

CURRICULUM VITAE

Nom : TASSI

Last name

Prénom : Enrico

First name

Adresse : Inria Sophia Antipolis, 2004 route des Lucioles – BP 93, 06902, France

Addres

Contacte : Enrico.Tassi@Inria.fr

Contact

Date : 13th May 2025

Date

1) Parcours Professionnel / *Professional history*

Date début <i>Start</i>	Date fin <i>End</i>	Établissement <i>Institutions</i>	Fonction et statut ³ <i>Positions and status⁴</i>
01/01/2025	...	Inria Sophia (STAMP)	CRHC
01/09/2017	01/01/2025	Inria Sophia (Marelle/STAMP)	CRCN
01/09/2014	31/08/2017	Inria Sophia (Marelle)	CR1
01/09/2013	31/08/2014	Inria Saclay (SpecFun)	CR1
01/09/2012	31/08/2013	Inria Saclay (SpecFun)	CR2
01/11/2011	31/08/2012	Inria Saclay (Typical)	Post-doc
01/11/2009	31/10/2011	Microsoft Research-INRIA Joint Center, Paris	Post-doc
01/01/2009	31/10/2009	University of Padua	Post-doc
20/12/2007	20/12/2008	University of Bologna	Post-doc
01/01/2005	19/12/2007	University of Bologna	PhD

2) Encadrement d'étudiants et de jeunes chercheurs / *Supervision of students and early-stage researchers*

Phd thesis:

- Davide Fissore: "Elaboration in Type Theory via Logic Programming", started in 2023, UCA-DS4H, 95% supervision (5% by Yves Bertot — HDR). Davide continues the work from his master's thesis by designing and implementing a new, improved elaborator for the Coq system. He is specifically focusing on enhancing the resolution of type classes in the Iris Coq library. Iris is a key library for the research community interested in applying formal proof techniques to imperative programming languages. The performance and ergonomics (including error detection and reporting) of type class resolution are critical aspects for the users of Iris.

Post-doc:

- Paolo Torrini 2023-2024 on the [CoREACT ANR project](#), 50% supervision with Cyril Cohen. Paolo added to the Hierarchy-Builder tool the capability of mixing interfaces on related, but different, subjects. Paolo also used this new feature of Hierarchy-Builder to formalize double categories.
- Cvetan Dunchev 2015, 20% supervision (80% by Claudio Sacerdoti Coen). Cvetan worked on the implementation of Elpi and in particular he designed a more efficient term representation and an indexing data structure based on term-hashes. He spent the summer in Sophia-Antipolis under my direct supervision to finalize the implementation of the indexing facility.
- François Poulain 2013-2014 on the [DoCoq DIGITEO project](#), 50% supervision with Joris van der Hoeven. François worked on interfacing the Coq system with the TeXmacs document editor in order to experiment with the rendering and edition of bi-dimensional mathematical notations (e.g. fractions, iterated sums or integrals).
- Karst Tankink 2013-2014 on the [ParallTP ANR project](#), 100% supervision. Karst developed a PIDE backedn for Coq, making it possible to use Coq in conjunction with the Jedit editor and the Eclipse IDE via the dedicated Coqoon plugin.

Master thesis:

- Hjalte Dalland, Jakob Israelsen and Simon Kristensen: "Expanding Coq with Type Aware Code Completion", 2023, ITU Copenhagen, joint supervision with Jesper Bentson. Hjalte, Jakob and Simon developed theorem-name completion algorithms to be used in the VSCoq2 user interface. I supervised their work in accessing Coq's internals to find and filter relevant completion candidates. Their code has been integrated in VSCoq2 by Romain Tetley. ([manuscript](#))
- Davide Fissore: "Type-class solver in Type Theory via Logic Programming", 2023, UCA-DS4H, full supervision. Davide studied the state of the art type-class solver for Coq and implemented a new prototype using Coq-Elpi improving some aspects of the current Coq solver. ([manuscript](#))

Internships:

- Luc Chabassier: "Program synthesis from Coq data type declarations", École Normale Supérieure (rue d'Ulm, Paris), 2017, 100% supervision. Luc used an early prototype of Coq-Elpi to synthesize automatically programs out of data type declarations. In particular he worked on the systesys of equality tests and their correctness proof.

