

---

## SageMathCloud for collaborative document editing and scientific computing

Hal Snyder

### Abstract

The open-source platform SageMathCloud (SMC) lets users collaborate in real time on scientific computing and authoring technical documents with standard  $\text{\LaTeX}$  tools. This article offers a survey of key features, presenting SMC as a unique, unified platform for teachers and researchers to be productive in today's mixed software environments and interdisciplinary problem spaces.

### 1 Introduction

All features of SageMathCloud are available as soon as a free login is created. The following sections offer a tour of major features of SMC.

### 2 Collaboration

SMC collaboration features apply to both static documents and executable files. Supported file types include  $\text{\LaTeX}$ , Markdown, and HTML documents, as well as Sage worksheets, Jupyter notebooks, and Linux terminal sessions.

**Multiple users** SMC has simultaneous real-time collaboration with no explicit limit on the number of simultaneous users. Each user can customize the color of their cursor. Figure 1 shows an example of a second user collaborating during an edit session.

**Work through network interruptions** If your network connection temporarily fails, you can continue editing as long as you want, and your changes will be merged into the live document when you reconnect.

**Text and video chat** SMC has text chat on the side of each document. Unlike most other chat systems, you can include inline and display MathJax-compatible  $\text{\LaTeX}$  formulas and markdown in the side chat. Users may edit any past chat message. Other collaborators are notified of messages via a bell in the upper right. Figure 2 shows chat with  $\text{\LaTeX}$  beside file views.

Video chat is available as well by clicking a button in the side chat panel.

**Course management** SMC has an integrated course management system, which makes it easy to fully use  $\text{\LaTeX}$  with students in the context of teaching courses and workshops. For example, you can create a document template with questions, push it to all students, let them work on it, then collect it later, grade it, and return it.

**Multiple cursors for each user** If a user creates multiple cursors using command- or alt-click, all cursors are visible to other users.

### 3 Editing documents

**Nothing to install** Because SMC is a cloud service, there is no software to install locally and projects are reachable from any device with an internet connection. Files can be compiled online to PDF without the need for local installation of  $\text{\LaTeX}$  software.

**Document history and backups** Every edit (at 2-second resolution) is recorded and stored indefinitely in SMC's backend database. You can browse the history using the TimeTravel view, which also includes a mode showing exactly what changed between two points in time (and who made those changes). Figure 3 shows a TimeTravel view comparing two versions of a file.

SMC stores several hundred read-only snapshots of the complete filesystem state, which users can easily browse. This lets them recover older versions of files that might not have been edited via the web-editor (e.g., edited with vim or emacs via a terminal).

**Preview** SMC supports online preview, even for documents that are 150 or more pages. It progressively refines the resolution of the preview images and nearby pages. One of the SMC developers wrote the 2016 Cambridge University Press book *Prime Numbers and the Riemann Hypothesis* using SMC [2].

**Forward and inverse search** SMC supports *inverse search*, which means that by double-clicking on an area on a preview page, the cursor in the text editor jumps to the corresponding location. Similarly, you can jump from a point in the text editor to the corresponding point in the preview via *forward search*.

**Editor user interface** SMC has a full-featured text editor for  $\text{\LaTeX}$  documents. It has horizontal and vertical split view, which lets you look at two points in the document simultaneously. SMC developers wrote a code folding mode ([github.com/codemirror/CodeMirror/pull/4498](https://github.com/codemirror/CodeMirror/pull/4498)), so one can easily toggle display of sections, subsections, etc. Emacs, Vim, and Sublime key-bindings are supported along with many color schemes.

**Customizable  $\text{\LaTeX}$  build system** SMC fully supports processing very complicated  $\text{\LaTeX}$  documents using custom build systems, including several  $\text{\LaTeX}$  engines — `pdflatex`, `latexmk`,

