CCLI PHASE 2: SAGE MATHEMATICAL SOFTWARE

Contents

Title: An Open Approach to Realizing the Potential of Mathematics Software in the Undergraduate Curriculum (shorter?)

1. Project Description

Software for mathematical explorations, often known as "computer algebra systems," have held great promise for mathematics education since the appearance of Maple in the early 1980's and Mathematica in the late 1980's. Studies have documented the improvement in students' understanding [Tom - back this up with one or two sentences]. Yet the use of these systems is rarely integrated well with the curriculum. To accomodate competing systems, textbooks typically offer supplements for several different programs or generic "technology" sidebars. Only rarely does a textbook take a chance by tying itself to the strengths of one particular commercial system. Licensing restrictions for campus use, the expense of personal copies, and/or underpowered hardware means students must often work with the software only in campus labs. The Calculus & Mathematica project generated much excitement in the early 1990's, but today its use seems limited to the two institutions where it originated. Why have we not seen the broad transformative effect of these powerful tools for increasing the understanding of mathematics? What is needed to increase the uptake of software for mathematical exploration and have it be an essential ingredient in the undergraduate mathematics curriculum?

Our proposition is that open software, open standards and open licenses can allow mathematics software to effect the long envisioned transformation of the undergraduate mathematics curriculum. We will test this hypothesis using the open source mathematics software, Sage (cite web, SIAM Review), integrated into existing open textbooks and new, shorter curricular materials, authored with LaTeX, the open standard for technical typesetting. By creating a system to easily convert textbooks and other materials into electronic Sage worksheets, we will place the full computational power of Sage *directly into* a student's text, usable at all times and from anywhere, simply via a web browser. For the institution and instructor, the cost and time-consuming inconveniences of commercial software are removed by free software, installed on servers accessible with no license restrictions. Further, the essential nature of open software means that curricular decisions and needs can drive the development of the software, with classroom teacher and software developer frequently being one and the same. Leveraging the inherent advantages of an open approach to software and text, the promise of mathematics software for education can be fully realized by faculty and students.

More specifically, we will build and test a model for easily integrating mathematics software and open educational resources into the mathematics curriculum and classroom by:

- integrating Sage with existing mature open textbooks and creating other open Sage-enhanced curricular materials, plus building a model and tools that allow others to easily do the same;
- partnering with diverse institutions to test and refine this model and materials in a wide variety of courses while providing support for teachers;
- conducting formative and summative evaluation to determine the effectiveness of our model in making open materials easy to use, and measuring the resulting impact on teaching practices and the learning of mathematics.

Materials enhanced or created in our project will of course be distributed with open licenses and made freely available as part of Sage's website (XX visitors a month), or as part of Sage itself (over 6,000 downloads a month). But more importantly, the process for creating these materials will be made as easy as possible for authors, and workshops will be organized to train new authors and further refine the process of creating Sage-enhanced textbooks, guides and demonstrations.

2. Mathematics Software Today

2.1. **Three crises in the academy.** The crisis in research publication is well-known (cite). Faculty have come a long way in their efforts to return scientific publication to a free exchange of ideas. Electronic journals and public repositories now publish articles with open licenses that explicitly allow for sharing new results among research communities and others, without copyright fees and with ubiquitous access via the internet. Government initiatives, such as the Public Access Policy of the National Institutes of Health, are working to accelerate this trend.

Following on the sea change in research publication, the next wave is open textbooks. Again, the problems with commercial publication of textbooks, such as high prices, edition churn, and orphaned works, are all too familiar to faculty. Open licenses are now being used to assert control of these critical resources for education, with faculty in mathematics, computer science and business as the leaders. Government is poised to similarly accelerate this trend at all levels, with a recent bill before Congress (cite Durbin) directing the Department of Education to award grants for the creation of open textbooks by faculty, Washington State's initiative to provide open textbooks for their eighty highest-enrollment courses in the community colleges, and California's initiative to create free digital textbooks for its high schools.

With the emergence of viable and comprehensive open source software for mathematics, there is now a spectacular opportunity for faculty to use, modify, shape, create, and extend this critical tool for mathematics education and allow it to reach its full potential in the service of educating students. This project will ease the path for faculty making the transition to open software and open textbooks in their courses. Tools for authoring materials using open standards, existing open textbooks, networks made ubiquitous by adherence to open standards, software built from open components and held together with an open industry-standard language, and a steadfast dedication to an open development process will combine to make it far easier for schools and faculty to deliver the full power of mathematics software to students. This proposal is an opportunity for government to accelerate these inevitable changes.

2.2. **Computer algebra systems.** Computer algebra systems (CAS) have held great promise for education since their first appearance in the 1960s (Reduce, Macsyma) and accelerated with the introduction of Maple in the early 1980s and Mathematica in the late 1980s. Currently, CAS are widely found in the undergraduate classroom and a considerable amount of mathematics education research has focused on the use of CAS in the learning of undergraduate mathematics (cite survey article by K. Heid or a more recent article). However, in many cases, computer algebra systems have driven the curriculum (cite Calculus & Mathematica project) and the result has not been the broad transformative effect on mathematics education that was envisioned. The closed development process creates a high barrier for teachers and students to alter or extend the software to meet curricular needs. For example, external extensions to proprietary systems, such as libraries or packages, can extend proprietary systems, but must be purchased, distributed, and installed by every end-user. However, an open development process, such as the open-source model that Sage follows, allows students and faculty to shape the core technology in a timely fashion to support the transformation of teaching and learning of mathematics on a broad scale. Open source software, such as Sage, allows the teaching and learning of mathematics to drive the technology, rather than the technology driving the teaching and learning.

2.3. **Choosing a computer algebra system.** Institutional, departmental and course-by-course decisions about the use of a CAS require a commitment to a particular vendor's vision for the curriculum and in some cases may even require a choice of supported operating systems. With a significant investment in hardware, training of faculty, and bringing a group of students up-to-speed on a particular product, there is a significant penalty to switching to another product that might be superior for certain courses. With open software, hardware choices and expenses become irrelevant when it may only require a student to have a web browser, and similarly the choice of operating system becomes irrelevant. As individuals and institutions extend the software, the entire mathematics community benefits from a comprehensive program that student and faculty can employ in courses ranging from pre-calculus and introductory statistics through to advanced courses like abstract algebra and number theory.

