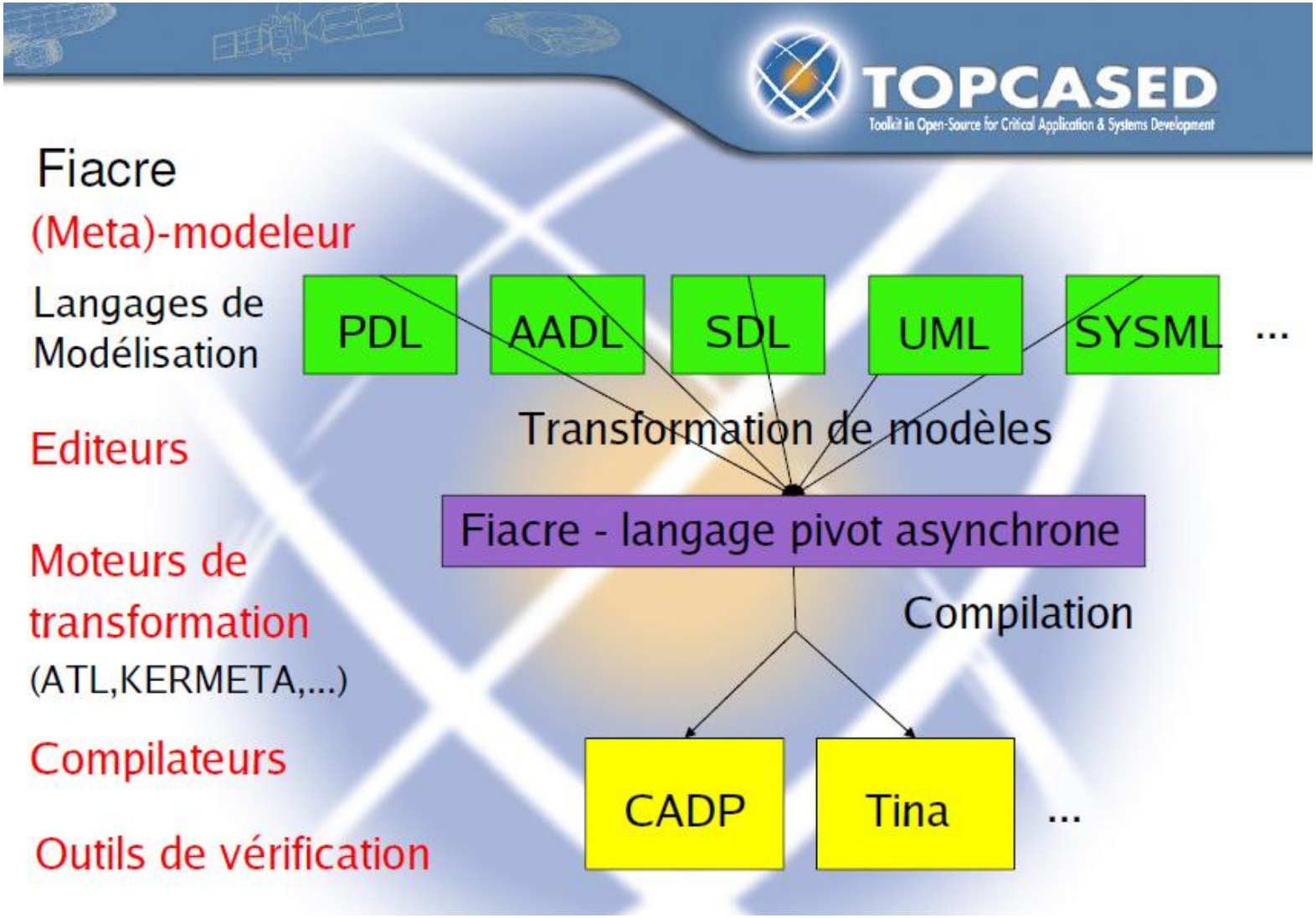
- ACADIE
  - Théorie de la démonstration et du calcul.
  - Méthodes de spécification et vérification formelles.
  - Sémantique des langages et modèles.
  - Analyse statique par résolution de contraintes et interprétation abstraite.
  - Algorithmique répartie
  - Langages d'architecture et Systèmes embarqués.
- MACAO
  - Transformation de modèles.
  - Modèles et langages dédiés.
  - Processus de développement à base de modèles.

Projet FCE TOPCASED (2005-2010)

Participation ACADIE et MACAO
Boîte à outils open source pour le développement de systèmes critiques



Projet FCE TOPCASED (2005-2010)



Le langage FIACRE

Fiacre is an intermediate formal language of the TOPCASED project.

- behavioral aspects.
- timing aspects.
  timed Petri nets based.