3) Encadrement de développements technologiques / *Supervision of technological development*

Engineers:

- Romain Tetley: “VSCoq2”, 2023 — ongoing, SED, (joint supervision with Maxime Dénès during 2023)
Romain was involved in the complete rewrite of the VSCoq extension for VSCode together with Maxime Dénès. This effort is meant to continue for a few years and provide a modern and stable user interface for Coq. Together with Maxime Dénès we worked in regular sprints since February 2023, helping Romain dive into the Coq language server.
<https://github.com/coq-community/vscoq>
- Thomas Portet: “Hierarchy Builder Instance Saturation”, 2023, SED – LIBERABACI, joint supervision with Yves Bertot.
Thomas improved the ergonomic of Hierarchy Builder by making the order in which certain commands are given irrelevant. This makes the tool easier to use for newcomers but also enables power users to combine different libraries efficiently. I supervised Thomas in his learning of Elpi, Coq-Elpi and finally diving into the internals of Hierarchy-Builder.
<https://github.com/math-comp/hierarchy-builder/>
- Julien Wintz: “ADT Elpi”, 2022, SED, full supervision
Julien implemented an execution-trace browser for Elpi programs in the VSCode editor as part of an ADT executed in 2022. In August 2023 the CDT approved a follow up ADT on Elpi to further improve the trace browser. I consider this experience particularly successful not only because the trace browser proved to be useful (especially during Davide Fissore internship), but also because this ADT kick started many interactions between the STAMP team and the SED of Sophia-Antipolis.
<https://github.com/LPCIC/elpi-lang/pull/3>
- Matej Kosic: “Coq API and benchmarks”, 2016, joint supervision with Maxime Dénès.
Matej worked on two orthogonal tasks. On the one hand he improved the Coq APIs used by Coq plugins. On the other hand he developed a benchmark system for Coq, still in use today, that proved to be essential to improve the performances of Coq.

4) Responsabilités collectives / *Community service*

1. Program committee of international conferences and workshops

[COQ-2024](#), [OCAML-2024](#), [COQPL-2024](#), [CPP-2024](#), [CADE29-2023](#), [FIDE-2022](#), [COQPL-2022](#), [MC-2022](#), [FIDE-2021](#), [PADL-2020](#), [WFLP-2020](#), [CPP-2019](#), [LFMTP-2019](#), [UITP-2018](#), [ITP-2018](#), [FIDE-2016](#), [UITP-2016](#), [CICM-2014](#), [JFLA-2013](#), [PSATT-2011](#), [CICM-2011](#), [MKM-2010](#)

2. Chair

[COQ-2023](#), [LFMTP-2021](#), [COQ-2013](#)

3. Steering committee

[LFMTP](#) (since 2023)

4. Organizer

[Seventh Coq Users and Developers Workshop 2023](#), [Mathematical Components School 2022](#), [Advanced Coq Winter School 2019](#), [Advanced Coq Winter School 2018](#), [Advanced Coq Winter School 2017](#), [Advanced Coq Winter School 2016](#), [Second Coq Implementors Workshop 2016](#), [First Coq Coding Sprint 2015](#)

5) Enseignement / *Teaching*

I've been teaching Coq and the Mathematical Components library to Master level and PhD students. All the schools listed below took place at Inria Sophia-Antipolis. On average I did lecture for 6h per school and I did take care of preparing the teaching material for the entire school (website, online interactive slides and exercises).

- [Mathematical Components School 2022](#)
- [Advanced Coq Winter School 2019](#)

- [Advanced Coq Winter School 2018](#)
- [Advanced Coq Winter School 2017](#)
- [Advanced Coq Winter School 2016](#)
- [Coq Winter School 2015](#)

I did teach introductory courses of Type Theory in Coq:

- [EUTypes Summer School 2018](#), 4h, Phd students, Ohrid, Macedonia.
- [Corso di teoria dei tipi in Coq](#). 2017, 16h, master students, University of Padua.

6) **Visibilité / *Visibility***

Plenary talks and panels :

- Invited talk at [CoqPL-2025](#) with title “Elpi: rule-based meta-language for Rocq”.
- Panel member as a Coq core developer at CoqPL-2023, [CoqPL-2021](#), CoqPL-2019, CoqPL-2018.
- Invited talk at [ML-Family-Workshop-2018](#) with title “ELPI: an extension language with binders and unification variables”.
- Invited talk at [UITP-2014](#) with title “Asynchronous Processing of Formal Documents in Coq”.
- Invited tutorial at [ITP-2013](#) with title “The Mathematical Components library: principles and design choices”.