3. Why Sage Now?

Sage is the most natural choice for software that will realize the benefits of an open approach to the undergraduate mathematics curriculum. An open development process, the intuitive and familiar notebook interface, tight integration with LaTeX, a standard programming language and a modular design philosphy are simultaneously consequences of a steadfast commitment to open design principles and concrete examples of the benefits of this approach. With an open license, Sage removes substantial financial and logistical barriers to classroom use of mathematics software, such as purchase cost, restrictive licensing, expensive hardware, and limited availability on mobile devices.

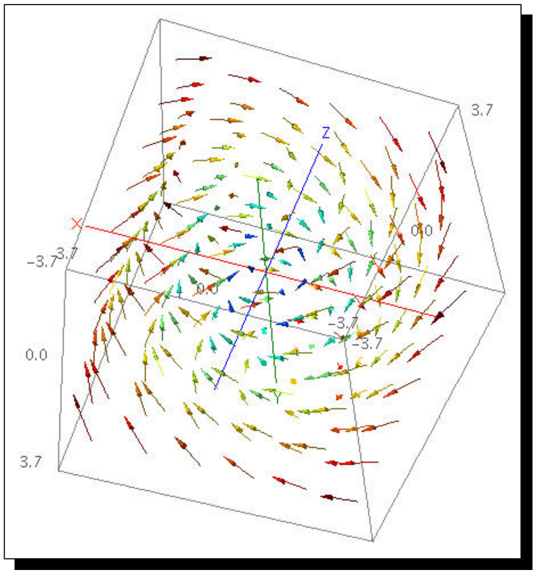
3.1. **Building the car, not reinventing the wheel.** The genius of Sage is the way it unifies over one hundred mature, best-of-breed, open source packages, each excelling at specific types of computations (e.g., the Integer Matrix Library (cites) for fast and exact matrix computations over the integers or rationals) or complete libraries for specific areas of mathematics (e.g., R for statistics, GAP for group theory, and SciPy for scientific computations) into a single open source system with a consistent interface. This unification greatly simplifies the use of mathematical software for both students and teachers.

3.2. **A standard programming language.** The many components of Sage are held together with a significant library of new code, written in the industry-standard language Python. More and more packages are appearing for mathematics and science written in Python, often with open licenses. So Sage continues to benefiit by easily integrating these new packages, bringing new functionality, or improving existing functionality, all with little disruption to the user.

Most importantly, with the decision to use Python to integrate the many components of Sage, it was entirely natural to make Python also the user language within Sage. This is in contrast to other comprehensive programs for mathematics that have chosen to create and develop their own languages, which are of little use outside of the program itself. A user with knowledge of Python is ready to be productive in Sage immediately, while a student new to programming receives a basic education in Python as a by-product of learning Sage — a skill that is readily transferable to a wide variety of applications in mathematics, science and engineering.

3.3. **A comprehensive program.** Sage aims to be a comprehensive program for mathematics. Its design allows numerical, exact and symbolic approaches to mathematics to coexist equally. Mathematical objects, such as functions, differential equations, rings, fields, modules, and vector spaces are "objects" that look and behave like their abstract mathematical definitions intend. While the relative strength of Sage in particular topics may vary through the copurse of its development, it does not rest on a foundation such as pattern-matching, which favors symbolic computation, or floating-point numbers, which favor numerical work. With a modular design that allows for packages specializing in different approaches, Sage can seamlessly support many approaches to the application of computation to mathematical problems. Further, additions and extensions to Sage, driven by real curricular needs can be easily and quickly incorporated.

3.4. **Curriculum and open development.** Like mathematics, the Sage model for development is driven by collaboration, sharing and openness. Just as major research problems drive the direction of pure mathematics, the curriculum and needs of the users directly drives the development of open mathematics software. As an example of how teaching has driven the development, Jason Grout was asked one day by a student if Sage had a 3D vector field plotting function. Jason quickly wrote one. Open development communities encourage sharing even experimental code, so he posted his code online in Sage's database of enhancements and bugs. Another developer that had written most of the 3D graphing in Sage suggested a small change that greatly increased the efficiency of the function. Several months later, another person authored a very similar function which had slightly improved options and published their code on the public Sage notebook server. In Fall 2008, Jason needed the 3D vector field plotting function for his multivariable class. He made a few changes to the code based on suggestions, and posted it for his class to use. In a closed proprietary system, that is where things likely would have stopped. Full stop.



The vector field
$\vec{f}(x, y, z) = y\vec{\imath} + z\vec{\jmath} + x\vec{k}$
(fix placement later)

Since then, several other deep changes to the internals of Sage had been made to simplify code in functions like the 3D plotting, some of which were directly the result of feedback about the function. Jason then taught the same class again. He incorporated the best ideas from the various sources and posted a documented, efficient version of the function to the database, this time asking the community for a formal review (a prerequisite for his code to be added to the core Sage library). Another Sage developer reviewed the function immediately, with the comment "Very nice, positive review. This is great timing since I am about to teach vector fields in a week or two." The function was incorporated into Sage and due to the frequent release cycle for the project, it was included in the next version two weeks later, ready for immediate download by anybody.

Now not not only do the original students benefit from the invested effort, but the collaborative effort of at least four people, from around the world, has led directly to a function that can, and will, serve all students and teachers using Sage, now and in the future.

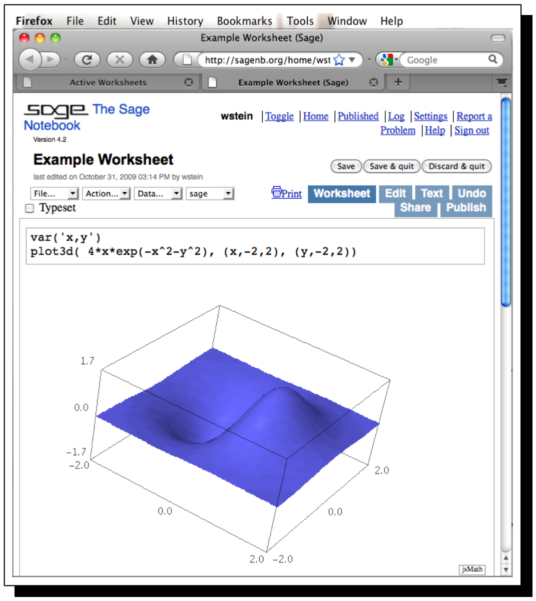
(Another example is the statistics functionality that (may?) be merged from the people in Europe that are using Sage to teach statistics?? Probably not room for this.)

