The BWare Project: Building a Proof Platform for the Automated Verification of B Proof Obligations[*]

David Delahaye[1], Catherine Dubois[2], Claude Marché[3], and David Mentré[4]
(for the BWare project consortium[**])

[1] Cedric/Cnam/Inria, Paris, France
[2] Cedric/ENSIIE/Inria, Évry, France
[3] Inria Saclay - Île-de-France & LRI, CNRS, Univ. Paris-Sud, Orsay, France
[4] Mitsubishi Electric R&D Centre Europe, Rennes, France

Abstract. We introduce BWare, an industrial research project that aims to provide a mechanized framework to support the automated verification of proof obligations coming from the development of industrial applications using the B method and requiring high integrity. The adopted methodology consists in building a generic verification platform relying on different automated theorem provers, such as first order provers and SMT (Satisfiability Modulo Theories) solvers. Beyond the multi-tool aspect of our methodology, the originality of this project also resides in the requirement for the verification tools to produce proof objects, which are to be checked independently. In this paper, we present some preliminary results of BWare, as well as some current major lines of work.

Keywords: B Method, Proof Obligations, First Order Provers, SMT Solvers, Logical Frameworks, Industrial Use, Large Scale Study.

1 Presentation

The BWare project is an industrial research project, funded by the INS ("Ingénierie Numérique & Sécurité") programme of the French National Research Agency (ANR), which aims to provide a mechanized framework to support the automated verification of proof obligations coming from the development of industrial applications using the B method and requiring high integrity. The BWare consortium gathers academic entities, i.e. Cedric, LRI, and Inria, as well as industrial partners, i.e. Mitsubishi Electric R&D Centre Europe, ClearSy, and OCamlPro.

The methodology used in this project consists in building a generic platform of verification relying on different automated theorem provers, such as first order

[*] This work is supported by the BWare project (ANR-12-INSE-0010), funded for 4 years by the INS programme of the French National Research Agency (ANR) and started on September 2012. For more details, see: `http://bware.lri.fr/`.

[**] The BWare project consortium consists of the following partners: Cedric, LRI, Inria, Mitsubishi Electric R&D Centre Europe, ClearSy, and OCamlPro.

Y. Ait Ameur and K.-D. Schewe (Eds.): ABZ 2014, LNCS 8477, pp. 290–293, 2014.

provers and SMT (Satisfiability Modulo Theories) solvers. This generic platform is built upon the Why3 platform [2] for deductive program verification. The considered first order provers are Zenon [4] and iProver Modulo [5], while we opted for the Alt-Ergo SMT solver [1]. The variety of these theorem provers aims to allow a wide panel of proof obligations to be automatically verified by our platform. The major part of the verification tools used in BWare were already involved in some experiments, which consisted in verifying proof obligations or proof rules coming from industrial applications.

Beyond the multi-tool aspect of our methodology, the originality of BWare also resides in the requirement for the verification tools to produce proof objects, which are to be checked independently. To verify these proof objects, we consider two proof checkers: the Coq proof assistant and the Dedukti universal proof checker [3]. These backends should allow us to increase confidence in the proofs produced by the considered automated theorem provers.

To test our platform, a large collection of proof obligations is provided by the industrial partners of the project, which develop tools implementing the B method and applications involving the use of the B method.

2 Preliminary Results

Currently, the BWare platform is already available and works as shown on Fig. 1. The proof obligations are initially produced by Atelier B. They are then translated by a specific tool into Why3 files, which are compatible with a Why3 encoding of the B set theory [8]. From these files, Why3 can produce (by means of appropriate drivers) the proof obligations for the automated theorem provers, using the TPTP format for Zenon and iProver Modulo, and a native format for Alt-Ergo. This translation together with the encoding of the B set theory aims to generate valid statements that are appropriate for the automated theorem provers, i.e. whose proofs can be found by these provers. Finally, once proofs have been found by these tools, some of these provers can generate proof objects to be verified by proof checkers. This is the case of Zenon, which can produce proof objects for Coq and Dedukti [4,7], and iProver Modulo, which can also produce proof objects for Dedukti [6].

In order to assess the BWare platform, two industrial partners of the project provided proof obligations coming from several industrial applications. In particular, Mitsubishi Electric R&D Centre Europe provided the proof obligations of a complete railway level crossing system use case, while ClearSy provided the proof obligations coming from three deployed industrial projects. This constitutes an initial bench of more than 10,500 proof obligations on which we evaluate the BWare platform. The results obtained at the beginning of the project are as follows: the main prover (mp) of Atelier B (4.0) is able to prove 84% of these proof obligations, while Alt-Ergo (0.95.1) obtains a rate of 58%, iProver Modulo (over iProver 0.7) 19%, and Zenon (0.7.2) less than 1%. As can be observed, the first order provers (iProver Modulo and especially Zenon) encounter difficulties, which can be explained by the fact that these provers do not know the B set theory.