PhD Jury member :

- [Enzo CRANCE](#) (2023, Nantes)
- [Gabriel HONDET](#) (2022, Online)
- [Ulysse GERARD](#) (2019, Paris)
- [Roberto BLANCO](#) (2017, Paris)

2 — LISTE COMPLÈTE DES CONTRIBUTIONS

2 — COMPLETE LIST OF CONTRIBUTIONS

1. Publications

1.1 Revues internationales / *International journals*

- [GCT19] Ferruccio Guidi, Claudio Sacerdoti Coen, and Enrico Tassi. “Implementing type theory in higher order constraint logic programming”. In: *Mathematical Structures in Computer Science* 29.8 (2019), pp. 1125–1150. doi: [10.1017/S0960129518000427](https://doi.org/10.1017/S0960129518000427).
- [Fai+18] Alexander John Faithfull, Jesper Bengtson, Enrico Tassi, and Carst Tankink. “Coqoon - An IDE for interactive proof development in Coq”. In: *Int. J. Softw. Tools Technol. Transf.* 20.2 (2018), pp. 125–137. doi: [10.1007/S10009-017-0457-2](https://doi.org/10.1007/S10009-017-0457-2). URL: <https://doi.org/10.1007/s10009-017-0457-2>.
- [Asp+12a] Andrea Asperti, Wilmer Ricciotti, Claudio Sacerdoti Coen, and Enrico Tassi. “A Bi-Directional Refinement Algorithm for the Calculus of (Co)Inductive Constructions”. In: *Log. Methods Comput. Sci.* 8.1 (2012). doi: [10.2168/LMCS-8\(1:18\)2012](https://doi.org/10.2168/LMCS-8(1:18)2012). URL: [https://doi.org/10.2168/LMCS-8\(1:18\)2012](https://doi.org/10.2168/LMCS-8(1:18)2012).
- [Asp+12b] Andrea Asperti, Wilmer Ricciotti, Claudio Sacerdoti Coen, and Enrico Tassi. “Formal Metatheory of Programming Languages in the Matita Interactive Theorem Prover”. In: *J. Autom. Reason.* 49.3 (2012), pp. 427–451. doi: [10.1007/S10817-011-9228-Z](https://doi.org/10.1007/S10817-011-9228-Z). URL: <https://doi.org/10.1007/s10817-011-9228-z>.
- [CT11] Claudio Sacerdoti Coen and Enrico Tassi. “Formalising Overlap Algebras in Matita”. In: *Math. Struct. Comput. Sci.* 21.4 (2011), pp. 763–793. doi: [10.1017/S0960129511000107](https://doi.org/10.1017/S0960129511000107). URL: <https://doi.org/10.1017/S0960129511000107>.
- [CT08] Claudio Sacerdoti Coen and Enrico Tassi. “A constructive and formal proof of Lebesgue’s Dominated Convergence Theorem in the interactive theorem prover Matita”. In: *J. Formaliz. Reason.* 1.1 (2008), pp. 51–89. doi: [10.6092/ISSN.1972-5787/1334](https://doi.org/10.6092/ISSN.1972-5787/1334). URL: <https://doi.org/10.6092/issn.1972-5787/1334>.
- [Asp+07] Andrea Asperti, Claudio Sacerdoti Coen, Enrico Tassi, and Stefano Zacchiroli. “User Interaction with the Matita Proof Assistant”. In: *J. Autom. Reason.* 39.2 (2007), pp. 109–139. doi: [10.1007/S10817-007-9070-5](https://doi.org/10.1007/S10817-007-9070-5). URL: <https://doi.org/10.1007/s10817-007-9070-5>.

