William Stein Interview

Conducted by Alexander Diaz-Lopez

William Stein is professor of mathematics at University of Washington and founder/CEO of SageMath, Inc. In addition to his research and books related to modular forms and number theory, he created the open source computer software Sage. William is an avid vert skateboarder and co-owner of Seattle Vert Ramp. His email address is wstein@gmail.com.

For permission to reprint this article, please contact: reprint-permission@ams.org.
DOI: http://dx.doi.org/10.1090/noti1673

Diaz-Lopez: *When did you know you wanted to be a mathematician?*

Stein: I'm not really sure what "a mathematician" is, and I don't necessarily label myself as such. I first got a taste of math research when I was a 12 year old in rural Texas, and I figured out how to quickly sum the first n integers, since I heard that Gauss had done that. I also really liked computing things using calculators, slides rules, and abacuses. I didn't know there was such a thing as a "mathematician" until college.

I was first exposed to the idea of research mathematics when I was a junior in college (at age 19) in Flagstaff, Arizona. I was browsing the computer programming book section at a used bookstore called Bookmans, when I hit a misfiled book called *An Introduction to Modern Algebra* by Burton Jones. I looked in it, and was amazed with what was there—groups, rings, fields, and all these beautiful ideas that I had never seen before!

I signed up for a summer course on how to write proofs and work with algebraic structures, and did every single problem in the textbook *Foundations of Higher Mathematics* by Fletcher and Patty, and was hooked. I changed my major from computer science to math, and pursued mathematics research starting then.

Diaz-Lopez: *Who encouraged or inspired you?*

Stein: At Northern Arizona University (NAU), two math professors—Peter Horn and Adrian Riskin—most encouraged me. They noticed and strongly supported my interest in mathematics, and encouraged me to go to graduate school at NAU for a year, and then go to Berkeley when I got admitted. (In contrast, another professor at NAU told me *not* to go to Berkeley, which had a notorious reputation in the 1990s for failing out many of their grad students.)

At Berkeley, I met Hendrik Lenstra, who became my thesis advisor and spent an enormous amount of time just doing mathematics with me, mostly involving group cohomology. I also talked frequently with the prolific author and instigator Serge Lang, who spent his summers at Berkeley; he gave me dozens of books on math, politics, and much more, and really got me into arithmetic geometry. Hartshorne's courses on algebraic geometry and Ken Ribet's on modular forms were also extremely inspiring.

Then Kevin Buzzard showed up and breathed incredible new energy into the department. I also worked with Robert Coleman and with Barry Mazur when he came to visit for a month. Then I graduated and became Barry's postdoc at Harvard, where he, Dick Gross, and Noam Elkies really inspired me.

For writing software, in grad school Joe Wetherell first got me interested in computing modular forms. After taking a few years off from programming, I started writing a LOT of code again, and David Kohel encouraged me to make my code open source and expand the value of what I was doing by reimplementing my code in Magma, so it could be combined with other number theory code. I visited the Magma group in Sydney, Australia three times, and while there I was deeply inspired by Allan Steele's passion for first rate mathematical software and John Cannon's deep long term strategic approach to building large software systems.

I also worked a lot on computational number theory research with undergraduate students such as Jennifer Balakrishnan and Corina Tarnita (pictured below) when I was an assistant professor at Harvard University.

Corina Tarnita, a former student and collaborator at Harvard University.

Though many people have encouraged and inspired me, some others have instead discouraged me. For example, Richard Fateman, a computer science emeritus professor at Berkeley, has frequently and steadily publicly questioned and discouraged the Sage math software project ever since I started it.[1]

Diaz-Lopez: *How would you describe your research to a graduate student?*

Stein: I have created tools for computing with mathematical objects, mostly in number theory and linear algebra. I started SageMath, which is open source free software for computational mathematics research and teaching. I also published three books; they are on modular forms, undergrad number theory, and the Riemann Hypothesis (with Barry Mazur).

I've developed some new techniques for explicitly computing with objects connected to modular forms, motivated by the problem of computing all quantities appearing in generalizations of the Birch and Swinnerton-Dyer Conjecture.

I also created CoCalc, which is a web application that makes it easy to collaboratively use open source mathematical software in research and teaching.

Diaz-Lopez: *What theorem are you most proud of and what was the most important idea that led to this breakthrough?*

Stein: My biggest contribution to making the world of mathematics more accessible was starting Sage. The most important idea behind Sage is to build the car, instead of reinventing the wheel... then work really, really hard and ignore the "fact" that what I'm doing feels completely impossible. Many other people joined the project, and together we made Sage really useful.

The goal was always to create a practical and useful tool.

My perspective with Sage has always been to try to make a tool that people could use to compute mathematical objects more easily, with minimal friction. They should not have to pay a lot of money, they should have full access to readable source code, and have many good code examples that definitely work. With Sage, instead of worrying about getting a grant, or future grants, I focused completely on writing software to make computation in mathematics more accessible and opened. The goal was always to create a practical and useful tool, rather than do something new and impressive, then get a grant.

Diaz-Lopez: *What advice do you have for current graduate students in math?*

Stein: If you like programming at all, then learn to program well while you're still in math graduate school, since it significantly expands your job prospects. You might think you most want a career as a professor or academic researcher, but by putting a reasonable amount of effort into learning programming, you'll have more options later. In addition to taking a course in programming, do something deeper like getting involved with and contributing to open source projects like Sage!

