The Research Software Ecology in Pure Mathematics

William Stein (University of Washington)

University of Washington

May 15, 2012

A day in the life of pure mathematics research...

Date: Mon, **14 May 2012** 13:23:34 -0400
From: **John Tate <tate@math.utexas.edu>**
Subject: Dokchitser for Jacobi sum L's12
Hello William,

Joe Buhler tells me that you may have made Dokchitser easier to use in Sage, by making it able to accept a list of Euler factors rather than a vector of coefficients (here I use "list" in the non-technical sense). If so, I would like to learn how to use it. ... Can you help me?

Swinnerton-Dyer and I are interested in the special values at $s=2$ (the real point on the boundary of convergence), of the L-series of some motives over Q which occur in the H^2 of surfaces in P^3 [...]

Date: Mon, **14 May 2012** 17:53:30 -0400
From: **Barry Mazur <barry.mazur@gmail.com>**
Subject: some interesting computations for a student, maybe...
Hi William,

Sheldon Kamienny and I are looking at the fine structure of X_1(N) for small values of N (for the moment: N=17; later maybe N=29, 31) and we get bogged down by some computations having to do with cusps and rational torsion [...]

What and who?

1. **What is used:**
   - Closed source: Maple, Mathematica, Magma, Matlab
   - Open source: Sage (major library + nearly 100 components), specialized packages (e.g., Macaulay 2, CHomP, LiE)
2. **What is needed:**
   - Open source must **catch up with** the functionality of closed source, before it can solve the problems of *availability* and *flexibility*.
3. **Who maintains this software:**
   - Companies: support their closed source software.
   - Academics: adopt and support open source software that is abandoned by the original author(s), who might leave academia.

Where does the support come from?

1. **Closed source:**
   - License fees: on the order of a hundred million dollars/year
   - Grants and contracts
   - Training courses
   - Wolfram alpha (e.g., in Siri and Bing)
2. **Open source:**
   - NSF: postdocs, summer salary, workshops, students
   - Companies: Google & Microsoft have donated to Sage
   - People: Sage has recieved nearly $40K in private donations
   - DOD: support of "certain open source projects" (confidential)
   - Book royalties: a few thousand dollars
   - German government has a strong culture of supporting software
   - EU's 6 million dollar project to link GAP/Mupad/Maple
   - Institute workshops: MSRI, CMI, IPAM, ICERM, etc.
   - A commercial entity setup to generate money for Sage development via the web: subscriptions, ads, apps, etc., – doesn't exist, but I'm working on it now...

How is the software for research...

1. **Created?**
   - **Innovation:** Researchers generate new algorithms that they implement to prove their worth.
   - **Competition:** People design and implement algorithms from scratch in one system in order to replicate or beat functionality available in another (competitor's or expensive) system.

     (Opinion: I think competitive creation is what open source needs the most right now, since we are already innovators by nature. This is very difficult to fund and support in the US model, and European funding priorities seem too political to systematically get this done.)

2. **Used?**
   - Test ideas and make conjectures
   - "It is always a good idea to try to prove true theorems." –Birch
   - "The object of numerical computation is theoretical advance." –Atkin
   - Subtle modification of existing implementations –Bharghava story

What is most needed?

- **A Strong Foundation:** A team of 10 superstars to work fulltime to *systematically* discover and implement all algorithms important to pure math research that are in Magma, Mathematica, etc., but not in any open source software.

  Opinion: After 7 years of pushing this priority, I think it will **not happen** without a major game changer:
  - I start a company
  - I get major contract work with a company that uses Sage
  - Private foundation?

- NSF is strongly investing in **"software sustainability"**, but I'm concerned about the viability of their approach in my area. For open source to work, one needs a foundation (e.g., the Linux kernel as a foundation for the OS); one can't skip that crucial step. The longer we wait, the harder it is to build that foundation.
- **Sebastian Pancratz**