1.2 Conférences internationales avec comité de lecture / *Reviewed international conferences*

- [FT24] Davide Fissore and Enrico Tassi. “Higher-Order unification for free!: Reusing the meta-language unification for the object language”. In: *Proceedings of the 26th International Symposium on Principles and Practice of Declarative Programming*. PPDP ’24. Milano, Italy: Association for Computing Machinery, 2024. ISBN: 9798400709692. doi: [10.1145/3678232.3678233](https://doi.org/10.1145/3678232.3678233). URL: <https://doi.org/10.1145/3678232.3678233>.
- [GLT23] Benjamin Grégoire, Jean-Christophe Léchenet, and Enrico Tassi. “Practical and sound equality tests, automatically – Deriving eqType instances for Jasmin’s data types with Coq-Elpi”. In: *CPP ’23: 12th ACM SIGPLAN International Conference on Certified Programs and Proofs*. CPP 2023: Proceedings of the 12th ACM SIGPLAN International Conference on Certified Programs and Proofs. Boston, MA, USA: ACM, Jan. 2023, pp. 167–181. doi: [10.1145/3573105.3575683](https://doi.org/10.1145/3573105.3575683). URL: <https://inria.hal.science/hal-03800154>.
- [CST20] Cyril Cohen, Kazuhiko Sakaguchi, and Enrico Tassi. “Hierarchy Builder: Algebraic hierarchies Made Easy in Coq with Elpi”. In: *5th International Conference on Formal Structures for Computation and Deduction (FSCD 2020)*. Ed. by Zena M. Ariola. Vol. 167. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2020, 34:1–34:21. ISBN: 978-3-95977-155-9. doi: [10.4230/LIPIcs.FSCD.2020.34](https://doi.org/10.4230/LIPIcs.FSCD.2020.34). URL: <https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.FSCD.2020.34>.
- [Tas19] Enrico Tassi. “Deriving proved equality tests in Coq-elpi: Stronger induction principles for containers in Coq”. In: *ITP 2019 - 10th International Conference on Interactive Theorem Proving*. Portland, OR, United States, Sept. 2019. doi: [10.4230/LIPIcs.CVIT.2016.23](https://doi.org/10.4230/LIPIcs.CVIT.2016.23). URL: <https://inria.hal.science/hal-01897468>.
- [Fai+16] Alexander John Faithfull, Jesper Bengtson, Enrico Tassi, and Carst Tankink. “Coqoon - An IDE for Interactive Proof Development in Coq”. In: *Tools and Algorithms for the Construction and Analysis of Systems - 22nd International Conference, TACAS 2016, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2016, Eindhoven, The Netherlands, April 2-8, 2016, Proceedings*. Ed. by Marsha Chechik and Jean-François Raskin. Vol. 9636. Lecture Notes in Computer Science. Springer, 2016, pp. 316–331. doi: [10.1007/978-3-662-49674-9_18](https://doi.org/10.1007/978-3-662-49674-9_18). URL: https://doi.org/10.1007/978-3-662-49674-9_18.