3.5. **Sage anytime, anywhere.** Sage's notebook interface is perhaps its most impressive feature, especially for use in education. Much of the power of the Sage notebook is a direct result of decisions to employ open software and open standards in its design. From within a standard web browser (preferably one supporting open standards, such as Firefox), a user can create a "worksheet" and interactively edit and execute new commands, with computations performed remotely on a server, and output returned to the worksheet. The interface makes it easy to view graphical output, annotate a worksheet, receive help on commands, browse the manuals, browse the source code, manage multiple worksheets, publish completed worksheets, or share and collaborate on worksheets.

The Sage notebook interface relies heavily on industry-standard Javascript, acting as an application similar to GMail or Google Wave. Much of the development work on the notebook has been student designed and implemented, including at the extreme very significant contributions this past fall by Tim Dumol, a high school student in the Phillipines.

The Sage project hosts a public Sage server at `sagenb.org` which currently has over 20,000 accounts. So right now, anybody in the world with a network connection can productively harness the full power of Sage — even with minimal hardware or mobile devices, such as an underpowered desktop, a laptop, netbook, smartphone, or cell phone. (Cite Korean cell-phone interface.) Wide-open notebook servers continue to appear throughout the world [Hungary 2009/11/25, http://sage.math.u-szeged.hu/], while many more run behind campus firewalls for dedicated use. Public or private servers make hardware and operating decisions irrelevant. Underpowered hardware can control a web browser sufficiently to handle communications with a Sage server, and the use of portable devices means students can access Sage from anywhere they can get a network connection. Notice too, that it is trivial to install and run a Sage server on your own machine, so Sage can also be used productively on reasonably modern hardware without a network connection.

3.6. **Communicating mathematics.** TEX, with its add-on package LATEX, is another open source success story and is the typesetting language of choice for mathematicians and other technical disciplines. As a result, there is an impressive array of extensions to LATEX. Sage also includes extensive support for LATEX, through every mathematical object being *required* to have a LATEX representation. The Sage notebook then uses jsMath (cite) to render mathematics beautifully in a Sage worksheet (within a standard web browser) via Javascript control of the necessary fonts and their accurate placement. Furthermore, the Sage notebook interface allows a user to insert new text, including LATEX code that jsMath will similarly render accurately. This is another example of how open standards and open software serendipitouly combine to make powerful tools, which is especially relevant for our plan to convert LATEX documents into Sage worksheets.



Example Worksheet

last edited on October 31, 2009 03:14 PM by wstein

□ Typeset

```
var('x,y')
plot3d( 4*x*exp(-x^2-y^2), (x,-2,2), (y,-2,2))
```

The Sage notebook interface
(fix placement later)

3.7. **Freedom and liberty.** Open source software is more than about being free, as in no cost. It is about freedom — the liberty to use the software they way you want to use it, and in ways the developer might not have ever even imagined. If you make improvements, or worthwhile extensions, then it should be easy (as it is with Sage) to contribute those improvements back. At a minimum, license terms often require you to at least make your distributed improvements available publicly to everybody.

3.8. **Freedom and budgets.** But Sage is also free – as in no cost. There is no cost to download and install Sage. There are no costs for add-on packages. There are no costs to upgrade to the latest version. There is no reason to have incompatible versions installed in different locations. So even institutions with highly constrained budgets can have their students work on public servers such as `sagenb.org` if they have even a minimal network connection. Imagine students at the poorest universities running the same version of Sage, on the same hardware cluster at the University of Washington, side-by-side with leading researchers in computational number theory.

Certainly, there are costs to installing a campus Sage server. But the hardware does not need to be particularly expensive or extravagant. There are costs to learning a new program for faculty, and for students. But this is a big part of this project — just how simple can we make it for an institution, department or course to enjoy the benefits of open mathematics software and open curricular materials? With an open approach, our successes will be easy to distribute and replicate.

4. Implementation

Our goal is to provide a broad, positive influence on the teaching and learning of mathematics. By using open curricular materials, integrated with powerful and comprehensive open software, we will realize a transformative effect. Project Sage will revolve around three coordinated efforts: (1) creating Sage-enhanced curricular materials, (2) providing the Sage infrastructure to support these materials, and (3) working with, and providing classroom support for, teachers using these materials in the classroom for the first time. More specifically our proposal consists of five main activities.

