

The History and Evolution of B and Event-B

Philipp Körner¹  , Sebastian Krings¹ , Michael Butler² , Thierry Lecomte³, Michael Leuschel¹ , Luis-Fernando Mejia⁴, and Laurent Voisin⁵ 

¹ Institut für Informatik, Universität Düsseldorf
Universitätsstr. 1, D-40225 Düsseldorf, Germany
`{p.koerner,sebastian.krings,leuschel}@hhu.de`

² University of Southampton
University Road, Southampton, SO17 1BJ, UK
`mjb@ecs.soton.ac.uk`

³ CLEARSY
320 avenue Archimède, 13100 Aix en Provence, France
`thierry.lecomte@clearsy.com`

⁴ Alstom Transportation Systems
48 rue Albert Dhalenne, 93400 Saint-Ouen, France
`luis-fernando.mejia@alstomgroup.com`

⁵ Systereel
1090 rue René Descartes, 13100 Aix-en-Provence, France
`laurent.voisin@systereel.fr`

Abstract of HFM 2019 Talk

The B method for software and systems development together with the specification language B and its successor Event-B offer a rich history. Method, language and tools have been influenced by research stemming from other communities, but also have driven research in formal methods themselves. At the same time, the B method has been successfully used in industry, in particular in the railway domain.

B has originally been developed as a successor to Z by Jean-Raymond Abrial in the 1990s, focusing on two key concepts: using refinement to gradually develop models and tool support for proof and model checking. At the time of writing, three classes of industrial applications of B have been established, which evolved from the original ideas:

- B for software (classical B) [1]: refine specifications until B0, a low-level subset of B, is reached and apply code generators
- B for system modelling (Event-B) [2]: verify critical properties, understand why a system is correct
- B for data validation: express properties in B and check data (possibly using a second tool-chain)

In our talk, we will first give a primer on B and Event-B, introducing the main language features and how they are used. Afterwards, we will describe the history of B, starting with B's genesis as a tool for software validation [3,4,5], discussing industrial applications of B in projects such as train speed control [6] and

signalling [7] and other projects with RATP and SNCF performed by Alstom, Line 14 (Meteor) [8] or Canarsie [9].

Following, we will focus on the evolution of B into Event-B and from software to systems modelling, again focusing on industrial applications such as the flushing line NY [10], OCTYS [11], GIK/Railground [12], the HL3 standard [13] and cooperations with Peugeot. Additionally, we will discuss ventures of using B in other domains such as smart cards [14,15].

The latest language evolution, B for data validation, will again highlight B's prevalence in the railway domain, discussing its use for Paris Line 1 [16] and the (subway) trains in Barcelona, Amsterdam, Calcutta, Cairo, Singapore and many more locations.

Additionally, we will briefly present other data validation projects and how they influenced language and tool evolution, e.g., projects by RATP using Ovado with predicateB as first and PROB as secondary tool chain [17,18] and projects by Alstom using B for their URBALIS 400 CBTC system in 2014 using a tool based on PROB called DTVT developed by CLEARSY for various lines, e.g., in Mexico, Toronto, São Paulo and Panama [19]. While discussing B for data validation, we will also take a brief glance at minor language evolutions, where some parties extended the B language in order to increase usability and flexibility.

Language evolution aside, we want to discuss tool evolution in the B ecosystem. Both B and Event-B are supported by a range of tools, from provers to animators to model checkers. We want to give an overview over the B-method tools currently in use and their development and history, such as the B-Toolkit [20], Atelier-B [4], PROB [21,22] as well as to Rodin [23]. As not all tools are still available, we will also honorably mention the ones that disappeared or never really appeared.

In addition to industrial success stories, the academic reception of the B-method and its tools is notable as well and will be a distinct part of the talk. Starting with the B User Workshop, to the ZB conference and further to the ABZ conference series, which brings together researchers working on different specification languages.

Switching from history and evolution to outlook, we want to discuss new language features such as extensions and customisations on top of classical B and Event-B as understood by Rodin or PROB. Furthermore, we intend to discuss new areas of application both for B as a language as well as for the B-method tools.

Acknowledgements We would like to express our gratitude to Jean-Raymond Abrial, who provided us with sources, discussions, insider information and knowledge from his personal experiences developing B and Event-B.