- [BTT15] Bruno Barras, Carst Tankink, and Enrico Tassi. “Asynchronous Processing of Coq Documents: From the Kernel up to the User Interface”. In: *Interactive Theorem Proving - 6th International Conference, ITP 2015, Nanjing, China, August 24-27, 2015, Proceedings*. Ed. by Christian Urban and Xingyuan Zhang. Vol. 9236. Lecture Notes in Computer Science. Springer, 2015, pp. 51–66. doi: [10.1007/978-3-319-22102-1_4](https://doi.org/10.1007/978-3-319-22102-1_4). URL: https://doi.org/10.1007/978-3-319-22102-1_4.
- [Dun+15] Cvetan Dunchev, Ferruccio Guidi, Claudio Sacerdoti Coen, and Enrico Tassi. “ELPI: Fast, Embeddable, λ Prolog Interpreter”. In: *Logic for Programming, Artificial Intelligence, and Reasoning - 20th International Conference, LPAR-20 2015, Suva, Fiji, November 24-28, 2015, Proceedings*. Ed. by Martin Davis, Ansgar Fehnker, Annabelle McIver, and Andrei Voronkov. Vol. 9450. 2015, pp. 460–468. doi: [10.1007/978-3-662-48899-7_32](https://doi.org/10.1007/978-3-662-48899-7_32). URL: <https://inria.hal.science/hal-01176856v1>.
- [Chy+14] Frédéric Chyzak, Assia Mahboubi, Thomas Sibut-Pinote, and Enrico Tassi. “A Computer-Algebra-Based Formal Proof of the Irrationality of $\zeta(3)$ ”. In: *Interactive Theorem Proving - 5th International Conference, ITP 2014, Held as Part of the Vienna Summer of Logic, VSL 2014, Vienna, Austria, July 14-17, 2014. Proceedings*. Ed. by Gerwin Klein and Ruben Gamboa. Vol. 8558. Lecture Notes in Computer Science. Springer, 2014, pp. 160–176. doi: [10.1007/978-3-319-08970-6_11](https://doi.org/10.1007/978-3-319-08970-6_11). URL: https://doi.org/10.1007/978-3-319-08970-6_11.
- [Bar+13] Bruno Barras, Lourdes Del Carmen González-Huesca, Hugo Herbelin, Yann Régis-Gianas, Enrico Tassi, Makarius Wenzel, and Burkhart Wolff. “Pervasive Parallelism in Highly-Trustable Interactive Theorem Proving Systems”. In: *Intelligent Computer Mathematics - MKM, Calculemus, DML, and Systems and Projects 2013, Held as Part of CICM 2013, Bath, UK, July 8-12, 2013. Proceedings*. Ed. by Jacques Carette, David Aspinall, Christoph Lange, Petr Sojka, and Wolfgang Windsteiger. Vol. 7961. Lecture Notes in Computer Science. Springer, 2013, pp. 359–363. doi: [10.1007/978-3-642-39320-4_29](https://doi.org/10.1007/978-3-642-39320-4_29). URL: https://doi.org/10.1007/978-3-642-39320-4_29.
- [Gon+13] 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: *Interactive Theorem Proving - 4th International Conference, ITP 2013, Rennes, France, July 22-26, 2013. Proceedings*. Ed. by Sandrine Blazy, Christine Paulin-Mohring, and David Pichardie. Vol. 7998. Lecture Notes in Computer Science. Springer, 2013, pp. 163–179. doi: [10.1007/978-3-642-39634-2_14](https://doi.org/10.1007/978-3-642-39634-2_14). URL: <https://inria.hal.science/hal-00816699v1>.
- [MT13] Assia Mahboubi and Enrico Tassi. “Canonical Structures for the Working Coq User”. In: *Interactive Theorem Proving*. Ed. by Sandrine Blazy, Christine Paulin-Mohring, and David Pichardie. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 19–34. ISBN: 978-3-642-39634-2.
- [GT12] Georges Gonthier and Enrico Tassi. “A Language of Patterns for Subterm Selection”. In: *Interactive Theorem Proving - Third International Conference, ITP 2012, Princeton, NJ, USA, August 13-15, 2012. Proceedings*. Ed. by Lennart Beringer and Amy P. Felty. Vol. 7406. Lecture Notes in Computer Science. Springer, 2012, pp. 361–376. doi: [10.1007/978-3-642-32347-8_25](https://doi.org/10.1007/978-3-642-32347-8_25). URL: https://doi.org/10.1007/978-3-642-32347-8_25.
- [Asp+11] Andrea Asperti, Wilmer Ricciotti, Claudio Sacerdoti Coen, and Enrico Tassi. “The Matita Interactive Theorem Prover”. In: *Automated Deduction - CADE-23 - 23rd International Conference on Automated Deduction, Wroclaw, Poland, July 31 - August 5, 2011. Proceedings*. Ed. by Nikolaj S. Bjørner and Viorica Sofronie-Stokkermans. Vol. 6803. Lecture Notes in Computer Science. Springer, 2011, pp. 64–69. doi: [10.1007/978-3-642-22438-6_7](https://doi.org/10.1007/978-3-642-22438-6_7). URL: https://doi.org/10.1007/978-3-642-22438-6_7.
- [AT10] Andrea Asperti and Enrico Tassi. “Smart Matching”. In: *Intelligent Computer Mathematics, 10th International Conference, AISC 2010, 17th Symposium, Calculemus 2010, and 9th International Conference, MKM 2010, Paris, France, July 5-10, 2010. Proceedings*. Ed. by Serge Autexier, Jacques Calmet, David Delahaye, Patrick D. F. Ion, Laurence Rideau, Renaud Rioboo, and Alan P. Sexton. Vol. 6167. Lecture Notes in Computer Science. Springer, 2010, pp. 263–277. doi: [10.1007/978-3-642-14128-7_23](https://doi.org/10.1007/978-3-642-14128-7_23). URL: https://doi.org/10.1007/978-3-642-14128-7_23.
- [Asp+09] Andrea Asperti, Wilmer Ricciotti, Claudio Sacerdoti Coen, and Enrico Tassi. “Hints in Unification”. In: *Theorem Proving in Higher Order Logics*. Ed. by Stefan Berghofer, Tobias Nipkow, Christian Urban, and Makarius Wenzel. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 84–98. ISBN: 978-3-642-03359-9.
- [AT09] Andrea Asperti and Enrico Tassi. “Superposition as a logical glue”. In: *Proceedings Types for Proofs and Programs, Revised Selected Papers, TYPES 2009, Aussois, France, 12-15th May 2009*. Ed. by Tom Hirschowitz. Vol. 53. EPTCS. 2009, pp. 1–15. doi: [10.4204/EPTCS.53.1](https://doi.org/10.4204/EPTCS.53.1). URL: <https://doi.org/10.4204/EPTCS.53.1>.