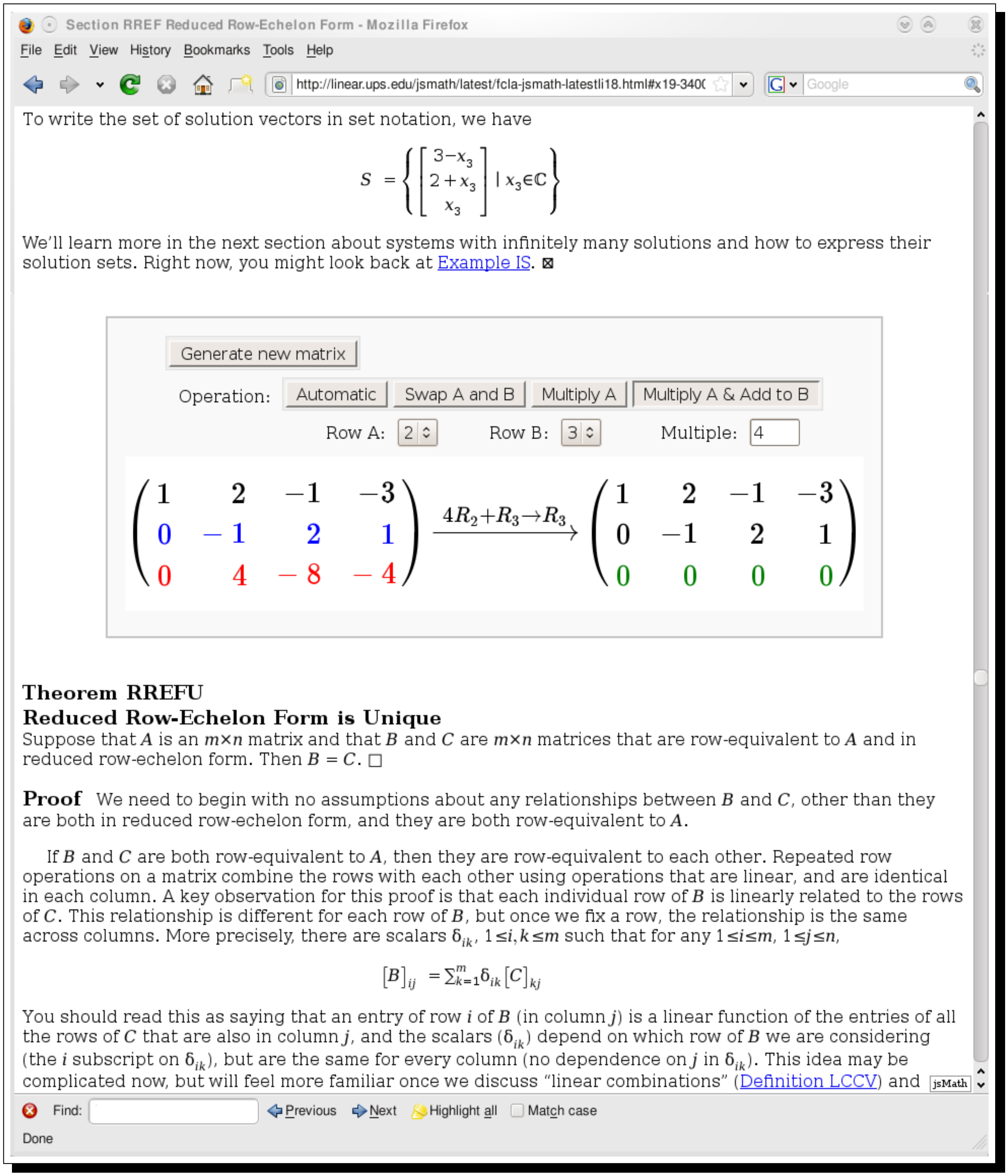
- Convert existing open textbooks to Sage worksheets, and incorporate interactive demonstrations powered by Sage, along with guidance on how to use Sage to further understand the topic of the textbook.
- Create new curricular materials for standard undergraduate courses, especially for courses not covered by the textbooks we work to convert. These will include an organized collection of interactive demonstrations of high technical and pedagogical quality, plus subject-specific guides to introduce Sage to both faculty and students in the context of a particular topic.
- Improve the Sage library and its surrounding infrastructure (e.g., server design, notebook usability, collaboration tools) where the improvements have a direct and obvious benefit for the use of Sage in educational settings.
- Identify and work closely with ten to fifteen (number???) different undergraduate institutions to test, refine, improve and extend the work described just above. These institutions will be chosen to represent as diverse as possible an array of student populations and institutional profiles.
- These materials will all carry open licenses and be available through the Sage website, which has a global audience. Much, or all, of what we do will be incorporated into Sage itself. Sage Days workshops, MAA PREP workshops, and MAA minicourses (national and regional) will be utilized as opportunities to train faculty (outside of the fifteen test sites) in the use of these materials for their courses, and in the creation of new materials.

4.1. Sage-enhanced open textbooks.

4.1.1. *Imagine this.* A student is learning about row-reducing a matrix in a beginning linear algebra course. The electronic version of their textbook is an online Sage worksheet they can view from anywhere in the world. Mathematical equations, with publication-quality formatting, describe the procedure. An interactive demonstration, embedded in the text where an example normally would be, allows the student to step through row-reducing a small matrix, generated on-the-fly. [cite mock screenshot in appendix] When prepared to guide the procedure themselves, the student may choose the row operation to apply at each step of the reduction. In either case, the correct notation for the operation is displayed and the operation itself is highlighted with color-coded entries of the displayed matrix. Another example shows the student how to use a built-in command to row-reduce a matrix. With a single click, the student creates an empty code cell under the example to experiment with the command.

In class, the instructor does one simple example on the board. Then the instructor opens a Sage worksheet projected in class containing the same demonstration that is in their electronic textbook. Students guide the choice of row operations at each step for several example matrices of increasing complexity. At the end of class, the instructor clicks a button to publish the work from that day's class session to the class Sage server. After class, students view the worksheet as they review their notes. They can also, with a click, copy the worksheet into their notes, as well as annotate or change it.

The book's exercises include an interactive problem generator that creates matrices of student-specified sizes and complexity for the student to row-reduce. There is a "Solution" button which


Section RREF Reduced Row-Echelon Form - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://linear.ups.edu/jsmath/latest/fcla-jsmath-latestli18.html#x19-340(
Google
jsMath
Find:
Previous
Next
Highlight all
Match case
Done

To write the set of solution vectors in set notation, we have

$$S = \left\{ \left[\begin{array}{c} 3-x_3 \\ 2+x_3 \\ x_3 \end{array}\right] \mid x_3 \in \mathbb{C} \right\}$$

We'll learn more in the next section about systems with infinitely many solutions and how to express their solution sets. Right now, you might look back at Example IS. ⊠

Generate new matrix

Operation: Automatic | Swap A and B | Multiply A | Multiply A & Add to B

Row A: 2 Row B: 3 Multiple: 4

$$\begin{pmatrix} 1 & 2 & -1 & -3 \\ 0 & -1 & 2 & 1 \\ 0 & 4 & -8 & -4 \end{pmatrix} \xrightarrow{4R_2+R_3 \to R_3} \begin{pmatrix} 1 & 2 & -1 & -3 \\ 0 & -1 & 2 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

Theorem RREFU
Reduced Row-Echelon Form is Unique
Suppose that A is an $m \times n$ matrix and that B and C are $m \times n$ matrices that are row-equivalent to A and in reduced row-echelon form. Then $B = C$. □

Proof We need to begin with no assumptions about any relationships between B and C, other than they are both in reduced row-echelon form, and they are both row-equivalent to A.

