Student: Ben Karel Field: Computer Science Advisor: David Saunders

Modernizing the LinBox Software Ecosystem

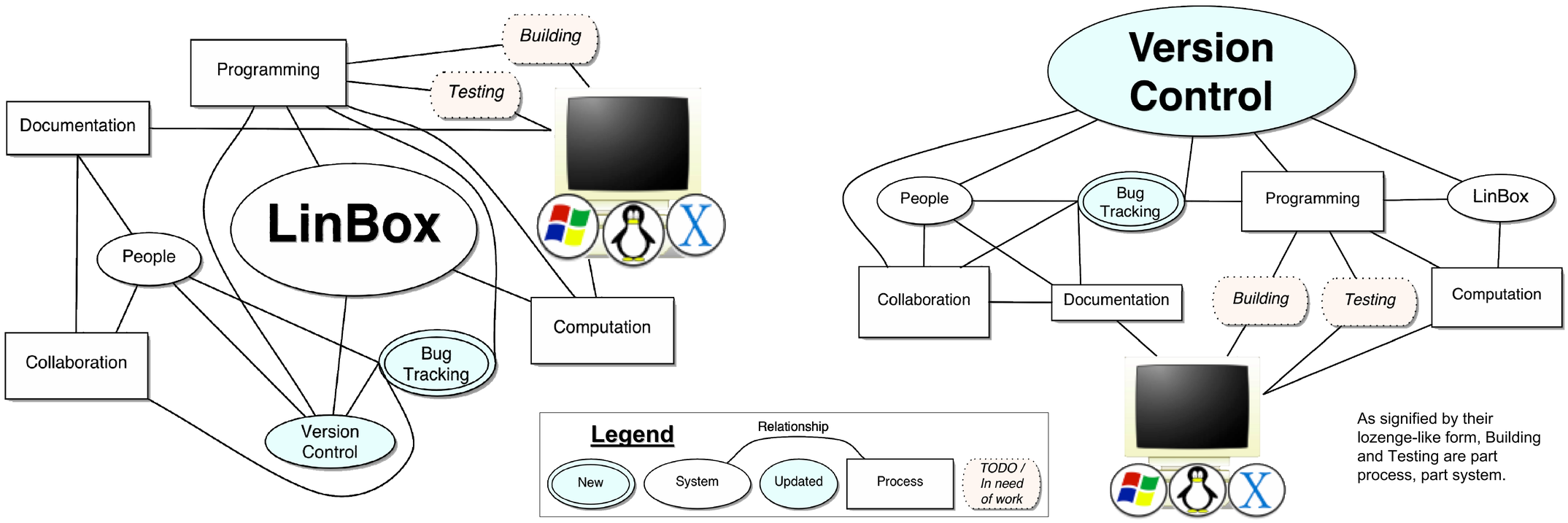
John Donne once mused that no man is an island. So, too, does every software project stand upon the shoulders of giants. There are many thousands of man-hours of intellectual labor represented in the ancillary software tools that support, manage, and organize software projects, especially ones as large as LinBox.

My research this past summer involved working with Dr. Saunders to find, identify, and oversee appropriate upgrades to the support infrastructure for LinBox. In addition to relatively minor tweaks to the project, my primary contributions were adding Trac, and moving the LinBox software repository from CVS to Subversion. These upgrades should enable current and future contributors to the project to work more efficiently and thus further the state of the art in linear algebra software.

Here we see two different views of the same software "ecosystem"*:

On the left, a graph set up with the LinBox project as the center, primary node in the system. The highly-connected graph shows the complex connections and relationships between the various components in the LinBox software ecosystem.

On the right, a different layout of the same graph better mirrors the inherent organization in the tools and systems connected with the LinBox project. Here, the highly-connected Subversion system for version control takes front-and-center, reflecting its central role in the day-to-day life of the LinBox project.



* What, exactly, constitutes a software ecosystem? Roughly speaking, as evidenced from the legend above, a software ecosystem comprises the systems and processes used in developing a piece of software such as LinBox.