- [CT09a] Claudio Sacerdoti Coen and Enrico Tassi. “Natural Deduction Environment for Matita”. In: *Intelligent Computer Mathematics, 16th Symposium, Calculemus 2009, 8th International Conference, MKM 2009, Held as Part of CICM 2009, Grand Bend, Canada, July 6-12, 2009. Proceedings*. Ed. by Jacques Carette, Lucas Dixon, Claudio Sacerdoti Coen, and Stephen M. Watt. Vol. 5625. Lecture Notes in Computer Science. Springer, 2009, pp. 486–491. doi: [10.1007/978-3-642-02614-0_40](https://doi.org/10.1007/978-3-642-02614-0_40). URL: https://doi.org/10.1007/978-3-642-02614-0_40.
- [CT09b] Claudio Sacerdoti Coen and Enrico Tassi. “Nonuniform Coercions via Unification Hints”. In: *Proceedings Types for Proofs and Programs, Revised Selected Papers, TYPES 2009, Aussois, France, 12-15th May 2009*. Ed. by Tom Hirschowitz. Vol. 53. EPTCS. 2009, pp. 16–29. doi: [10.4204/EPTCS.53.2](https://doi.org/10.4204/EPTCS.53.2). URL: <https://doi.org/10.4204/EPTCS.53.2>.
- [AT07] Andrea Asperti and Enrico Tassi. “Higher order Proof Reconstruction from Paramodulation-Based Refutations: The Unit Equality Case”. In: *Towards Mechanized Mathematical Assistants, 14th Symposium, Calculemus 2007, 6th International Conference, MKM 2007, Hagenberg, Austria, June 27-30, 2007, Proceedings*. Ed. by Manuel Kauers, Manfred Kerber, Robert Miner, and Wolfgang Windsteiger. Vol. 4573. Lecture Notes in Computer Science. Springer, 2007, pp. 146–160. doi: [10.1007/978-3-540-73086-6_14](https://doi.org/10.1007/978-3-540-73086-6_14). URL: https://doi.org/10.1007/978-3-540-73086-6_14.
- [CT07] Claudio Sacerdoti Coen and Enrico Tassi. “Working with Mathematical Structures in Type Theory”. In: *Types for Proofs and Programs, International Conference, TYPES 2007, Cividale del Friuli, Italy, May 2-5, 2007, Revised Selected Papers*. Ed. by Marino Miculan, Ivan Scagnetto, and Furio Honsell. Vol. 4941. Lecture Notes in Computer Science. Springer, 2007, pp. 157–172. doi: [10.1007/978-3-540-68103-8_11](https://doi.org/10.1007/978-3-540-68103-8_11). URL: https://doi.org/10.1007/978-3-540-68103-8_11.
- [Gon+07] Georges Gonthier, Assia Mahboubi, Laurence Rideau, Enrico Tassi, and Laurent Théry. “A Modular Formalisation of Finite Group Theory”. In: *Theorem Proving in Higher Order Logics, 20th International Conference, TPHOLs 2007, Kaiserslautern, Germany, September 10-13, 2007, Proceedings*. Ed. by Klaus Schneider and Jens Brandt. Vol. 4732. Lecture Notes in Computer Science. Springer, 2007, pp. 86–101. doi: [10.1007/978-3-540-74591-4_8](https://doi.org/10.1007/978-3-540-74591-4_8). URL: https://doi.org/10.1007/978-3-540-74591-4_8.
- [Asp+06] Andrea Asperti, Claudio Sacerdoti Coen, Enrico Tassi, and Stefano Zacchiroli. “Crafting a Proof Assistant”. In: *Types for Proofs and Programs, International Workshop, TYPES 2006, Nottingham, UK, April 18-21, 2006, Revised Selected Papers*. Ed. by Thorsten Altenkirch and Conor McBride. Vol. 4502. Lecture Notes in Computer Science. Springer, 2006, pp. 18–32. doi: [10.1007/978-3-540-74464-1_2](https://doi.org/10.1007/978-3-540-74464-1_2). URL: https://doi.org/10.1007/978-3-540-74464-1_2.
- [Asp+04] Andrea Asperti, Ferruccio Guidi, Claudio Sacerdoti Coen, Enrico Tassi, and Stefano Zacchiroli. “A Content Based Mathematical Search Engine: Whelp”. In: *Types for Proofs and Programs, International Workshop, TYPES 2004, Jouy-en-Josas, France, December 15-18, 2004, Revised Selected Papers*. Ed. by Jean-Christophe Filliâtre, Christine Paulin-Mohring, and Benjamin Werner. Vol. 3839. Lecture Notes in Computer Science. Springer, 2004, pp. 17–32. doi: [10.1007/11617990_2](https://doi.org/10.1007/11617990_2). URL: https://doi.org/10.1007/11617990_2.