If B and C are both row-equivalent to A, then they are row-equivalent to each other. Repeated row operations on a matrix combine the rows with each other using operations that are linear, and are identical in each column. A key observation for this proof is that each individual row of B is linearly related to the rows of C. This relationship is different for each row of B, but once we fix a row, the relationship is the same across columns. More precisely, there are scalars δ_{ik}, $1 \leq i,k \leq m$ such that for any $1 \leq i \leq m$, $1 \leq j \leq n$,

$$[B]_{ij} = \sum_{k=1}^{m} \delta_{ik} [C]_{kj}$$

You should read this as saying that an entry of row i of B (in column j) is a linear function of the entries of all the rows of C that are also in column j, and the scalars (δ_{ik}) depend on which row of B we are considering (the i subscript on δ_{ik}), but are the same for every column (no dependence on j in δ_{ik}). This idea may be complicated now, but will feel more familiar once we discuss "linear combinations" (Definition LCCV) and

Prototype of a Sage-enhanced mathematics textbook
(belongs in an appendix)

generates a step-by-step solution. A group exercise asks students to collaboratively write and test a short function to implement row-reduction using row operations. The student clicks a button to open a new Sage worksheet and types in a few lines using Sage matrices and commands in Python. The student evaluates the code with a single click. Satisfied with their work, the student clicks another button, publishing the worksheet to the rest of the group for testing. Later that evening, the student checks the worksheet and sees that other students have tested the function, found a typo, and fixed it.

An advanced exercise guides a student through investigating numerical issues that arise in row reduction. By changing one parameter within a command, the matrices track numeric error bounds via interval arithmetic, and the student discovers that numerical errors can be a significant problem for some matrices. (Ditch advanced exercise if space dictates.)

The next week, the instructor goes to the online server and looks over all of the published worksheets for the group projects. The instructor annotates each worksheet online, using an embedded word processor that supports mathematical typesetting.

The instructor writes a quiz covering row-reduction techniques. Inside of the LaTeX code, the instructor embeds a Sage command to create a simple matrix in reduced-row echelon form, which will be the *answer* to a quiz problem. The instructor then calls Sage (automatically from within his LaTeX document using Dan Drake's SageTeX package(cite)) to convert the answer to a matrix that forms the question and only requires a short sequence of simple row operations for a solution. The question matrix is automatically incorporated in the printed quiz, while the answer matrix and a detailed solution are automatically incorporated in the answer key.

4.1.2. *Infrastructure for textbooks.* (reprise tex stuff from above) Many mathematicians and scientists use the LaTeX language to create technical articles and books.

Our project proposes to convert open-source textbooks into folders of interactive Sage worksheets from existing mature open textbooks. The tex4ht translator (an NSF funded project) can convert extremely complex LaTeX into jsMath, which is then simple to convert to the Sage worksheet format since it relies heavily on HTML and jsMath. It is possible to insert Sage code into a LaTeX source file, and have it migrate to the eventual Sage worksheet as an input cell. The end result is a folder of Sage worksheets, each a mixture of text, high-quality typeset mathematics, Sage input cells and Sage interactive demonstrations.

4.1.3. *Targeted textbooks.* Section 4.1.4 lists many open textbooks that could be converted by our process. However, conversion is more that a technical process. We plan to incorporate interactive demonstrations, live Sage code, and guidance on the Sage library itself. So there is a significant care and thought required to integrate these uses of mathematical software properly.

Listed below are our initial candidates for modeling how we can incorporate live code and interactive demonstrations to bring the full power mathematics software to the student *directly within their book.* The open textbook landscape is changing rapidly, so it will be best to decide later which books make the most sense. (Should we commit to a certain number here? With the sabbatical to get started, I'd expect to work on one or two more. Others? Jason - Woodruff notes? William - Riemann Hypothesis? Other volunteers, other books?)

- Beezer's *A First Course in Linear Algebra.* This is a mature textbook designed from the start to be open-souce, and thus has a highly modular design. It will be an excellent choice as a full-scale test of the technical conversion process, the integration of Sage code and classroom use.
- Stein's *Elementary Number Theory.* This text already has significant use of Sage, and the author is the founder of the Sage project, making it another ideal candidate for early conversion.
- Judson's *Abstract Algebra.* This is another text authored by a project member, and Beezer has taught from this text the past two years, creating supplementary material utilizing Sage, in addition to already contributing code to the Sage library to support the book. As a year-long course, this is a larger project to tackle next, and will be a collaborative project betweeen two team members.

4.1.4. *Other open textbooks.* Beyond our initial candidates there is a wide range of books available that could be converted to cover portions of the standard undergraduate curriculum. With one exception, each of the textbooks below is licensed with a GNU Free Documentation Licenses (GFDL) or a Creative Commons License that allows anyone to distribute modified versions, typically only requiring attribution of the original author's work and the use of the same license. New open-source textbooks continue to be released and evidence suggets many authors are considering the option seriously.

(reorder?)

- *A First Course in Linear Algebra*, Robert Beezer
  Second-year university level text, concentrating on proof techniques. Used at thirteen schools since Fall 2007. Contains some Sage commands.
- *Abstract Algebra: Theory and Applications*, Tom Judson
  Standard upper-division treatment of groups, rings, domains, fields, Galois Theory. Used by seven schools in its first year available as open-source.
- *Combinatorics Through Guided Discovery*, Kenneth Bogart
  Problem book in combinatorics with roughly 400 problems designed to teach the subject. Released as open source (NSF Grant No. XXXX).
- *Elementary Number Theory: Primes, Congruences, and Secrets*, William Stein
  Classical elementary number theory and elliptic curves, with applications and extensive Sage. Published by Springer-Verlag, open-source release May 2010. (NSF Grant No. 0653968)
- *Vector Calculus*, Michael Corral
  Standard treatment of multivariate calculus. Topics include vector calculus, partial derivatives, multiple integrals, theorems of Green and Stokes.
- *Trigonometry*, Michael Corral
  An in-depth, comprehensive and unified treatment of the standard topics.
- *Cryptography*, David Kohel
  Classical ciphers and their cryptanalysis, modern stream ciphers and public-key cryptography. Significant appendices on Sage.
- *What is Riemann's Hypothesis?*, Barry Mazur, William Stein
  Richly illustrated elementary treatment of a central problem of modern mathematics. Backed up with extensive Sage worksheets already. (NSF Grant No. 0653968)
- *Numerical Analysis*, Steven Pav
  Comprhensive book-quality lecture notes. Extensive use of Octave, an open-source clone of Matlab.
- *Calculus: Modeling and Application*, David A. Smith, Lawrence C. Moore
  Calculus textbook with significant electronic components. Mathematical Association of America project, currently copyrighted, all rights reserved. (NSF Grant No. 0231083)

4.2. **New curricular materials.** This project is not about creating new textbooks, which is why we are starting with existing open books. But targeted materials will make it easier for faculty to quickly use Sage in their courses.