About LinBox

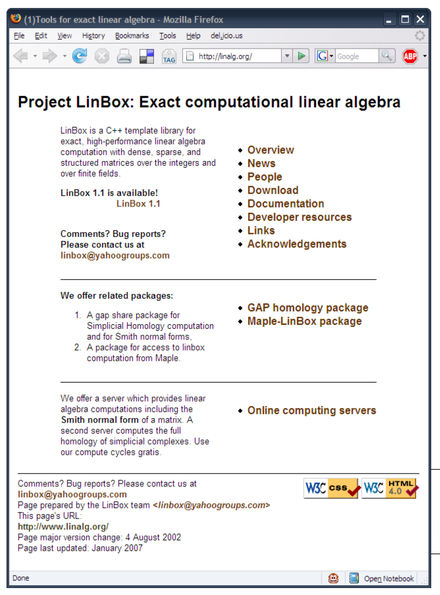
LinBox is a software library that provides programmers with high-performance computational tools in the field of linear algebra. It is implemented though a flexible, extensible, generic system based on C++ templates. The templates enable containers called "black boxes" to separate the internal representation of specialized matrix types from the logic used to manipulate them.
Using these black boxes, LinBox can perform many calculations over many types of entities:

Linear Operators:
- Determinants
- Trace
- Rank
- MinPoly
- CharPoly
- Eigenvalues
- Eigenvectors

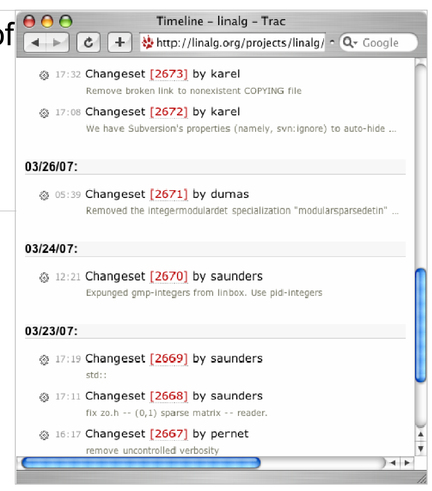
Canonical Matrix Forms:
- Smith Normal Form
- Frobenius

Domains:
- Finite fields
- Rationals
- Integers

Other Features:
- Efficient handling of sparse and structured matrices
- Ability to solve systems of hundreds of thousands of equations

