Langage intermédiaire et transformations de modèles pour le développement de systèmes temps-réel : retour d'expérience sur la chaîne de vérification formelle Fiacre

Bernard Berthomieu[1,3] — **Jean-Paul Bodeveix**[2,3] — **Silvano Dal Zilio**[1,3] — **Mamoun Filali**[2,3] — **Marc Pantel**[2,3] — **François Vernadat**[1,3]

1 – CNRS ; LAAS ; 7 avenue colonel Roche, F-31077 Toulouse, France
2 – CNRS ; IRIT ; Université de Toulouse, 118 route de Narbonne, F-31062 Toulouse, France
3 – Université de Toulouse ; UPS, INSA, INP, ISAE, UT1, UTM ; F-31062 Toulouse, France

RÉSUMÉ. Nous présentons les résultats obtenus durant le développement d'une chaîne de vérification formelle pour le langage d'architecture AADL *basé sur une approche ingénierie dirigée par les modèles. Notre approche se caractérise par l'utilisation du langage pivot* FIACRE *pour faciliter les activités de vérification et de transformations entre modèles. Nous commentons les premiers retours d'expérience issus de la mise en oeuvre de cette chaîne de vérification et présentons en conclusion les travaux en cours dans le cadre du projet Quarteft qui visent à l'améliorer.*

ABSTRACT. We discuss the results obtained during the development of a formal verification toolchain for AADL *based on a model driven engineering approach. Our approach is characterized by the use of the pivot language* FIACRE *to facilitate verification activities and transformations between models. We quickly analyse the first return on experience and present ongoing work started in the scope of the Quarteft project to improve the verification chain.*

MOTS-CLÉS : Vérification statique de modèles, langages d'architecture

KEYWORDS: Model Checking, Architecture Languages

1. Ce travail est partiellement financé par le pôle de compétitivité Aerospace Valley ; le projet Topcased ; le projet FNRAE Quarteft et la région Midi-Pyrénées.

1. Introduction

L'évolution de la complexité des systèmes embarqués critiques conduit actuellement à exploiter d'une part des langages de modélisation les plus proches possible des spécifications du métier du concepteur, et d'autre part des outils de vérification de modèle (model checker) permettant d'assurer la correction du système par rapport à des exigences de sûreté, de respect de contraintes temps-réel, ...

C'est dans ce contexte qu'a été initié notamment le projet TOPCASED (`http://www.topcased.org`) visant à proposer un atelier de développement de systèmes critiques temps-réel basé sur la vérification formelle et les langages dédiés. Les méthodes et outils issus de ces travaux visent une intégration des activités de vérification formelle au sein du processus et de la plate-forme de développement. Pour factoriser les transformations entre les langages de modélisation « métier » et les dialectes des outils de vérification, le langage intermédiaire FIACRE (Format Intermédiaire pour les Architectures de Composants Répartis Embarqués) a été développé. Il permet de représenter les aspects comportementaux et temporisés de systèmes – en particulier de systèmes embarqués et systèmes distribués – pour leur vérification ou simulation.

Nous présentons les résultats obtenus durant le développement, basé sur une approche ingénierie dirigée par les modèles, de la chaîne de vérification formelle pour le langage d'architecture AADL. Nous commentons les premiers retours d'expérience issus de la mise en oeuvre de cette chaîne de vérification et présentons en conclusion les travaux en cours dans le cadre du projet Quarteft [QFT] qui visent à améliorer cette chaîne.

2. Le langage intermédiaire Fiacre

Les langages de modélisation exploités par les environnements de vérification de modèles sont souvent conçus pour faciliter la vérification, mais n'offrent pas en général les éléments nécessaires à une expression simple et directe des « patrons opérationnels » utilisés dans les systèmes critiques temps-réel. Les solutions proposées consistent en général à traduire directement des langages utilisateurs expressifs (haut-niveau) vers des formalismes (bas-niveau) dédiés à la vérification. Pour réduire la distance entre les langages utilisateurs et les outils de vérification, et réduire la complexité des ces traductions, le langage FIACRE [FAR 08] a été conçu comme un modèle formel intermédiaire entre les langages utilisateurs et les outils de vérification. Il embarque les notions suivantes :

