

The History and Evolution of B and Event-B

Philipp Körner, Sebastian Krings,
Michael Butler, Thierry Lecomte, Michael Leuschel,
Luis-Fernando Mejia, Laurent Voisin

This is a pre-print version archived by P. Körner at
<https://pkoerner.github.io/pages-output/publications/>.

The final authenticated version is available online at
<https://doi.org/10.1007/978-3-030-54997-8>.

The History and Evolution of B and Event-B

Michael Butler², Philipp Körner¹, Sebastian Krings¹, Thierry Lecomte³,
Michael Leuschel¹, and Laurent Voison⁴

¹ Institut für Informatik, Universität Düsseldorf
Universitätsstr. 1, D-40225 Düsseldorf
{koerner,krings,leuschel}@cs.uni-duesseldorf.de

² University of Southampton

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

⁴ Systerel

1 Abstract

The specification language B and its successor Event-B are used for the formal development of software and systems following the correct-by-construction approach. Both are based on first order predicate logic with higher-order sets, relations, functions.

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. There are three classes of industrial applications of B:

- 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 double chain)

In our article, 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], discussing industrial applications of B in projects such as Line 14 (Meteor) [5] or Canarsie [6].

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 [7], OCTYS [8], GIK/Railground [9] as well as the HL3 standard [10] and cooperations with Peugeot. Additionally, we will discuss ventures of using B in other domains such as smart cards [11, 12].

The latest language evolution, B for data validation will be presented by discussing its use for Paris Line 1 [13] and the (subway) trains in Barcelona, Amsterdam, Singapore and many more locations.

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. As not all tools are still available, we will also discuss the ones that disappeared or never really appeared.

In addition to the 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 article. Starting with the B User Workshop, to the ZB conference and further to the ABZ conference series, which brings together researchers working on different specifications languages. Additionally, while Rodin is not (yet) very successful when it comes to industrial applications, it has allowed for a lot of academic experimentation.

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

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. France Steria, Aix-en-Provence. *Atelier B, User and Reference Manuals*, 1996. Available at <http://www.atelierb.societe.com>.
4. J. R. Abrial. The B tool (Abstract). In Robin E. Bloomfield, Lynn S. Marshall, and Roger B. Jones, editors, *VDM '88 VDM — The Way Ahead*, pages 86–87, Berlin, Heidelberg, 1988. Springer Berlin Heidelberg.
5. 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.
6. Didier Essamé and Daniel Dollé. B in Large-Scale Projects: The Canarsie Line CBTC Experience. In Jacques Julliand and Olga Kouchnarenko, editors, *Proceedings B'2007*, LNCS 4355, pages 252–254. Springer-Verlag, 2007.
7. Denis Sabatier. Using Formal Proof and B Method at System Level for Industrial Projects. In *Proceedings RSSRail 2016*, pages 20–31, 2016.
8. 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 2017*, pages 148–159, 2017.
9. Michael J. Butler, Dana Dghaym, Tomas Fischer, Thai Son Hoang, Klaus Reichl, Colin F. Snook, and Peter Tummeltshammer. Formal Modelling Techniques for Efficient Development of Railway Control Products. In *Reliability, Safety, and Security of Railway Systems. Modelling, Analysis, Verification, and Certification - Second International Conference, RSSRail 2017, Pistoia, Italy, November 14-16, 2017, Proceedings*, pages 71–86, 2017.
10. 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 *Abstract State Machines, Alloy, B, TLA, VDM, and Z - 6th International Conference, ABZ 2018, Southampton, UK, June 5-8, 2018, Proceedings*, pages 292–306, 2018.
11. J-L Lanet. The use of B for Smart Card. In *Forum on Design Languages (FDL02)*, September 2002.
 12. Marc Benveniste. On Using B in the Design of Secure Micro-controllers: An Experience Report. *Electronic Notes in Theoretical Computer Science*, 280:322, 12 2011.
 13. Michael Leuschel, Jérôme Falampin, Fabian Fritz, and Daniel Plagge. Automated Property Verification for Large Scale B Models. In A. Cavalcanti and D. Dams, editors, *Proceedings FM 2009*, LNCS 5850, pages 708–723. Springer-Verlag, 2009.