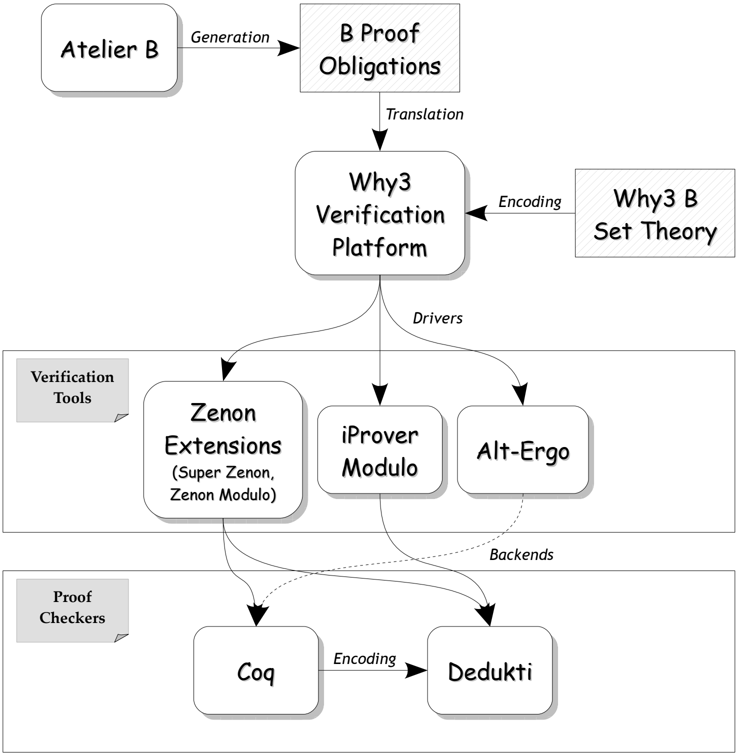

Fig. 1. The BWare Platform for the Automated Verification of B Proof Obligations

Some of the current lines of work of the project therefore focus on extending the first order provers to make them able to reason modulo the B set theory. Regarding SMT solvers, an intermediate set of results obtained with improved versions of Alt-Ergo is given at the OCamlPro blog[1]. These are very promising results: the development version of Alt-Ergo is now able to automatically discharge more than 98% of the proof obligations.

3 Current Lines of Work

The BWare project consists of several tasks, which cannot be exhaustively described in this paper due to space restrictions. We focus on two major current lines of work of the project.

The first current line of work is upstream and consists in completing the axiomatization of the B set theory in Why3 in order to be able to consider all the provided proof obligations. This is mainly carried out according to what is described in [8], i.e. by adding B constructs to the axiomatization and modifying

[1] Available at the following address: `http://www.ocamlpro.com/blog/2013/10/22/alt-ergo-evaluation-october-2013.html` .

accordingly the translator of proof obligations from Atelier B to Why3. This line of work will allow us to consider a broad scope of proof obligations related to different application domains and test the scalability of our platform as well.

A second current line of work focuses on the first order provers to make them able to reason modulo the B set theory. To do so, we rely on deduction modulo. The theory of deduction modulo is an extension of predicate calculus, which allows us to rewrite terms as well as propositions, and which is well suited for proof search in axiomatic theories, as it turns axioms into rewrite rules. Both first order provers considered in the project have been extended to deduction modulo to obtain Zenon Modulo [7] and iProver Modulo [5]. Both tools have also been extended to produce Dedukti proofs [7,6], which is natural as Dedukti relies on deduction modulo as well. Currently, most of the efforts in this line of work consist in building a B set theory modulo, which is appropriate for automated deduction and keeps some properties such as cut-free completeness.

In the longer term and among the tasks of the project, we plan to do a more extensive benchmarking of the different provers of the project in order to determine which proof coverage ratio we can obtain from our platform (in particular, after the development of the several extensions of the provers). Ultimately, we intend to disseminate and exploit the results of our project by integrating our platform into Atelier B and therefore realizing a multi-prover output of Atelier B.

References

1. Bobot, F., Conchon, S., Contejean, V., Iguernelala, M., Lescuyer, S., Mebsout, A.: *Alt-Ergo, version 0.95.2*. CNRS, Inria, and Université Paris-Sud (2013), http://alt-ergo.lri.fr
2. Bobot, F., Filliâtre, J.-C., Marché, C., Paskevich, A.: Why3: Shepherd Your Herd of Provers. In: Leino, K.R.M., Moskal, M. (eds.) International Workshop on Intermediate Verification Languages, Boogie, pp. 53–64 (2011)
3. Boespflug, M., Carbonneaux, Q., Hermant, O.: The $\lambda\Pi$-Calculus Modulo as a Universal Proof Language. In: Pichardie, D., Weber, T. (eds.) Proof Exchange for Theorem Proving, PxTP, vol. 878, pp. 28–43. CEUR Workshop Proceedings (2012)
4. Bonichon, R., Delahaye, D., Doligez, D.: Zenon: An Extensible Automated Theorem Prover Producing Checkable Proofs. In: Dershowitz, N., Voronkov, A. (eds.) LPAR 2007. LNCS (LNAI), vol. 4790, pp. 151–165. Springer, Heidelberg (2007)
5. Burel, G.: Experimenting with Deduction Modulo. In: Bjørner, N., Sofronie-Stokkermans, V. (eds.) CADE 2011. LNCS (LNAI), vol. 6803, pp. 162–176. Springer, Heidelberg (2011)
6. Burel, G.: A Shallow Embedding of Resolution and Superposition Proofs into the $\lambda\Pi$-Calculus Modulo. In: Blanchette, J.C., Urban, J. (eds.) Proof Exchange for Theorem Proving (PxTP). EPiC, vol. 14, pp. 43–57. EasyChair (2013)
7. Delahaye, D., Doligez, D., Gilbert, F., Halmagrand, P., Hermant, O.: Zenon Modulo: When Achilles Outruns the Tortoise Using Deduction Modulo. In: McMillan, K., Middeldorp, A., Voronkov, A. (eds.) LPAR-19. LNCS, vol. 8312, pp. 274–290. Springer, Heidelberg (2013)
8. Mentré, D., Marché, C., Filliâtre, J.-C., Asuka, M.: Discharging Proof Obligations from Atelier B Using Multiple Automated Provers. In: Derrick, J., Fitzgerald, J., Gnesi, S., Khurshid, S., Leuschel, M., Reeves, S., Riccobene, E. (eds.) ABZ 2012. LNCS, vol. 7316, pp. 238–251. Springer, Heidelberg (2012)