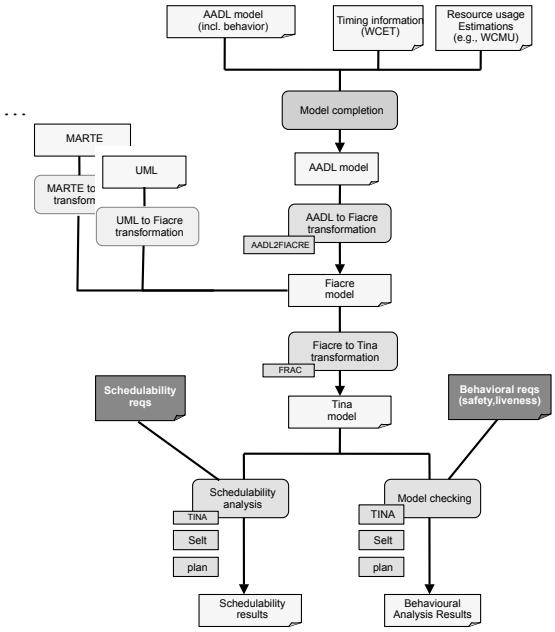
– Des processus décrivant le comportement de composants séquentiels. Un processus est défini comme un ensemble d'états de contrôle, chacun associé à un programme impliquant des constructions déterministes classiques (affectations, conditionnelles, boucles while, composition séquentielles) ; des constructions non-déterministes (choix non-déterministes, affectations non-déterministes) ; des événements d'interactions sur des ports de communication (synchronisation, émission, réception) ; ...

– Des composants décrivant la composition de processus, de manière hiérarchique. Un composant est défini comme une composition parallèle de composants et/ou de processus interagissant par des portes et/ou des variables partagées. Le sous-langage des composants permet aussi de restreindre le mode d'accès et la visibilité des portes et des variables partagées, ainsi que d'associer des contraintes temporelles et de priorité aux interactions.

Le langage FIACRE a été défini sémantiquement et est intégré à deux boîtes à outils logiciels pour la vérification, qui sont TINA [BER 04] et CADP [GAR 07]. Le langage FIACRE est aussi intégré dans l'environnement TOPCASED (un méta-modèle au format ECORE-EMF a été défini).

3. Chaîne de vérification AADL/Fiacre

AADL (the SAE Architecture Analysis and Design Language [SAE]) est un langage de description d'architecture qui permet de décrire autant les aspects matériels que les composants logiciels d'un système. Une extension particulièrement importante à cette norme est l'ajout d'une *annexe comportementale* [FRA 07], qui permet de décrire finement la dynamique des systèmes et qui raffine le concept de fil d'exécution de AADL. La chaîne de transformation utilisée est décrite dans la figure suivante.



FIACRE permet de simplifier grandement la traduction entre une spécification AADL et les outils de vérification [BER 09]. La plupart des mécanismes temps-réel et de communication AADL ont ainsi été traduits en FIACRE en utilisant le langage de

transformation de modèle Kermeta. Cette première étape a donc permis : de préciser la sémantique des mécanismes temps-réel et de communication AADL et d'expérimenter la définition du méta modèle FIACRE sur un exemple de taille industrielle.

Ce travail sur la transformation du langage AADL [PI 09] a montré qu'il était possible de capitaliser les traductions en FIACRE de constructions temps réel. À cet effet, il faudrait disposer d'un format intermédiaire plus orienté temps-réel permettant de factoriser les traductions de ces constructions temps réel vers Fiacre et de simplifier, de facto, la prise en compte de nouveaux langages utilisateurs temps réel.

Cette approche par langages intermédiaires introduit cependant deux inconvénients : la difficulté de remonter à l'utilisateur des résultats de vérification et une possible complexification du modèle final due à l'enchaînement des traductions. Les travaux envisagés dans le cadre du projet Quarteft visent à définir une nouvelle chaîne de vérification incorporant RT-Fiacre, une couche intermédiaire orientée temps réel, tout en réduisant les inconvénients constatés lors des expérimentations de la chaîne AADL/FIACRE.

4. Travaux futurs : Amélioration de la chaîne de vérification

Le projet Quarteft [QFT] a débuté en mai 2009. Il vise à améliorer la chaîne de vérification décrite dans la section 3 sur deux aspects complémentaires : d'une part, définir des extensions au langage FIACRE lui permettant de prendre en compte de façon native de nouveaux concepts de haut niveau, et d'autre part développer les techniques de validation et vérification des transformations de ce langage FIACRE étendu vers FIACRE et les outils de vérification pour disposer d'une chaîne sûre complète allant des outils de modélisation métier aux outils de vérification.

Le retour d'expérience FIACRE, acquis lors des travaux de traduction du langage AADL vers FIACRE et d'évaluation de la chaîne de transformations, a montré que si le langage est assez expressif, il n'est pas toujours suffisamment concis pour pouvoir exprimer simplement un pas de calcul au niveau AADL en un pas de calcul atomique au niveau FIACRE. Ce manque de concision complexifie la traduction de certaines constructions AADL et peut entraîner une augmentation de la taille de l'espace d'états généré à partir de cette traduction. Pour améliorer ce point, nous envisageons d'étendre FIACRE pour permettre les communications bidirectionnelles et les transactions (groupe de communications atomiques). Celles-ci renforceront sa capacité à traiter de façon atomique les événements complexes de communication et permettront de traduire plus simplement et plus efficacement certaines constructions AADL– telles que les changements de mode – ou encore la prise en compte du modèle d'exécution AADL.