4.2.1. *Subject-specific guides.* For selected courses where we do not provide a Sage-enhanced textbook, or no suitable book currently exists, we will create subject-specific tutorials. These short guides will explain how to use Sage to investigate questions in a standard course, following a typical order for such a course. While the target audience may be faculty, the guides will also be of use to students. Stipends will be provided to faculty outside the project team, ideally from our test sites, to author these materials. These will be designed from the start for conversion to Sage worksheets, providing a test of our system for this conversion.

4.2.2. *Interactive demonstrations.* Sage has a powerful, but extremely simple-to-use framework for creating interactive demonstrations. Input boxes, sliders, selectors and other controls are created simply, and may be layed out automatically or with guidance from the designer. Output can use text that is HTML or LaTeX along with 2D or 3D graphics. Computations can involve any command from the Sage library, combined in the form of a Python function. Stein created the first implementation and continues to improve it with significant contributions from Grout.

We will create many such demonstrations, for inclusion in Sage-enhanced textbooks and into Sage itself. It should be possible to create a topical library of high quality, reviewed Sage interacts directly into Sage. Here they could benefit from Sage's automated test system, so they would remain reliable.

4.3. **Sage infrastructure.** While the heart of Sage is its library of commands, the notebook interface and server configurations are key elements of a successful experience for students and faculty using Sage in their courses. So a portion of our work will be to improve Sage itself, in those areas where the improvements *directly support* educational applications of Sage. Professors at dozens of universities around the world are getting excited about how they can leverage the Sage notebook in their teaching.

> With some colleagues in our universitywe have built a project around Sage for undergraduate students… *And the University has decided to support this project.* Good news.
>
> Now we will be facing the problem to build a Sage configuration which will work for say 200 students at the same time (students will use the notebook), and prepare professors for Sage teaching. There are 'some' technical problems to solve…" (Thierry Dumont, University of Lyon, on the Sage mailing list.)

4.3.1. *The Sage library.* Experience has shown us that teaching with Sage invariably suggests new capabilities or exposes needed commands. So we will perform the one-time work needed to unify and extend the code relative to functions that are commonly used in a particular course. This will provide a more consistent and unified experience for the student. We have identified specific needs in this area for each of linear algebra, abstract algebra, calculus, differential equations, number theory, and combinatorics. Plotting is used many places in Sage, especially in calculus. This is another area of active development, where targeted fixes and improvments can vastly improve the educational experience for students.

In many cases this is work that can form projects for undergraduate students, so this is an avenue for incorporating student input to improve Sage.

4.3.2. *Notebook development.* The Sage notebook interface allows students to use the full power of Sage on a remote server with only a web browser and a network connection. Even when Sage runs on a single unnetworked machine the notebook is a powerful tool for experimentation and collaboration. So continued improvement of the notebook will translate directly into an improved student experience.

Using hardware purchased using an NSF SCREMS grant (DMS-…), William Stein created a free public Sage notebook server at `http://sagenb.org` in 20xx (?). Stein worked fulltime during Fall 2009, supported by the University of Washington, on vastly improving the robustness and scalability of the notebook. This work came at the right time, because use of the Sage notebook server has been growing rapidly. Typically, in Fall 2009, we have seen several hundred new accounts created on the public server every day.

4.3.3. *Sage servers the easy way.* Sage servers are critical for user's independence from specific hardware and operating systems, given the prevalence of web browsers and network connections. So initiating and maintaing a server should be as simple as possible, making it easy for faculty and sysadmins everywhere to ge their students started with Sage.

We will create a Virtual Box Sage notebook server appliance with a graphical interface, two virtual machines that provides a rock solid secure Sage notebook server setup. Users will be able to very easily install this appliance on Windows, Linux, OS X, and Solaris (x86) servers. The management interface will provide clear and easy documentation about setting up this server, creating new notebook servers for specific classes or instructors, starting and stopping notebook servers, monitoring resource usage, adding users and authentication frameworks (such as LDAP), and upgrading Sage with minimal user downtime.

For users at our selected test-sites we will create a new mailing list for server administration issues, and in addition to the help we provide ourselves we will hire a knowledgeable employee for a certain number of hours per day to answer support questions.

Several specific goals for notebook and server development include:

- Improve the notebook so that it will robustly handle up to 1000 simultaneous users when running on a single high-end server, as demonstrated by a robust automated test suite
- Implement management tools so administrators can manage the notebook load and better balance resources
- Create easy-to-use research tools for educators, so they can understand how their students use Sage

In summary, we propose to provide high-quality enterprise level software infrastructure and support to educators. What we propose is far from free or easy — it is in fact expensive. But with support now, literally millions of students will benefit at no cost to them. This is an excellent investment in mathematics education for the National Science Foundation.

4.4. **Test Sites.** The heart of Project Sage is placing Sage-enhanced materials in the hands of teachers and students to assist them in the teaching and learning of mathematics. We will work with selected faculty at test sites to provide support for using Sage in their classroom, for the Sage-enhanced materials that we have created, and teach them how to write their own materials. We will recruit two cohorts for our test sites with the first cohort beginning Summer 2011 and the second beginning the following summer. Several institutions have already expressed interest. [Note: We need to list some of these places or even attach letters of support expressing interest.] The teachers at our test sites will receive the following support.

- Teachers will commit to using Sage-enhanced materials in their classrooms for two years.
- Teachers will write and test a Sage-enhanced model for the class that their are teaching.
- Teachers will write and file a report at the end of each year of their two-year commitment.
- After the first year, teachers will train and conduct departmental workshops on Sage and how to use Sage-enhanced materials in the classroom. Teachers in the first cohort will also mentor teachers in the second cohort.

Project Sage will offer the following support for our teacher-participants.

- Teachers will receive a $5000 per year stipend for participating in Project Sage.
- Teachers will receive Sage support as well as support for Sage- enhanced materials and access to a Sage server.
- For each site a selected member of the project team will serve as liason to that site, which will include site visits for help and dissemination. Each member of the project team will serve in this role for two different sites.

