The dawn of formalized mathematics

Andrej Bauer
University of Ljubljana
`Andrej.Bauer@andrej.com`

When I was a student of mathematics I was told that someone had formalized an entire book on analysis just to put to rest the question whether mathematics could be completely formalized, so that mathematicians could proceed with business as usual. I subsequently learned that the book was Landau's "Grundlagen" [6], the someone was L. S. van Benthem Jutting [11], the tool of choice was Automath [12], and that popular accounts of history are rarely correct.

Formalized mathematics did not die out. Computer scientists spent many years developing proof assistants [1, 10, 8, 3] – programs that help create and verify formal proofs and constructions – until they became good enough to attract the attention of mathematicians who felt that formalization had a place in mathematical practice. The initial successes came slowly and took a great deal of effort. In the last decade, the complete formalizations of the odd-order theorem [4] and the solution of Kepler's conjecture [5] sparked an interest and provided further evidence of viability of formalized mathematics. The essential role of proof assistants in the development of homotopy type theory [9] and univalent mathematics [13] showed that formalization can be an inspiration rather than an afterthought to traditional mathematics. Today the community gathered around Lean [3], the newcomer among proof assistants, has tens of thousands of members and is growing very rapidly thanks to the miracle of social networking. The new generation is ushering in a new era of mathematics.

Formalized mathematics, in tandem with other forms of computerized mathematics [2], provides better management of mathematical knowledge, an opportunity to carry out ever more complex and larger projects, and hitherto unseen levels of precision. However, its transformative power runs still deeper. The practice of formalization teaches us that formal constructions and proofs are much more than pointless transliteration of mathematical ideas into dry symbolic form. Formal proofs have rich structure, worthy of attention by a mathematician as well as a logician; contrary to popular belief, they can directly and elegantly express mathematical insights and ideas; and by striving to make them slicker and more elegant, new mathematics can be discovered.

Formalized mathematics is changing the role of foundations of mathematics, too. A good century ago, a philosophical crisis necessitated the development of logic and set theory, which served as the bedrock upon which the 20th century mathematics was built safely. However, most proof assistants shun logic and set theory in favor of type theory, the original resolution of the crisis given by Bertrand Russell [14] and reformulated into its modern form by Per Martin-Löf [7]. The reasons for this phenomenon are yet to be fully understood, but we can speculate that type theory captures mathematical practice more faithfully because it directly expresses the structure and constructions of mathematical objects, whereas set theory provides plentiful raw material with little guidance on how mathematical objects are to be molded out if it.

References

[1] S.F. Allen, M. Bickford, R.L. Constable, R. Eaton, C. Kreitz, L. Lorigo, and E. Moran. Innovations in computational type theory using Nuprl. *Journal of Applied Logic*, 4(4):428–469, 2006.

[2] Jacques Carette, William M. Farmer, Michael Kohlhase, and Florian Rabe. Big math and the one-brain barrier – the tetrapod model of mathematical knowledge. *Mathematical Intelligencer*, 43(1):78–87, 2021.

[3] Leo de Moura, Sebastian Ullrich, and Dany Fabian. Lean theorem prover.

[4] Georges Gonthier, Andrea Asperti, Jeremy Avigad, Yves Bertot, Cyril Cohen, François Garillot, Stéphane Le Roux, Assia Mahboubi, Russell O'Connor, Sidi Ould Biha, Ioana Pasca, Laurence Rideau, Alexey Solovyev, Enrico Tassi, and Laurent Théry. A machine-checked proof of the odd order theorem. In Sandrine Blazy, Christine Paulin-Mohring, and David Pichardie, editors, *Interactive Theorem Proving - 4th International Conference, ITP 2013, Rennes, France, July 22-26, 2013. Proceedings*, volume 7998 of *Lecture Notes in Computer Science*, pages 163–179. Springer, 2013.

[5] Thomas Hales, Mark Adams, Gertrud Bauer, Tat Dat Dang, John Harrison, Le Truong Hoang, Cezary Kaliszyk, Victor Magron, Sean McLaughlin, Tat Thang Nguyen, Quang Truong Nguyen, Tobias Nipkow, Steven Obua, Joseph Pleso, Jason Rute, Alexey Solovyev, Thi Hoai An Ta, Nam Trung Tran, Thi Diep Trieu, Josef Urban, Ky Vu, and Roland Zumkeller. A formal proof of the Kepler conjecture. *Forum of Mathematics, Pi*, 5, 2017.

[6] E. G. H. Y. Landau. *Grundlagen der Analysis.* Chelsea Pub. Co., New York, NY, USA, 1965.

[7] Per Martin-Löf. An intuitionistic theory of types. In Giovanni Sambin and Jan M. Smith, editors, *Twenty-five years of constructive type theory (Venice, 1995)*, volume 36 of *Oxford Logic Guides*, pages 127–172. Oxford University Press, 1998.

[8] Ulf Norell. *Towards a practical programming language based on dependent type theory.* PhD thesis, Chalmers University of Technology, 2007.

[9] The Univalent Foundations Program. *Homotopy Type Theory: Univalent Foundations for Mathematics* , Institute for Advanced Study, 2013.

[10] Coq Development Team. The Coq proof assistant reference manual, version 8.7.

[11] L. S. van Benthem Jutting. Checking Landau's "Grundlagen" in the Automath system. In *Mathematical Centre Tracts*, volume 83. Mathematisch Centrum, The Netherlands, 1979.

[12] D. T. van Daalen. A description of automath and some aspects of its language theory. In *Selected Papers on Automath*, pages 101–126. North-Holland Pub. Co., Amsterdam, The Netherlands, 1994.

[13] Vladimir Voevodsky, Benedikt Ahrens, Daniel Grayson, et al. *UniMath*: Univalent Mathematics, 2016.

[14] Alfred North Whitehead and Bertrand Russell. *Principia Mathematica.* Cambridge University Press, 1925–1927.