Structure:

- Processes: sequential behavior description.
- Components: composition of processes.
- Process algebra based.

Fiacre process

```
process receiver [inp: msg, outp: data,
      ack: none] (expected:bool) is
  states idle, accept, ack
  var data:data, ssn:bool
  from idle
    inp? ssn,data;
    if ssn = expected then to accept
    else to ack end
  from accept
    ssn := not ssn; outp! data; to ack
  from ack ack; to idle
```

Fiacre component

```
component abp [inp: data, outp: data] is
  port timeout in [5,6],
       min, mout: msg in [0,1],
       ain, aout: none in [0,1],
       mloss, aloss: none in [0,1]
  par sender [inp,min,timeout,aout](false)
  ||  mmedium [min, mloss, mout]
  ||  receiver [mout,outp,ain](false)
  ||  amedium [ain, aloss, aout]
  end
```

The Fiacre intermediate language (tools)

3 tools (others to come)

- AADL2FIACRE tool (ACADIE).
  - Translation of a subset of AADL.
- TINA tool set (OLC LAAS).
  - Verification of systems with data, time and priorities.
  - Temporal logic model checker.
- CADP tool set (VASY INRIA).
  - Equivalence checker.
  - Distributed state generator.

- Built with MDE technologies:
  - AADL behaviour annex meta model;
  - FIACRE meta-model.
  - transformation with model to model tool.

Projet ITEA SPICES (2007-2009)

Support for Predictable Integration of mission Critical Embedded Systems.

- Equipes:
  - académique: ACADIE, LAAS, ONERA, CEA, VERIMAG, Univ. Cantabria, Univ. Louvain
  - industriels: AIRBUS, THALES, BARCO, SQS.
- Implication de ACADIE:
  - Formalisation du modèle d'exécution AADL.
  - Traduction AADL-FIACRE

Projet ITEA GeneAuto (2006-2008)

AUTOmated qualified code GENEration for safety critical systems

- Equipes:
  - académique: ACADIE, INRIA, Univ. Tallin
  - industriels: AIRBUS, ALIOSYS, ASTRIUM, BARCO, CONTINENTAL, IAI, KRATES, THALES ALENIA SPACE
- Implication de ACADIE:
  - Formalisation de la sémantique
  - Preuve de correction du générateur
  - Approche pour la qualification

ANR SPACIFY (2007-2010)

Ingénierie des modèles et méthodes formelles intégrées pour le développement des logiciels de vol spatiaux

- Equipes:
  - académique: ACADIE, LABRI, IRISA, ENSTBr.
  - industriels: CNES, THALES, ASTRIUM, TNI.
- Implication de ACADIE:
  - définition d'un langage de transformations destiné à exprimer des raffinements métier qui seront appliqués durant la conception de logiciels de vols jusqu'à obtenir un modèle exécutable.
  - Validation de ces transformations : analyse statique, génération d'obligations de preuve, génération de scripts de preuve.

Projet AIRSYS PAM (2007-2009)

Platform Architecture Management.

- Equipes: ACADIE, ONERA
  Implication de ACADIE:
  - Modèle (méta) de la plateforme avion.
  - Prise en compte des contraintes de sûreté pour l'allocation des ressources de la plateforme aux fonctions avion.

Autres projets

- ANR OpenEmbeDD : ACADIE : IDM et Vérification formelle, AADL.
- ANR DOMINO : MaCAO : Processus de développement à base de modèle.
- ITEA2 ES-PASS : ACADIE : Validation formelle par analyse statique du produit final.
- CNRS TAPIOCA : (2007-2009) ACADIE: langage architecture, AADL.
- FUI SOCKET (SoC toolKit for critical Embedded sysTems) : ACADIE et MACAO : Langages d'architecture, modélisation, vérification.
- RTRA STAE ROSACE : (2007-2010) RObots et Systèmes Auto-adaptatifs Communiquants Embarqués) ACADIE: Langages d'architecture, modélisation, vérification.
- ANR ITEMIS : (2009-2011) Systèmes d'Information Embarqués Intégrés. ACADIE: Modélisation et vérification de services SCA.
- JTI ARTEMIS CESAR : (2009-2011) ACADIE : Prise en compte de la certification et la qualification dans les approches composants, Evolution GeneAuto et FIACRE
- ITEA2 OPEES : (2009-2011) ACADIE: Certification et qualification à base de méthodes formelles. Evolution GeneAuto et FIACRE

Prospectifs

- Many Core Predictible (THALES).
  Implication de ACADIE:
  - Mécanismes de synchronisation et communication entre threads sur des multi-coeurs
  - Compositionnalité, modèles de programmation et génération de code
- QUARTET (FNRAE)
  Langages intermédiaires et transformations qualifiables pour le développement de systèmes temps-réel.
  Implication de ACADIE:
  - Définition d'extensions de FIACRE.
  - Chaîne sûre complète allant des outils de modélisation métier (AADL) aux outils de vérifications (FIACRE).

