

Beta Ziliani

Team Leader · Researcher · Developer · Professor

✉ beta.ziliani@gmail.com | 🏠 beta-ziliani | 🌐 @beta-ziliani | in beta-ziliani

🏢 Workplaces, roles and tools

Universidad ORT Uruguay

- 📅 Since October 2025
- 👤 *Catedrático* (Group Leader), Researcher, Professor.
- 🏠 Facultad de Ingeniería, Universidad ORT Uruguay
- 👤 Eduardo Magarelli, Head of the Engineering Department (✉ mangarelli@ort.edu.uy).

In this position I fill three different roles: I'm a *Catedrático*, the person that makes sure the courses on Theory of Computation run smoothly (7 curses as of today). This amounts to, among other things: finding the right people to teach them, help with curricula changes, and propose new courses. Then, I'm also a researcher and a professor of one of those courses.

Manas Technology Solutions

- 📅 April 2021 – November 2025
- 👤 Team Leader, Product Manager, Senior Engineer.
- 🔧 Crystal, Rust, TypeScript, GitHub, custom PjM tools.
- 🏠 manas.tech
- 👤 Diego Liberman, CEO (✉ dliberman@manas.tech).

I've lead the development of The Crystal Programming Language, of which I'm still part of the Steering Council and Core Team. This language has a vibrant community of thousands of users around the globe, and several companies and individuals help Manas to fund its development.

As Team Leader, I've coordinated and set the priorities of a diverse team that varied in composition over time, ranging from two to five people. And I stress *diverse*: at times we covered a 12hr span of timezones, from Argentina to Australia, with six different nationalities and cultures. I add that we had a neurodivergent developer. The keys to success are: clear communications; constant and non-intrusive tracking of progress; and periodic one-to-one meetings to hear and provide feedback.

As Product Manager, I've set the focus of the bulk of the work. That is, I got to decide what are the major milestones, naturally taking my team's opinions. Developers had the liberty to, besides the assigned topic, work on issues of their pick. This is both a recognition to their great commitment to the project, and a strategy to keep them motivated.

Since the team was small but the tasks were many, sometimes I had to "roll up my sleeves" and not only review PRs, but also to create them.

During this time, we advanced in stabilizing the language, the compiler and its ecosystem; we regularized the release schedule; and we make it grow in different platforms.

As the visible head of the language, I also took the role of Business Development Representative. For instance, I was the liaison between the current major funder of Crystal and Manas.

Lastly, I also took part of the Slang team, building compiler tools for Solidity in Rust and TypeScript.

CONICET and FAMAF, UNC

-  April 2015 – March 2021.
-  Researcher and Professor.
-  OCaml, Coq, $\LaTeX$.

In this period I started a research group in the topic of Foundations of Programming Languages, focusing on programming language's semantics and interactive theorem proving. I successfully draw funding from several agencies; publish in renowned journals and conferences; supervised a PhD thesis and three undergrad theses; and taught courses on programming languages and paradigms (see following sections for details). The teaching continued until early 2024.

Max Planck Institute for Software Systems (MPI-SWS)

-  January 2010 – March 2015.
-  PhD student.
-  OCaml, Coq, $\LaTeX$.

During my PhD I specialized in the topic of interactive theorem proving. From my own initiative, I built two plugins for Coq: *Mtac* and *Unicoq*. The first one is a typed meta-language for proofs, allowing to write proof scripts that are guaranteed to solve the expected goals. The second one is a drop-in replacement for the unification algorithm, one of the key parts of an interactive prover. The interpreter of *Mtac* and the unification algorithm are written in OCaml.

Manas Technology Solutions S.R.L.

-  October 2004 – November 2009.
-  Principal Engineer, Project Manager, Researcher.
-  Java, C#, Ruby, RoR, $\LaTeX$.
-  manas.tech
-  (00-54-11) 47-11-00-70
-  Nicolás Di Tada, CTO.

In this first iteration at Manas, I was responsible or part of the development of various apps, including a WYSIWYG SVG map drawer and an SMS notifier for health risk related events. During the last year I was granted the possibility to work on my MSc Thesis on the mathematical side of computations (details below).

Hasar Sistemas S.R.L.