Unfortunately, some of the mathematical community genuinely considers the creation of mathematical software as not being "real mathematics." If you most love improving mathematical software, you will probably end up having to leave academia (pure math, at least). For example, all of my PhD students now work at Google, Facebook, Microsoft, etc. If you're a student in a "pure math" department, be sure to look outside to other departments for courses on software.

[1] *See* `mathforum.org/kb/message.jspa?messageID=4132045`

Diaz-Lopez: *All mathematicians feel discouraged occasionally. How do you deal with discouragement?*

Stein: I am discouraged right now, and have been for a while.

In 2012, after about eight years of putting a huge amount of effort into Sage, I became discouraged first because I stopped getting grants. I got even more discouraged when Jim Simons and David Eisenbud told me that their plan to "fund open source math software" was to pay to make Magma freely available (but still closed source) to everybody in North America.[2]

I have received substantial grant support over the years, but the grants were mostly for traditional research mathematics in number theory. I naively thought applying for what I passionately really wanted to do, what I thought mathematics really needs, would work. It didn't. I don't know why. Maybe I was just unlucky regarding trendiness? Maybe I just wasn't politically savvy or connected enough? Maybe I got discouraged too quickly? Or maybe it's exactly what Fateman said in his quote:

> By avoiding applications (say, to engineering design, finance, education, scientific visualization, etc etc) the activity is essentially doomed. Why? Government funding for people or projects will be a small percentage of the funding for pure mathematics. That's not much. And the future is pretty grim…

As a result, I started a company, SageMath, Inc., whose main product right now is CoCalc. I work full time on this company now, rather than in academia. Eventually, I hope the company will be successful and will be able to fund what I've always really wanted to do. After much hard work, this dream is now on the horizon.

If my goals and ambition for what Sage can become weren't so big, then maybe I wouldn't feel so discouraged. It's especially hard for me because I see how amazing Sage could be, what it could offer the mathematical community, and how we could get it there, but I can't make my vision a reality because of a woeful lack of resources.

Diaz-Lopez: *Now that you have created your own company, what is the current status of SageMath, and the recently-created CoCalc?*

Stein: Though I'm the founder of SageMath, I currently have little money or resources (time, students) to put towards Sage development, so I do far less compared to what I did 2004–2012. I wish this lack of resources would change, but there is currently no end in sight. Fortunately, many other people contribute to Sage.

Sage itself is extremely powerful for work in combinatorics, linear algebra, manifolds, and number theory. Also, the programming language I chose for Sage is Python™, which has turned out to be a fortunate choice, because Python is incredibly popular in data science and many other related areas, which has led to Sage being part of a fantastic ecosystem.

Sage development proceeds at a steady pace, with many Sage Days workshops in both the US and Europe; for example IMA in Minnesota is sponsoring many workshops this year and OpenDreamKit in Europe too! Most work on Sage is motivated by the needs of research mathematicians for their own work. Releases keep happening, and around 100 people contribute to each release.[3]

CoCalc is a web application that makes it easy to collaboratively use LaTeX, R, and Sage.

CoCalc is a web application that I started writing in 2013, which makes it easy to collaboratively use LaTeX, R, and Sage for teaching and research. The collaborative aspects of CoCalc are also extremely useful for undergrad REUs, e.g., Edray Goins uses CoCalc extensively in an REU, so the students have easy collaborative access to both Sage and LaTeX.

It has about 20K weekly active users and the software stack is fairly mature at this point, after years of users battle testing it. Four people work fulltime on CoCalc.

Diaz-Lopez: *Looking into the future, what are your expectations and goals for both SageMath and CoCalc?*

Stein: The goal of the Sage project is to create a viable open source alternative to Magma, Maple™, Mathematica, and MATLAB, which are all closed source. This means that people have choice—they at least have the option to use open source software for their math research and teaching in all the academic areas represented by those software. Providing such a choice entails both implementing all relevant algorithms in Sage (with competitive efficiency and correctness), and creating corresponding textbooks and documentation.

William Stein together with Edray Goins, a frequent user of CoCalc in REU programs.

[2] *See* `sagemath.blogspot.com/2015/09/the-simons-foundation-and-open-source.html` *for more details.*

[3] *See* `https://wiki.sagemath.org/Workshops`.

Overall, Sage has so far failed at this goal, though it has succeeded in certain specific areas, such as algebraic combinatorics. There are probably less than 100K monthly active users of Sage, but the other commercial software have far more users, based on web traffic and combined annual revenue of likely over 100 million. There are over 4000 people working full time on the commercial math software competitors, whereas the Sage project has far less full-time equivalent contributors.

CoCalc makes it really easy (especially for beginners!) to collaboratively use open source mathematics software in their web browser. It works well now, and my goal is to greatly increase the number of people using it.

CoCalc makes it really easy to collaboratively use open source mathematics software.

Diaz-Lopez: *Any final comments or advice?*

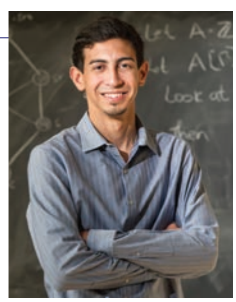
Stein: Rigorous proof greatly improved mathematics research in the 20th century, and open source software may play a similar role in the 21st.

Image Credits

CoCalc banner by Harald Schilly.
All other images courtesy of William Stein.

ABOUT THE INTERVIEWER

Alexander Diaz-Lopez, having earned his PhD at the University of Notre Dame, is now assistant professor at Villanova University. Diaz-Lopez was the first graduate student member of the *Notices* Editorial Board.



Alexander Diaz-Lopez