5. Preparation, Qualifications

The five faculty in the Project Sage team together have XX years of experience teaching undergraduates at a wide range of institutions, four are active Sage developers (including its founder), three are authors of open textbooks, all have significant mathematics research experience, and one specializes in mathematics education research. Everyone is familiar with, and has extensive experience using, open source software and tools in their teaching and other professional activities. Besides the applicability of this experience to the project's central theme, they are also extremely adept at working collaboratively while still being separated geographically. Working together, they have the experience and skills to make this project a success.

5.1. **Personnel.** **Dr. William A. Stein**, Associate Professor of Mathematics at the University of Washington, is

Dr. Kiran S. Kedlaya, Associate Professor of Mathematics at Massachusetts Institute of Technology and University of California, San Diego, is a leading researcher in arithmetic algebraic geometry. He has received an NSF CAREER grant, a PECASE award, and a Sloan Fellowship, and was selected to speak at the 2010 International Congress of Mathematicians. Dr. Kedlaya brings extensive experience in undergraduate and graduate education at all levels, from calculus to mentoring PhD students. He is also deeply involved with mathematics competitions and other programs for exceptional students, such as the USA Math Olympiad, the Putnam competition, and the Art of Problem Solving Foundation. In addition, Dr. Kedlaya brings extensive knowledge of the Sage system, having been an active user and developer since 2005. Dr. Kedlaya will review improvements for Sage related to the project, supervise test sites, and provide additional overall guidance.

Dr. Jason Grout, Assistant Professor of Mathematics at Drake University, is an undergraduate teacher with 5 years of experience, an active researcher in combinatorial matrix theory and graph theory, and a Sage developer. Dr. Grout has contributed extensively to the Sage code base over the last three years in the linear algebra, graph theory, plotting, and notebook interface components of Sage. For example, in the notebook, he helped design the framework for interactive demonstrations and integrated an open source Javascript word processor into the notebook. Dr. Grout has used Sage extensively in research and in teaching a number of undergraduate courses. He has also given numerous presentations and tutorials on Sage, and in Summer 2010 will be co-directing an MAA PREP workshop with Dr. Beezer and Dr. Karl-Dieter Crisman, to help participants develop new curricular materials in Sage. Dr. Grout will help direct and implement the infrastructure changes needed for Sage to display and use online textbooks and enhance Sage's usefulness to students and faculty in the classroom. He will also help direct the interactions with the mathematics community via workshops and presentations. Dr. Grout will also contribute a large and growing personal library of Sage curricular materials for undergraduate classes, and test and refine the process for converting textbooks to Sage-enhanced materials by converting a set of extensive curricular materials for multivariable calculus that he has been developing with Dr. Ben Woodruff.

Dr. Robert Beezer, Professor of Mathematics at the University of Puget Sound, is a undergraduate teacher with 31 years of experience, an active researcher in algebraic graph theory, one of the first open textbook authors and a Sage developer. He began writing his open source linear algebra textbook in 2004 and has assisted Dr. Judson with the recent release of his very successful open-source abstract algebra text. He began using Sage in 2007 and began contributing code in early 2009. He will lead the technical process of converting textbooks from LaTeX to Sage worksheets, producing a simple system for other authors to use. The pedagogical implications of this new capability will be explored as he incorporates Sage into existing textbooks on topics he teaches frequently, such as linear algebra, abstract algebra, combinatorics, calculus and cryptography. He will continue to contribute code to Sage where the new functionality enables a more complete experience for an undergraduate student, and will suggest, review and test the project's improvements to interactive demonstrations and the notebook interface.

Dr. Thomas Judson, Associate Professor of Mathematics at Stephen F. Austin Sate University, is an active researcher in both mathematics and mathematics education. He is the author of an open-source undergraduate abstract algebra textbook, and will work with Dr. Beezer to produce Sage-enhanced materials for abstract algebra. Dr. Judson has worked extensively with undergraduate mathematics teachers and has mentored graduate students in the teaching of mathematics. In addition to his teaching and research accomplishments, Dr. Judson brings added experience in working with diverse groups both in the U.S. and abroad. As such, he will work with Dr. Hassi to guide research and evaluation efforts for Project Sage. With experience in working with undergraduate faculty and graduate student mentoring, Dr. Judson will also provide insight and assistance in the implementation of Sage-enhanced materials in the undergraduate classroom. Project Sage efforts connected to research in mathematics education will be under the direction of Dr. Judson.

Tip to Maraj-Lisa, Sandra, etc. - cite evaluation section.

5.2. **Preparation.** Sage has come a long way in a short time. There are still a few rough edges, some of which we will polish as part of this project. But with over 200 developers and many thousands of users, its development pace will only increase. Members of the project team have significant experience with open textbooks, with using Sage in the classroom, and combining the two. For example, as a pilot project, a short primer on using Sage to study group theory was designed by Beezer to accompany Judson's abstract algebra text. This experiment clearly shows the feasibility of this process and clearly identifies the technical changes needed to make the process simple and routine for an author.

So the time is right to translate the experience of the project team into forms that faculty at other institutions can use to similarly benefit from teaching mathematics to their students with open mathematics software and open textbooks. Further, members of the project team, as Sage developers, are in a position to quickly and accurately shape changes in Sage based on the experience of working with faculty at other institutions that are new to Sage. National organizations, such as the MAA, are recognizing this trend by their sponsorship of WebWork and their support of a PREP workshop for Sage.

5.3. **Prior and current support.** (This is still a bit rough)

- Stein has successfuly administered many grants supporting Sage development from organizations such as UCSD, UW, Google, Sun, Microsoft, US Department of Defense. National Science Foundation grants include SCREMS for Sage cluster, and (what else).
- Judson has-been/is on several NSF education related grants.
- Beezer has been awarded a competitive year-long sabbatical leave from the University of Puget Sound for the 2010-11 academic year. This will allow him to begin on converting textbooks, maximizing the possibilities for test sites beginning in the Fall 2011 term.
- Some of Grout's recent work on Sage been partially supported by an NSF postdoctoral appointment in Summer 2008.
- Beezer and Grout will co-direct an MAA PREP workshop on Sage in Summer 2009,as part of a program funded by the NSF (Grant No.).

6. Dissemination

This project is principally about dissemination: how can enable the widespread implementation of approaches which have already been proven to work.