Un deuxième aspect concerne l'introduction d'un niveau intermédiaire spécifiquement temps réel – RT-FIACRE– permettant d'une part de factoriser les traductions de constructions temps réel classiques telles que les processus périodiques ou sporadiques, les automates de mode, ... Actuellement, ces différents aspects sont pris en

compte dans la traduction de chaque langage de modélisation visant FIACRE. Ces traductions comportent donc des points communs qu'il serait pertinent de pouvoir factoriser. Les techniques de définition de transformations dépendent beaucoup de la forme des méta-modèles. Il n'est donc pas possible de factoriser des aspects s'ils n'ont pas la même forme dans les différents méta-modèles. Il est donc nécessaire de capturer ces aspects dans un méta-modèle intermédiaire commun pour pouvoir ensuite exprimer une transformation unique depuis ce méta-modèle commun. Ainsi, ce langage intermédiaire de vérification orienté temps-réel offrira de façon native les constructions temps réel les plus habituelles (processus sporadique/périodique, automates de mode, ...) et permettra ainsi d'exprimer de manière plus simple les transformations de langages plus proches de l'utilisateur tels AADL, UML-MARTE, SYSML, SDL, ... Il sera ainsi possible de valider et vérifier à un moindre coût ces transformations. Cela permettra de construire plus simplement des chaînes correctes qualifiables de développement de systèmes critiques.

Un dernier aspect concerne le traitement des propriétés et l'interprétation des retours de vérification. L'approche consiste à traiter de façon symétrique et coordonnée les traductions respectives de la partie comportement et de la partie propriétés et la génération de la « fonction » de traduction inverse qui opérera sur les contre-exemples lors de la phase de retours de vérification. Ainsi, les propriétés du niveau AADL sont traduites pour prendre en compte la traduction du modèle AADL vers le modèle RT-FIACRE. De nouvelles propriétés spécifiques aux constructions AADL peuvent être ajoutées et traduites elles aussi dans les éléments du modèle RT-FIACRE, par exemple le fait qu'un buffer ait une taille limitée ou qu'un processus périodique aura des échéances temporelles à respecter. Cette double traduction comportement/propriétés s'accompagne de la construction de relations qui, en plus de lier les éléments sources de la traduction aux éléments cibles, lient aussi entre-eux les éléments des modèles et des propriétés. Ce deuxième type de relations peut être exploité pour remonter les contre-exemples obtenus sur le langage cible vers les langages sources.

5. Bibliographie

[BER 04] BERTHOMIEU B., RIBET P.-O., VERNADAT F., « The tool TINA – Construction of Abstract State Spaces for Petri Nets and Time Petri Nets », *International Journal of Production Research*, vol. 42-No 14, 2004.

[BER 09] BERTHOMIEU B., BODEVEIX J.-P., CHAUDET C., DAL ZILIO S., FILALI M., VERNADAT F., « Formal Verification of AADL Specifications in the Topcased Environment », *International Conference on Reliable Software Technologies - Ada-Europe*, n° 5570 LNCS, 2009.

[FAR 08] FARAIL P., GAUFILLET P., PERES F., BODEVEIX J.-P., FILALI M., BERTHOMIEU B., SAAD R. T., VERNADAT F., GARAVEL H., LANG F., « FIACRE : an intermediate language for model verification in the TOPCASED environment », *European Congress on Embedded Real-Time Software (ERTS)*, 2008.

[FRA 07] FRANCA R. B., BODEVEIX J.-P., FILALI M., CHEMOUIL D., THOMAS D., « The AADL behaviour annex – experiments and roadmap », *12th IEEE International Conference*

on Engineering Complex Computer Systems (ICECCS), 2007.

[GAR 07] GARAVEL H., MATEESCU R., LANG F., SERWE W., « CADP 2006 : A Toolbox for the Construction and Analysis of Distributed Processes », *CAV*, 2007.

[QFT] Quarteft : Langages intermédiaires et technologies de transformations qualifiables pour le développement de systèmes temps-réel, `http://quarteft.loria.fr`.

[PI 09] PI L., BODEVEIX J.-P., FILALI M., KAI H., DIANFU M., « A comparative study of FIACRE and TASM to define AADL real time concepts », *UML&AADL'2009 –14th IEEE International Conference on Engineering of Complex Computer Systems*, 2009.

[SAE] SAE Aerospace. Architecture Analysis & Design Language (AADL). AS-5506, SAE International, 2004.