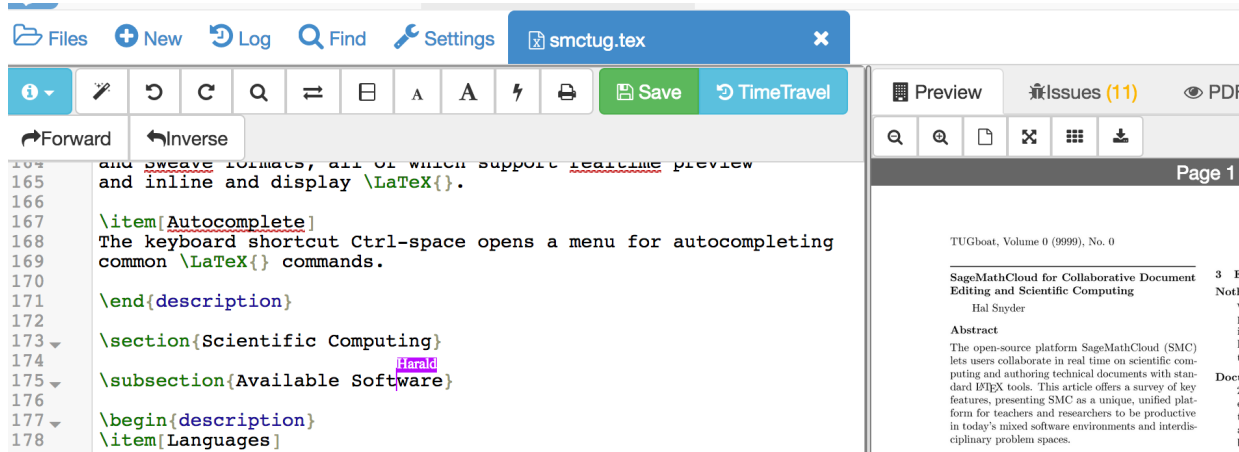


Figure 1: Editing .tex file, showing cursor of second user Harald.

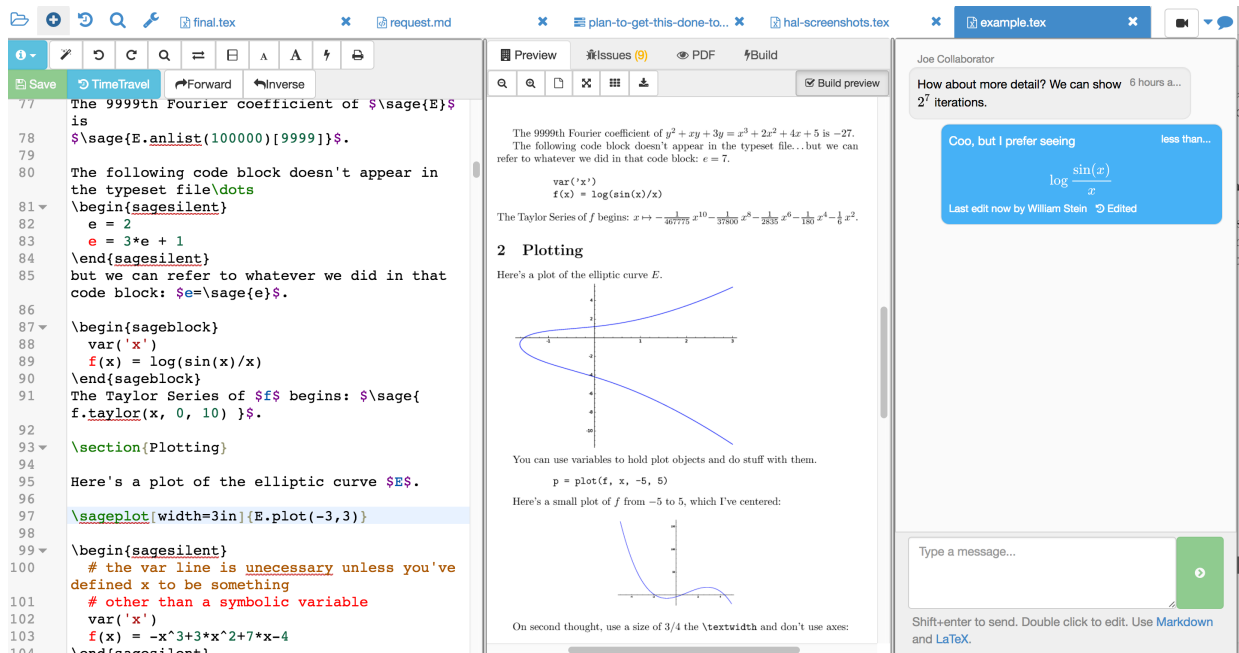


Figure 2: Editable chat beside file views.



Figure 3: Edit history showing difference between two versions.

and `xelatex` — with most packages preinstalled. Users can easily request additional packages (by clicking the help button), or install them themselves in their local projects. The  $\LaTeX$  build command is fully customizable, and can involve running arbitrary command sequences, since we offer a full Linux environment. It is, for example, even possible to use GNU `make` to orchestrate the full compilation via a `Makefile`.

**Dynamic content** Our  $\LaTeX$  editing environment comes with SageTeX, which makes it easy to add the output of Python (and SageMath!) computations to any  $\LaTeX$  document. Note that the plots shown in Figure 2 are dynamically generated using SageTeX commands. The figure displays an excerpt from the example file for the `sagetex` package by Dan Drake [1].

Besides SageTeX, SMC supports embedding R code via `knitr` into  $\LaTeX$  documents. This technique is popular for generating documents with statistical and data science content.