Thèses récemment soutenues

- Combining temporal and deontic logics for the specification of security policies
  Julien Brunel — Décembre 2007
- Spécification et vérification des ordonnanceurs Temps Réel en B
  Odile Nasr — Janvier 2008
- Analyse statique d'un calcul d'acteurs par interprétation abstraite
  Pierre-Loic Garoche — Juin 2008
- Approche de métamodélisation pour la simulation et la vérification de modèle
  Benoit Combemale — Juillet 2008
- Une approche catégorique unifiée pour la récriture de graphes attribués
  Maxime Rebout — Juillet 2008

Thèses ACADIE en cours

- Commutativité et dépendance des diagrammes catégoriques
- Réductions non-standards dans le système UTT et sous-typage
- Développement d'un noyau sûr de transformations de modèles
- Correction de programmes avec pointeurs, application aux transformations de modèles
- Développement certifié d'un générateur de code pour Simulink/Stateflow
- Sémantique formelle de mécanismes temps réel AADL
- Développement et validation d'architectures dynamiques
- Techniques de test temporisé
- Cohérence temporelle des données dans les systèmes répartis embarqués
- Comportement temporisé des données dans les systèmes temps réel distribués

Sur la recherche “fondamentale” dans le thème

Les préoccupations des permanents et leurs doctorants actuels:

- Théorie de la démonstration et du calcul.
- Récriture de graphes.
- Bases théorique d’assistants de preuve.

Les sujets de thèses en cours sont:

- Commutativité et dépendance des diagrammes catégoriques
- Réductions non-standards dans le système UTT et sous-typage
- Développement d’un noyau sûr de transformations de modèles

On vise à assurer l’applicabilité des résultats obtenus. Si le temps le permet, le doctorant devrait déjà se réjouir d’une application non-triviale et publiable de ses propres résultats. Et certainement des publications sur les résultats de fond.

Sur la “Théorie de la démonstration et du calcul”

Les démonstrations/preuves sont des objets à part entière

- construction (faire la preuve)
- transformation (améliorer la preuve)
- normalisation: vers une forme normale avec de bonnes propriétés

Il y a des liens forts avec la programmation fonctionnelle: l’isomorphisme de Curry et Howard qui se simplifie à

formule = type preuve = terme

Théorie de types: on entrelace les termes et les formules/types. Ça donne des types dépendants. Exemple:

$$\begin{array}{lcl} \phi(0) & := & \text{nat} \quad \text{(les entiers)} \\ \phi(n+1) & := & \phi(n) \Rightarrow \phi(n) \quad \text{(l'espace de fonctions)} \end{array}$$

Avec cela, on peut former le *type* $\phi(3^2)$ dans le langage. Une théorie de types très riche est à la base d’un assistant de preuve comme Coq (INRIA).

Sur la “Récriture de graphes”

C’est un sujet vaste qui est étudié depuis fort longtemps. Encore, on trouve de vraies lacunes de connaissance quand on prend au sérieux l’application des méthodes mathématiques pour le génie logiciel.

Le sujet de thèse “Une approche catégorique unifiée pour la récriture de graphes attribués” (co-encadré par les deux équipes et soutenu en juillet dernier) présente une telle lacune: il y avait la théorie catégorique et des implantations “inspirées” de cette théorie, mais non une théorie assez riche pour les phénomènes de calcul des attributs.

Ce problème est issu des efforts du thème sur la méta-modélisation, et on pouvait prendre du recul pour un résultat fondamental.

Pour les travaux sur les fondements de la méta-modélisation, on cherche la double qualification en mathématiques et en informatique qui sont bien complémentaires – sémantique et syntaxe.

Sur les “Bases théorique d’assistants de preuve”

On trouve une situation privilégiée pour la coopération entre recherche fondamentale et appliquée: il n’y a ni institut ni thème ni équipe d’informatique fondamentale, mais on a une intégration verticale.

Les théoriciens peinent souvent à trouver des directions de généralisation de leurs résultats. Ici, les exigences applicatives dépassent presque naturellement les méta-théorèmes déjà établis. Par exemple, avec les critères de terminaison de récriture et les définitions coinductives acceptés par les assistants de preuve.

On développe un noyau sûr de transformation de modèles, et en même temps on présente des vrais défis à Coq et les autres systèmes, ce qui nous mène à une interaction continue avec les développeurs et à proposer des moyens d’utilisation qui contournent les limites d’expressivité directe et sont considérés comme des contributions à l’état de l’art des théories de types.