1.3 Ateliers internationaux avec comité de lecture / *Reviewed international workshops*

- [FT23] Davide Fissore and Enrico Tassi. “A new Type-Class solver for Coq in Elpi”. In: *The Coq Workshop 2023*. Bialystok, Poland, July 2023. URL: <https://inria.hal.science/hal-04467855>.
- [PTZ22] Karl Palmkog, Enrico Tassi, and Théo Zimmermann. “Reliably Reproducing Machine-Checked Proofs with the Coq Platform”. In: *Proceedings of RRRR 2022, Reproducibility and Replication of Research Results*. 2022.
- [Aff+21] Reynald Affeldt, Xavier Allamigeon, Yves Bertot, Quentin Canu, Cyril Cohen, Pierre Roux, Kazuhiko Sakaguchi, Enrico Tassi, Laurent Théry, and Anton Trunov. “Porting the Mathematical Components library to Hierarchy Builder”. In: *the COQ Workshop 2021*. virtuel- Rome, Italy, July 2021. URL: <https://hal.science/hal-03463762>.
- [Tas18] Enrico Tassi. “Elpi: an extension language for Coq (Metaprogramming Coq in the Elpi λ Prolog dialect)”. In: *The Fourth International Workshop on Coq for Programming Languages*. Los Angeles, CA, United States, Jan. 2018. URL: <https://inria.hal.science/hal-01637063>.
- [DCT16] Cvetan Dunchev, Claudio Sacerdoti Coen, and Enrico Tassi. “Implementing HOL in an Higher Order Logic Programming Language”. In: *Proceedings of the Eleventh Workshop on Logical Frameworks and Meta-Languages: Theory and Practice*. LFMTTP ’16. Porto, Portugal: Association for Computing Machinery, 2016. ISBN: 9781450347778. doi: [10.1145/2966268.2966272](https://doi.org/10.1145/2966268.2966272). URL: <https://doi.org/10.1145/2966268.2966272>.
- [GT16] Benjamin Grégoire and Enrico Tassi. “Boolean reflection via type classes”. In: *Coq Workshop 2016*. Nancy, France, Aug. 2016. URL: <https://inria.hal.science/hal-01410530>.

- [AT08] Andrea Asperti and Enrico Tassi. “An Interactive Driver for Goal-directed Proof Strategies”. In: *Proceedings of the 8th International Workshop on User Interfaces for Theorem Provers, UITP@TPHOLs 2008, Montréal, Canada, August 22, 2008*. Ed. by Serge Autexier and Christoph Benzmüller. Vol. 226. Electronic Notes in Theoretical Computer Science. Elsevier, 2008, pp. 89–105. doi: [10.1016/j.entcs.2008.12.099](https://doi.org/10.1016/j.entcs.2008.12.099). URL: <https://doi.org/10.1016/j.entcs.2008.12.099>.
- [CTZ06] Claudio Sacerdoti Coen, Enrico Tassi, and Stefano Zacchiroli. “Tincals: Step by Step Tacticals”. In: *Proceedings of the 7th Workshop on User Interfaces for Theorem Provers, UITP@FLoC 2006, Seattle, WA, USA, August 21, 2006*. Ed. by Serge Autexier and Christoph Benzmüller. Vol. 174. Electronic Notes in Theoretical Computer Science 2. Elsevier, 2006, pp. 125–142. doi: [10.1016/j.entcs.2006.09.026](https://doi.org/10.1016/j.entcs.2006.09.026). URL: <https://doi.org/10.1016/j.entcs.2006.09.026>.