**Spell checking** In addition to normal spell checking, SMC has  $\TeX$ -aware spell checking.

**Other document formats** Side-by-side editing also supports HTML, Markdown, R Markdown, and Sweave formats, all of which support real-time preview and inline and display  $\LaTeX$ . Figure 4 shows SMC compiling R Markdown.

**Autocomplete** The keyboard shortcut `ctrl-space` autocompletes common  $\LaTeX$  commands.

## 4 Scientific computing

### 4.1 Available software

**Languages** SMC includes dozens of programming languages and thousands of libraries and packages, including computer algebra systems for theoretical mathematics, scientific packages for physical sciences and bioinformatics, and statistical and machine learning software for data science. Here are some of the supported languages: Python, Sage, R, Julia, C, C++, Haskell, Scala.

**Packages and environments** Popular packages and libraries are provided for each programming language. For Python, the entire Anaconda suite is available. Sage alone includes symbolic and numeric packages, including NumPy, SciPy, matplotlib, Sympy, Maxima, Pari/GP, GAP, R, and Singular. See Figure 5.

### 4.2 Programming frameworks

**Interactive notebooks** In the category of interactive notebook computing, SMC offers Sage worksheets and Jupyter notebooks, both of which

have chat and TimeTravel and support  $\LaTeX$  mixed with code. Sage worksheets allow users to easily mix different languages and document formats in different cells of a single worksheet.

**Terminal sessions** Any number of text-based terminals may be opened to the underlying Ubuntu operating system, providing access to standard Linux tools and command-line interfaces to programming languages.

## 5 Free and paid accounts

Free accounts provide unlimited projects and collaborators, with 3GB of disk space per project. SMC charges for upgrades, including more disk space, CPU power, and outside network access, which makes it possible to connect to the Internet from within a project in order to push and pull data to remote sites (e.g., GitHub). For more about pricing, see <https://cloud.sagemath.com/policies/pricing.html>

## 6 Further study

The best way to learn more about SMC is to create a free account at <https://cloud.sagemath.com> and try it out for editing and programming.

There is a wealth of tutorial and reference information at the SMC wiki at <https://github.com/sagemathinc/smc/wiki>.

Articles about new features and system internals can be found at the SMC blog, <http://blog.sagemath.com>. Some of the information in this article appeared in the SMC blog posting [3].

Source code is on GitHub at <https://github.com/sagemathinc/smc>.

## References

- [1] Dan Drake. `dandrake/sagetex`: embed code, results of computations, and plots from the Sage mathematics software suite (<http://sagemath.org>) into  $\LaTeX$  documents. <https://github.com/dandrake/sagetex>, 2017. [Online; accessed 2017-02-15].
- [2] Barry Mazur and William Stein. *Prime Numbers and the Riemann Hypothesis*. Cambridge University Press, 2016.
- [3] Hal Snyder. SMC for Collaborative  $\LaTeX$  Editing. <http://blog.sagemath.com/latex/2017/02/06/smc-for-latex.html>, 2017. [Online; accessed 2017-02-14].

◇ Hal Snyder  
Sagemath, Inc.  
`hsnyder (at) sagemath.com`  
<https://cloud.sagemath.com/>