We will broadly disseminate the results of Project Sage through a variety of forums, several of which are already in place. The Project Sage curricular materials will be accessible on the Sage website, `http://www.sagemath.org/`, where users can read documentation, make contributions to Sage, download Sage for free, and keep abreast of new developments in Sage.

The Sage website (cite) is an established venue for sharing materials related to Sage. So what we learn and create will be made available here with open licenses. Further, some of our work will be to Sage itself, which already has an elaborate distribution system, designed to make it extremely easy to download and run Sage, on a personal machine or a server. Complete, integrated documentation ships with every copy of Sage (such as the 4,641 page reference manual), and it is possible that much of our material, *including complete textbooks*, could ship as a core component of Sage itself. Standard interactive demonstrations may also find a home in Sage itself where they could be organized topically, be searchable, and benefit from Sage's automated test-suite that users routinely perform on a wide variety of hardware.

Sage Days, is a series of conferences and workshops devoted to the development of Sage, are an ideal venue for dissemination. To date there have been nineteen Sage Days conferences, with nine more planned, and these workshops now include sessions dedicated to using Sage in the classroom. A Sage Days devoted to training others to use our materials, and receiving and incorporating feedback and new ideas, will be entirely isomorphic to the way Sage Days have been used to drive Sage development. (Are we committed to some number of these?)

We will apply for to present special sessions and workshops on the use of Sage in the classroom at the Joint Mathematics Meetings and MathFest, where we will be able to share the results of Project Sage with the undergraduate teaching community. We will also publish articles about the impact of using our materials on the teaching and learning of undergraduate mathematics. Drs. Beezer and Grout are part of a team delivering a workshop on how to use Sage in the classroom under the MAA's Professional Enhancement Program (PREP). We will apply to lead future PREP workshops that will incorporate the results of Project Sage.

7. Evaluation

The evaluation plan will measure the impact of Sage on undergraduate students, faculty, and the curriculum. More specifically, the plan will include methodologies for measuring

(1) the effectiveness of the project in developing Sage integrated curriculum materials such as Sage integrated with open source textbooks,
(2) the effectiveness of the use of Sage integrated materials within the secondary and undergraduate classroom.

The evaluation plan will also include a mechanism for tracking the implementation of Sage materials into the undergraduate and high school classroom by working with selected test sites. We will create an online tracking system to ensure that timely and accurate qualitative and quantitative data are collected for all test sites, including demographic data and statistics as well as service obligations.

Experience with previous, similar projects guides the evaluation plan for this effort. [Note: Add in the qualifications and experience of the evaluators.]

7.1. Formative evaluation.

The formative component of the evaluation will focus on monitoring the quality of project activities, enabling the project to make mid-course corrections where needed and will address (1) the effectiveness of the project in developing Sage integrated materials. The summative component will focus on the impact of the project on the students and teachers at the test sites and will address (2) the effectiveness of the Sage-integrated materials in the undergraduate and high school classroom. We plan to gather information on the contributions of the project to the fields knowledge about future development of CAS integrated curricular materials.

The formative evaluation will be guided by the following questions:

(1) To what extent are Sage integrated materials and the support for these materials designed to the needs of the test sites.
(2) To what extent are the Sage integrated materials transportable/scalable to other undergraduate and high school classrooms?

The formative aspects of the evaluation will collect information on key aspects of the project to assess and provide feedback on the extent to which the project is designed and implemented to address important needs effectively. The key aspects of the project on which the evaluation will focus are: the need for, and resources to support, Sage enhanced materials in the classroom; and creation of materials that can be used more broadly in high school and university mathematics classrooms. Classroom needs and resource commitments will be documented through teacher surveys and interviews. The basis for selection of participating teachers and classrooms will be based on interest in the project and support from the teacher's institution.

7.2. **Summative evaluation.** The summative evaluation will be guided by the following questions:

(1) What is the impact of Sage-enhanced materials on teachers content knowledge, pedagogical content knowledge and classroom instructional practice?
(2) What is the impact of Sage-enhanced materials on in their own classrooms, campuses/districts on their students mathematics learning and participation in advanced-level mathematics courses?
(3) What is the ease of implementing Sage-enhaced integrated materials on a wide scale basis?

To measure these changes Dr. Judson will collaborate with the project director and other members of the project and will consult with Marja-Liisa Hassi and Sandra Laursen of Ethnography & Evaluation Research (E&ER) at the Center to Advance Research and Teaching in the Social Sciences (CARTSS) at the University of Colorado in the selection of appropriate instruments to identify measures of for specific areas of the project. Drs. Hassi and Laursen have extensive experience evaluating other large mathematics and science education projects (NSF Awards # 0920126, # 0723600 # 0450088). [Note: I am not sure that these are the appropriate projects that we should mention, but I am assuming that Marja-Liisa will rewrite this part of the proposal.] To insure alignment to project goals, measures for the classroom will not be selected until the Sage-enhanced materials have been developed. The measures will be administered to participants at appropriate intervals throughout the project period.

Drs. Hassi and Judson and the project PIs will conduct site visits to partner institutions to observe and document the use of Sage-enhanced materials as the project progresses. Additional data will be collected through periodic surveys of all participants, and interviews of a sample. All surveys and interviews will be prepared and conducted by Drs. Hassi and Judson.

Drs. Hassi and Judson will collaborate with the appropriate institutions to collect demographic data on schools, teachers, and students within participating schools. They will analyze all data using appropriate statistical measures and will prepare one annual report per year documenting all evaluation activities and results and will provide formative evaluation feedback to the project PIs in presentations and correspondence timed to inform design decisions and mid-course corrections.

8. Budget Overview

(Save this for later)

- Total Salaries
- Total Stipends to test sites (travel money here?)
- Stipends for curricular materials
- Total travel, equipment
- Total package for infrastructure

9. Future Directions

The limited scope of this proposal suggests several broader new directions. In particular, the project team was intrigued by the possibilies for using Sage to generate and check homework through integration with a system like WebWork, but decided such an initiative was too ambitious and disjoint to be included in this proposal.

- Create new open textbooks with Sage enhancements designed in from the start.

- Create Sage-enhanced open textbooks for every standard course in the undergraduate mathematics curriculum.
- Integrate Sage with an open-source homework delivery system such as WebWork.
- Integrate collaboration features of the Sage notebook with a course management system such as Moodle.
- More????

End of OPEN-VERSION proposal

10. References

The list of references here.