-  July 1998 – October 2004
-  Developer and Project Manager.
-  C, C++, C#, Java, Visual Basic 6.0, VBA, PalmOS, Windows, Windows CE, Unix.
-  www.hasar.com
-  (00-54-11) 47-27-89-00
-  Daniel A. Bonet, Manager.

In Hasar I developed several applications and services for desktops (Windows), servers (Windows and Unix) and mobile (PalmOS, Windows CE). Given my impulse to learn, I was given first access to the new technology stack to learn, test and report about its possible inclusion in the company's projects. I

was also technically responsible from end-to-end for a handful of small projects: I collected the client's requirements, estimated the effort, developed, tested and installed the program or service.

Fun fact: more than fifteen years later, I was surprised to know that one of the services I developed was still running in the stores of one of the major electronics supply store in Argentina.

Formal Education and Research

Nov 2015 – Mar 2021	Researcher (Inv. Asistente and then Adjunto ¹) CONICET and FAMAF, Universidad Nacional de Córdoba (UNC), member of the group Logics, Interaction and Intelligent Systems (LIIS) lead by Dr. Carlos Areces.
Apr – Oct 2015	Postdoc CONICET in the group LIIS.
2015	PhD in Computer Science, Saarland University & Max Planck Institute for Software Systems (MPI-SWS), Saarbruecken, Germany. Tesis: “Interactive Typed Tactic Programming in the Coq Proof Assistant”. Thesis Supervisor: Dr. Derek Dreyer Graded as Summa Cum Laude .
2010	Oregon Programming Languages Summer School (OPLSS '10).
2009	Licenciatura (equiv. MSc) in Computer Science, Facultad de Ciencias Exactas y Naturales (FCEyN), Universidad de Buenos Aires (UBA), Argentina. Thesis: “ $\lambda\sigma_{GC}$: a calculus based on $\lambda\sigma$ with Garbage Collection.” Thesis Supervisor: Prof. Alejandro Ríos

Selected teaching

Courses in charge

I was responsible for the following courses:

- **Advanced Concepts of Programming Languages:** In this course I let the students guide the agenda; making each edition unique. In some cases we cover theoretical aspects like operational semantics and their soundness proofs, while in others we got a dive of more practical matters, such as Rust's borrow checker.
- **Programming Paradigms (laboratory):** This course students learn basics of functional programming, object oriented programming, and frameworks (e.g. concurrency via actors) while coding interesting projects.
- **Interactive Theorem Proving: Theory and Practice:** A short course on Type Theory and Coq.
- **Databases:** Being outside of my comfort zone, this course is the “bullet I take for the team”. That said, I adapted it to my taste and, for instance, instead of seeing DB semantics through the inadequate set semantics, I based it on list semantics.

¹First and second researcher position after postdocs.

Professorships

These are the positions I held as professor:

- 2023: Associate Professor (Profesor Asociado) in FAMAF, UNC.
- 2016 – 2022: Adjoint Professor (Profesor Adjunto) in FAMAF, UNC.
- 2014: Visiting professor in the *Escuela de Ciencias Informáticas* (ECI 2014), FCEyN, UBA, for the course “Demostración interactiva de teoremas: teoría y práctica”.

Q Scientific Activity

Supervision

Postdoc

1. 2021 – 2024: Mallku E. Soldevila Raffa (FAMAF, UNC y CONICET).

PhD

1. 2017 – 2020: Co-supervisor with Daniel Fridlender (FAMAF, UNC) the PhD thesis of Mallku E. Soldevila Raffa (FAMAF, UNC y CONICET). “*Operational semantics and its application for the study of Garbage Collection in Lua 5.2*”.

Undergrad

4. May 2018 – Jun 2019: Co-supervisor (with Raúl Fervari) of the MSc thesis of Valentina Trucco. “*Verifying Dynamic Modal Logics in Coq*”.
3. Jan 2018 – Apr 2020: Supervisor of the MSc thesis of Ignacio Tiraboschi. “*Generalization of Meta-programs with Dependent Types in Mtac2*”.
2. Dec 2013 – Apr 2014: Co-supervisor (with Derek Dreyer) of the research lab report of Jan-Oliver Kaiser.
1. Aug 2009 – Apr 2010: Co-supervisor (with Alejandro Ríos) of the MSc thesis of Ariel Mendelzon, “Swapping: a natural bridge between named and indexed explicit substitution calculi”.