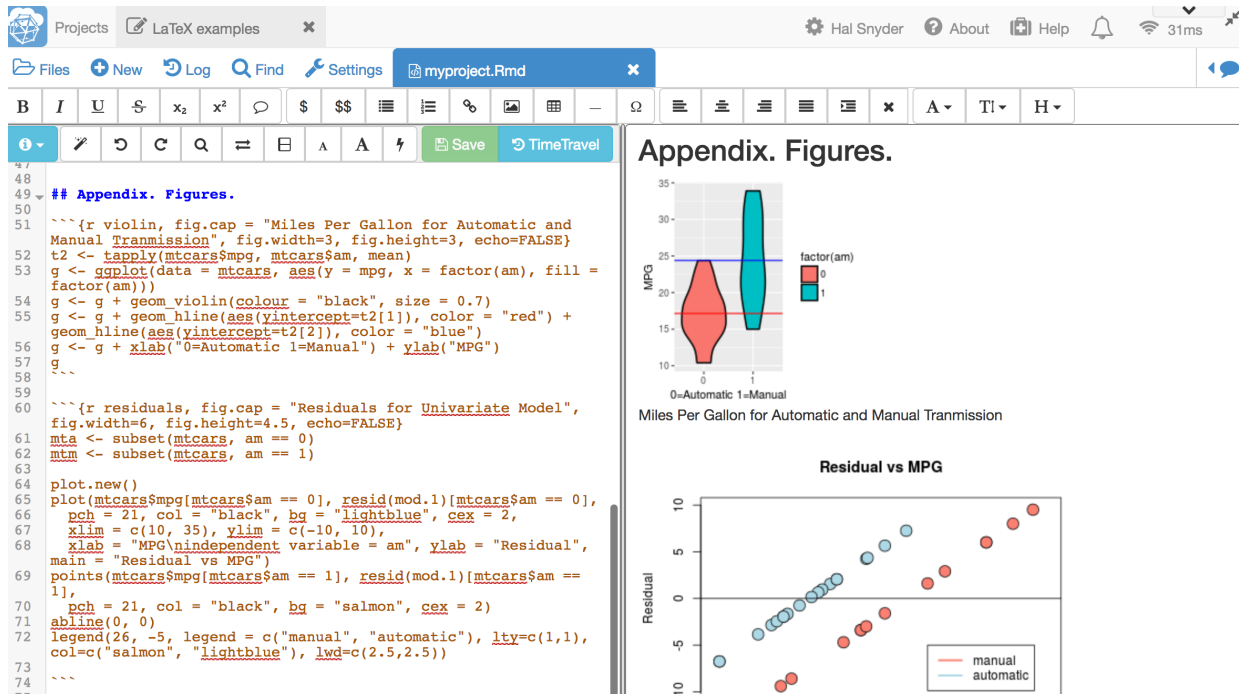


Figure 4: Compiling an R Markdown file in SMC.

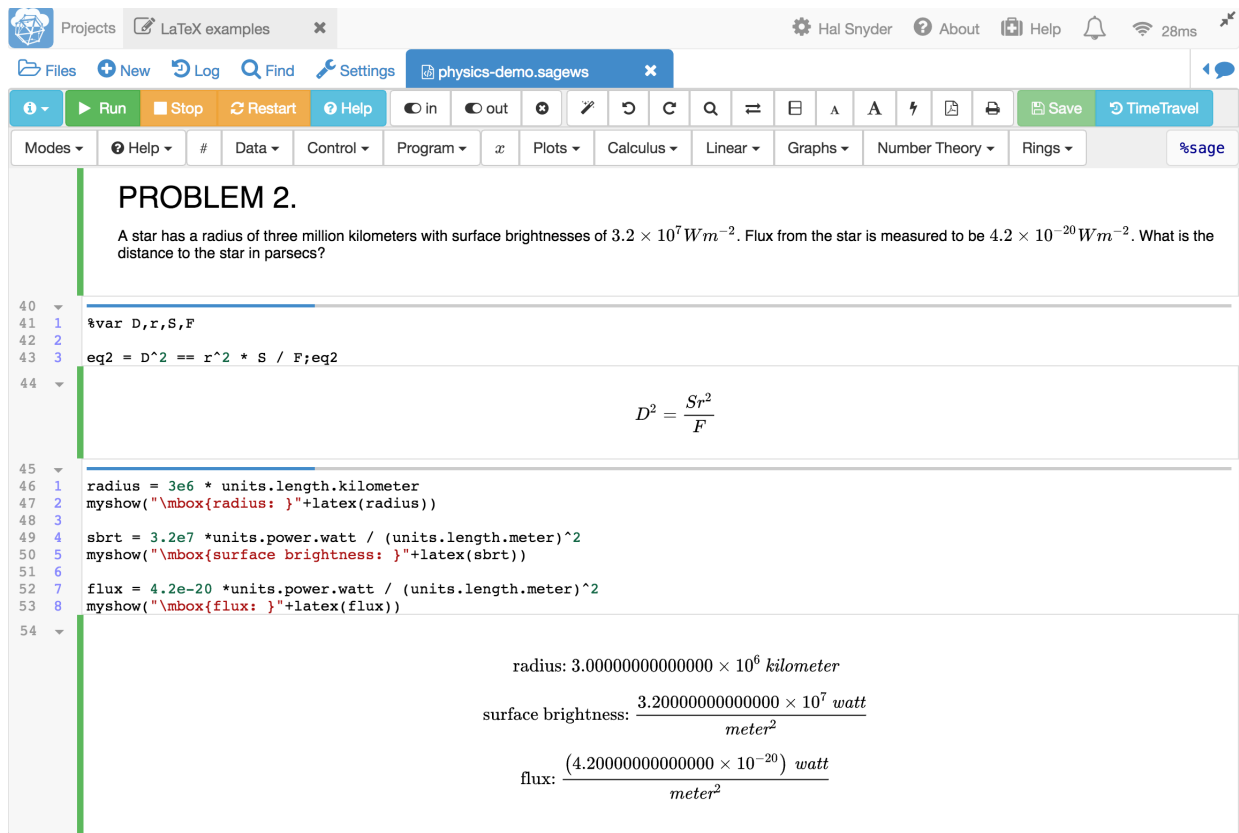


Figure 5: Sage worksheet with L<sup>A</sup>T<sub>E</sub>X in markdown, symbolic expressions, and units of measure.