References

1. Jean-Raymond Abrial. *The B-Book*. Cambridge University Press, 1996.
2. Jean-Raymond Abrial. *Modeling in Event-B: System and Software Engineering*. Cambridge University Press, 2010.
3. J. R. Abrial, M. K. O. Lee, D. S. Neilson, P. N. Scharbach, and I. H. Sørensen. The B-method. In Søren Prehn and Hans Toetenel, editors, *Proceedings VDM*, pages 398–405. Springer, 1991.
4. ClearSy. *Atelier B, User and Reference Manuals*. Aix-en-Provence, France, 2009. Available at <http://www.atelierb.eu/>.
5. J. R. Abrial. The B tool (Abstract). In Robin E. Bloomfield, Lynn S. Marshall, and Roger B. Jones, editors, *Proceedings VDM*, pages 86–87. Springer, 1988.
6. M. Carnot, C. DaSilva, B. Dehbonei, and F. Mejia. Error-free software development for critical systems using the B-Methodology. In *Proceedings ISSRE*, pages 274–281. IEEE, 1992.
7. Babak Dehbonei and Fernando Mejia. Formal methods in the railways signalling industry. In *Proceedings FME*, volume 873 of *LNCS*, pages 26–34. Springer, 1994.
8. Daniel Dollé, Didier Essamé, and Jérôme Falampin. B dans le transport ferroviaire. L’expérience de Siemens Transportation Systems. *Technique et Science Informatiques*, 22(1):11–32, 2003.
9. Didier Essamé and Daniel Dollé. B in Large-Scale Projects: The Canarsie Line CBTC Experience. In *Proceedings B*, volume 4355 of *LNCS*, pages 252–254. Springer, 2007.
10. Denis Sabatier. Using Formal Proof and B Method at System Level for Industrial Projects. In *Proceedings RSSRail*, volume 9707 of *LNCS*, pages 20–31. Springer, 2016.
11. Mathieu Comptier, David Déharbe, Julien Molinero Perez, Louis Mussat, Pierre Thibaut, and Denis Sabatier. Safety Analysis of a CBTC System: A Rigorous Approach with Event-B. In *Proceedings RSSRail*, volume 10598 of *LNCS*, pages 148–159. Springer, 2017.
12. Michael J. Butler, Dana Dghaym, Tomas Fischer, Thai Son Hoang, Klaus Reichl, Colin F. Snook, and Peter Tummelshammer. Formal Modelling Techniques for Efficient Development of Railway Control Products. In *Proceedings RSSRail*, volume 10598 of *LNCS*, pages 71–86. Springer, 2017.
13. Dominik Hansen, Michael Leuschel, David Schneider, Sebastian Krings, Philipp Körner, Thomas Naulin, Nader Nayeri, and Frank Skowron. Using a Formal B Model at Runtime in a Demonstration of the ETCS Hybrid Level 3 Concept with Real Trains. In *Proceedings ABZ*, volume 10817 of *LNCS*, pages 292–306. Springer, 2018.
14. J-L Lanet. The use of B for Smart Card. In *Proceedings FDL vol.2*, 2002.
15. Marc Benveniste. On Using B in the Design of Secure Micro-controllers: An Experience Report. *ENTCS*, 280:3–22, 2011.
16. Michael Leuschel, Jérôme Falampin, Fabian Fritz, and Daniel Plagge. Automated Property Verification for Large Scale B Models. In *Proceedings FM*, volume 5850 of *LNCS*, pages 708–723. Springer, 2009.
17. Robert Abo and Laurent Voisin. Formal implementation of data validation for railway safety-related systems with OVADO. In *Proceedings SEFM 2013*, volume 8368 of *LNCS*, pages 221–236. Springer, 2014.
18. Frédéric Badeau and Marielle Doche-Petit. Formal data validation with Event-B. *CoRR*, abs/1210.7039, 2012. Proceedings of DS-Event-B 2012, Kyoto.

19. Thierry Lecomte, Lilian Burdy, and Michael Leuschel. Formally Checking Large Data Sets in the Railways. *CoRR*, abs/1210.6815, 2012. Proceedings of DS-Event-B.
20. Matthew Lee and Ib. H. Sørensen. B-tool. In *Proceedings VDM*, volume 551 of *LNCS*, pages 695–696. Springer, 1991.
21. Michael Leuschel and Michael J. Butler. ProB: an automated analysis toolset for the B method. *STTT*, 10(2):185–203, 2008.
22. Michael Leuschel, Jens Bendisposto, Ivaylo Dobrikov, Sebastian Krings, and Daniel Plagge. From Animation to Data Validation: The ProB Constraint Solver 10 Years On. In *Formal Methods Applied to Complex Systems: Implementation of the B Method*, chapter 14, pages 427–446. Wiley ISTE, 2014.
23. Jean-Raymond Abrial, Michael Butler, Stefan Hallerstede, and Laurent Voisin. An Open Extensible Tool Environment for Event-B. In *Proceedings ICFEM*, volume 4260 of *LNCS*, pages 588–605. Springer, 2006.