Publications

At a glance

Indicator	Scopus	Google Scholar
h-index	9	9
Cites	155	353
# co-authors	18	-

Books

1. R. Ribeiro and B. Ziliani, editors. *SBLP 2019: Proceedings of the XXIII Brazilian Symposium on Programming Languages*, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450376389

Journals

6. M. Soldevila, B. Ziliani, and B. Silvestre. From specification to testing: semantics engineering for Lua 5.2. *Journal of Automated Reasoning*, 66:905–952, 2022. **SJR 2022: Q2**
5. R. Fervari, F. Trucco, and B. Ziliani. Verification of dynamic bisimulation theorems in Coq. *Journal of Logical and Algebraic Methods in Programming*, 120:100642, 2021. ISSN 2352-2208. doi: <https://doi.org/10.1016/j.jlamp.2021.100642>. URL <https://www.sciencedirect.com/science/article/pii/S2352220821000055>. **SJR 2020: Q2**
4. J.-O. Kaiser, B. Ziliani, R. Krebbers, Y. Régis-Gianas, and D. Dreyer. Mtac2: Typed tactics for backward reasoning in Coq. *Proc. ACM Program. Lang.*, 2(ICFP):78:1–78:31, July 2018. ISSN 2475-1421. doi: 10.1145/3236773. URL <http://doi.acm.org/10.1145/3236773>. **SJR 2021: Q1**
3. B. Ziliani and M. Sozeau. A comprehensible guide to a new unifier for CIC including universe polymorphism and overloading. *Journal of Functional Programming*, 27, 2017. doi: 10.1017/S0956796817000028. **SJR 2017: Q2**
2. B. Ziliani, D. Dreyer, N. R. Krishnaswami, A. Nanevski, and V. Vafeiadis. Mtac: A monad for typed tactic programming in Coq. *Journal of Functional Programming*, 25, 2015. doi: 10.1017/S0956796815000118. **SJR 2015: Q2**
1. G. Gonthier, B. Ziliani, A. Nanevski, and D. Dreyer. How to make ad hoc proof automation less ad hoc. *Journal of Functional Programming*, 23(4):357–401, 2013. doi: 10.1017/S0956796813000051. **SJR 2013: Q2**

Proceedings

Peer-reviewed proceedings, with at least 16 single-column pages or 12 double-column pages.

9. M. Soldevila, R. Ribeiro, and B. Ziliani. Redex2Coq: Towards a Theory of Decidability of Redex's Reduction Semantics. In Y. Bertot, T. Kutsia, and M. Norrish, editors, *15th International Conference on Interactive Theorem Proving (ITP 2024)*, volume 309 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 34:1–34:18, Dagstuhl, Germany, 2024. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. ISBN 978-3-95977-337-9. doi: 10.4230/LIPIcs.ITP.2024.34. **CORE 2023: A**.
8. M. Soldevila, B. Ziliani, and D. Fridlender. Understanding Lua's garbage collection: Towards a formalized static analyzer. In *Proceedings of the 22nd International Symposium on Principles and Practice of Declarative Programming*, PPDP 2020, 2020. **Qualis 2012: B1**.
7. R. Fervari, F. Trucco, and B. Ziliani. Mechanizing bisimulation theorems for relation-changing logics in Coq. In *Proceedings of the 2nd Workshop on Dynamic Logic: New Trends and Applications*, DaLi 19, 2019.
6. M. Soldevila, B. Ziliani, B. Silvestre, D. Fridlender, and F. Mascarenhas. Decoding Lua: Formal semantics for the developer and the semanticist. In *Proceedings of the 13th ACM SIGPLAN Dynamic Languages Symposium*, DLS 2017, 2017. **CORE 2017: B**.
5. B. Ziliani and M. Sozeau. A unification algorithm for coq featuring universe polymorphism and overloading. In *Proceedings of the 20th ACM SIGPLAN International Conference on Functional Programming*, ICFP 2015, pages 179–191, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3669-7. doi: 10.1145/2784731.2784751. **CORE 2017: A***.
4. B. Ziliani, D. Dreyer, N. R. Krishnaswami, A. Nanevski, and V. Vafeiadis. Mtac: A monad for typed tactic programming in Coq. In *Proceedings of the 18th ACM SIGPLAN International Conference on Functional Programming*, ICFP '13, pages 87–100, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-2326-0. doi: 10.1145/2500365.2500579. **CORE 2013: A***.

