Un langage de contexte de preuve pour la validation formelle de modèles logiciels

Philippe Dhaussy[*], Julien Auvray*
Stéphane De Belloy**, Frédéric Boniol***, Eric Landel [* +]

* Laboratoire DTN, ENSIETA, BREST, F-29806 cedex 9
{dhaussy, auvrayju, landeler} @ensieta.fr
http://www.ensieta.fr/dtn
** THALES AIR SYSTEMS, BP 20351 94628 RUNGIS Cedex
stephane.debelloy@fr.thalesgroup.com
*** IRIT-ENSEEIHT, 2 rue C. Camichel BP 7122 – F-31071 Toulouse, cedex 7
frederic.boniol@enseeiht.fr
+ CS-SI, 6, avenue Saint Granier, Toulouse

Résumé. Pour améliorer les pratiques dans le domaine de la validation formelle de modèles, nous explorons un axe de recherche dans lequel nous formalisons la notion de « contexte de preuve » intégrant la description du comportement de l'environnement interagissant avec le modèle et les propriétés à vérifier dans ce contexte. L'article présente le langage CDL (Context Description Language) proposé à l'utilisateur pour la description des contextes de preuve. Ceux-ci sont exploités, actuellement dans nos travaux, par une technique de vérification de type model-checking avec la mise en œuvre d'observateurs. Dans une approche Ingénierie Dirigée par les Modèles (IDM), les modèles de contextes sont transformés en modèles d'automates temporisés puis en codes exploitables par l'outil OBP/IFx (Observer-Based Prover). Ce travail a donné lieu à plusieurs expérimentations industrielles comme la validation formelle d'un protocole de communication avionique pour l'AIRBUS A380. Dans cet article, nous décrivons l'application de notre approche la validation d'un modèle de contrôleur de système aérien conçu par THALES. L'article rend compte de la mise en œuvre du langage CDL et d'un retour d'expérience.

Summary

To improve the practice of formal techniques, we propose a concept of "proof context" for representing the environment behavior interacting with a software model and the properties to be checked in this context. The article presents a language named CDL (Context Description Language) proposed to the user to describe the proof contexts. Those are exploited, currently in our work, by a technique of model checking based on observer automata. In an MDE approach, the proof contexts are translated into timed automata then into codes exploited by the OBP/IFx (Observer-Based Prover) tool for formal analysis. We experienced this approach on some industrial case studies as the formal validation of a AIRBUS-A380 communication protocol. In this paper, we describe an application of our