1.4 Livres et chapitres de livre / *Books and book chapters*

- [MT22] Assia Mahboubi and Enrico Tassi. *Mathematical Components*. Zenodo, Sept. 2022. doi: [10.5281/zenodo.7118596](https://doi.org/10.5281/zenodo.7118596). URL: <https://doi.org/10.5281/zenodo.7118596>.
- [PT21] Elaine Pimentel and Enrico Tassi, eds. *Proceedings of the Sixteenth Workshop on Logical Frameworks and Meta-Languages: Theory and Practice, LFMT 2021, Pittsburgh, USA, 16th July 2021*. Vol. 337. EPTCS. 2021. doi: [10.4204/EPTCS.337](https://doi.org/10.4204/EPTCS.337). URL: <https://doi.org/10.4204/EPTCS.337>.

1.5 Rapports de recherche et articles soumis / *Research reports and publications under review*

- [GMT15] Georges Gonthier, Assia Mahboubi, and Enrico Tassi. *A Small Scale Reflection Extension for the Coq system*. Research Report RR-6455. Inria Saclay Ile de France, 2015. URL: <https://inria.hal.science/inria-00258384>.

2. Développements technologiques / *Technology development*

- Elpi: Nature=research; Audience=partners; Evolution=Long term; Duration>=9; Contribution=leader, instigator, developer; Url=<https://github.com/LPCIC/elpi/>
Elpi is a programming language well-tailored for the manipulation of syntax trees with binders and placeholders, the primary data structures employed within the Coq system to represent code, specifications, and proofs. It counts about 20K lines of OCaml code. I’m the main author of this piece of software.
- Coq-Elpi: Nature=research; Audience=partners; Evolution=Long term; Duration>=7; Contribution=leader, instigator, developer; Url=<https://github.com/LPCIC/coq-elpi>
Coq-Elpi is a Coq plugin that implements the glue needed in order to extend the Coq system via the Elpi programming language. It counts about 25K lines of code (16K OCaml, 9K Elpi). I’m the main author of this piece of software.
- Hierarchy-Builder: Nature=research; Audience=community; Evolution=Long term; Duration>=4; Contribution=instigator, developer; Url=<https://github.com/math-comp/hierarchy-builder>
Hierarchy-Builder is a declarative language to describe interfaces and organize them in a hierarchy. It is designed to ease the task of structuring Coq libraries and organize their contents into reusable theories. It is implemented in Coq-Elpi. It counts 5K lines of Elpi code. The entire code base was written in “pair programming” with Cyril Cohen.
- Coq Platform: Nature=vehicle; Audience=universe; Evolution=Long term; Duration>=4; Contribution=instigator, developer; Url=<https://github.com/coq/platform/>
The Coq Platform is a distribution of Coq together with libraries built with Coq. It provides a coherent set of reusable libraries to: 1) lower the cost of adoption of Coq in the industry; 2) make scientific work based on Coq easier to reproduce; 3) finally make Coq easier to install and teach. It counts 7K lines of shell script. I contribute to the continuous integration infrastructure and the Linux and Windows installers.
- Coq: Nature=research; Audience=community,universe; Evolution=Long term; Duration>=13; Contribution=developer; Url=<https://coq.inria.fr/>
Coq is an interactive theorem prover used to machine-check a proof, giving an ultimate degree of certainty about its correctness. Coq is used to both verify mathematical theories and the conformance of software to its specification. It counts 300K lines of OCaml code and 200K lines of Coq code. I contribute to the development of the system, in particular the user facing language and user interfaces. I contribute to the project management as a release manager. I contribute to the community by organizing schools and coding sprint events.

- Mathematical-Components: Nature=research; Audience=community; Evolution=Long term; Duration>=14; Contribution=developer; Url=<https://github.com/math-comp/math-comp>

Largest and most widely used mathematical library for Coq. It counts 130K lines of Coq/SSReflect/HB code. I develop the SSReflect proof language and the Hierarchy Builder (HB) tool. I contribute the theory of finite sets, tuples, finitely supported functions. I contribute to the project management and the tools the documentation, i.e. I wrote a book on the library.