3. G. Claret, L. del Carmen González Huesca, Y. Régis-Gianas, and B. Ziliani. Lightweight proof by reflection using a posteriori simulation of effectful computation. In *Proceedings of the 4th International Conference on Theorem Proving*, ITP '13, pages 67–83, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg. ISBN 978-3-642-39634-2. doi: 10.1007/978-3-642-39634-2_{_}8. **CORE 2013: A**.
2. G. Gonthier, B. Ziliani, A. Nanovski, and D. Dreyer. How to make ad hoc proof automation less ad hoc. In *Proceedings of the 16th ACM SIGPLAN International Conference on Functional Programming*, ICFP '11, pages 163–175, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0865-6. doi: 10.1145/2034773.2034798. **CORE 2013: A***.
1. A. Mendelzon, A. Ríos, and B. Ziliani. Swapping: a natural bridge between named and indexed explicit substitution calculi. In *Proceedings of the 5th international workshop on Higher-Order Rewriting*, HOR '10, pages 1–15, 2010. doi: <http://dx.doi.org/10.4204/EPTCS.49.1>.

Selected invited talks

7. B. Ziliani. Tactic languages. In *Coq Andes Summer School*, CASS 2020, Cajón del Maipo, Chile, 2020.
6. B. Ziliani. Tutorial: Interactive theorem proving: where are we now, and where are we going? In *X Brazilian Conference on Software: Theory and Practice*, CBSOFT '19, Salvador, Brasil, 2019a.
5. B. Ziliani. Keynote: Using dependent types to build robust tactics in Coq. In *XXIII Brazilian Symposium on Programming Languages*, SBLP '19, Salvador, Brasil, 2019b.
4. B. Ziliani. Introducing Mtac2: first-class tactics with first-class types in Coq. In *Logic and Foundations of Programming Languages (INFINIS)*, LFPL '18, FCEyN, UBA, Argentina, 2018b.
3. B. Ziliani. Towards typed-tactics in Coq: the what, the why, and the how. In *CSEC: Certified Software Engineering in Coq*, CSEC '18, DCC, UChile, Santiago de Chile, Chile, 2018a.
2. B. Ziliani. Interactive typed tactic programming in the Coq proof assistant. In *Logic and Foundations of Programming Languages (INFINIS)*, LFPL '18, FCEyN, UBA, Argentina, 2015.
1. B. Ziliani. Mtac: a monad for typed tactic programming. In *Coq Users Meeting*, San Diego, USA, 2014.

Service

Organizer and/or Program Chair: CrystalConf 2024, FunLeP 2024, FACAS 2022, Crystal Conf 1.0, SBLP 2019.

Scientific Committee: FSCD 2021, SLALM 2019, FunLeP 2019.

Program Committee: VSSTE 2025, IFL 2025, OCaml Workshop 2022, PEPM 2022, TyDe 2021, SBLP 2020, POPL 2020, CoqPL 2018, ICFP 2017, Coq Workshop 2016, HOPE 2016.

External Review Committee: ICFP 2020, ICFP 2016.

Selected funding

Note: Argentina's funding in research has been decreasing over the years. This is why, for instance, the PICT B project got more funding than the PICT D one, even when the latter is aimed to consolidate a research team, while the former is aimed to fund one researcher only.

3. Director of the project “Formal Verification of Reduction Semantics”, SECyT Consolidar II, UNC, 2018–2022 (US\$2,500).
2. Director of the project “Formalization of the Crystal programming language”, PICT D No. 2017-3315, Agencia, 2018–2020 (US\$7,750).
1. Director of the project “Automation of interactive proof assistants”, PICT B No. 2015-2584, Agencia, 2016–2019 (US\$9,500).