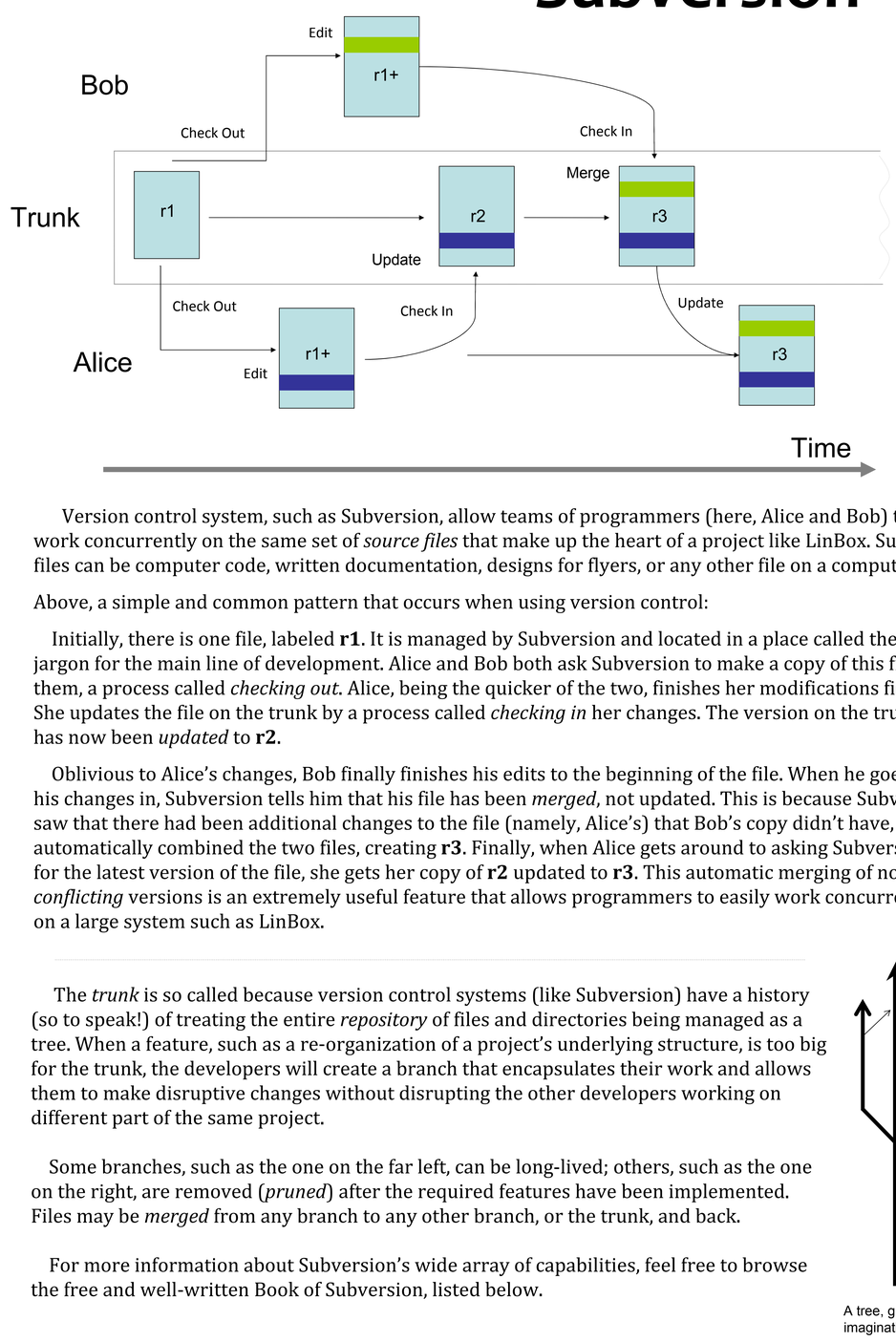


www.linalg.org, home of LinBox!



A screenshot of LinBox's Trac install, which keeps track of code changes.

Version Control with Subversion



Version control system, such as Subversion, allow teams of programmers (here, Alice and Bob) to work concurrently on the same set of *source files* that make up the heart of a project like LinBox. Such files can be computer code, written documentation, designs for flyers, or any other file on a computer.

Above, a simple and common pattern that occurs when using version control:

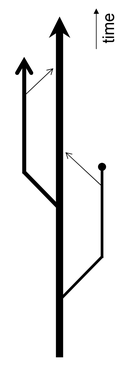
Initially, there is one file, labeled **r1**. It is managed by Subversion and located in a place called the *trunk*, jargon for the main line of development. Alice and Bob both ask Subversion to make a copy of this file for them, a process called *checking out*. Alice, being the quicker of the two, finishes her modifications first. She updates the file on the trunk by a process called *checking in* her changes. The version on the trunk has now been *updated* to **r2**.

Oblivious to Alice's changes, Bob finally finishes his edits to the beginning of the file. When he goes to his changes in, Subversion tells him that his file has been *merged*, not updated. This is because Subversion saw that there had been additional changes to the file (namely, Alice's) that Bob's copy didn't have, and automatically combined the two files, creating **r3**. Finally, when Alice gets around to asking Subversion for the latest version of the file, she gets her copy of **r2** updated to **r3**. This automatic merging of non-*conflicting* versions is an extremely useful feature that allows programmers to easily work concurrently on a large system such as LinBox.

The *trunk* is so called because version control systems (like Subversion) have a history (so to speak!) of treating the entire *repository* of files and directories being managed as a tree. When a feature, such as a re-organization of a project's underlying structure, is too big for the trunk, the developers will create a branch that encapsulates their work and allows them to make disruptive changes without disrupting the other developers working on different part of the same project.

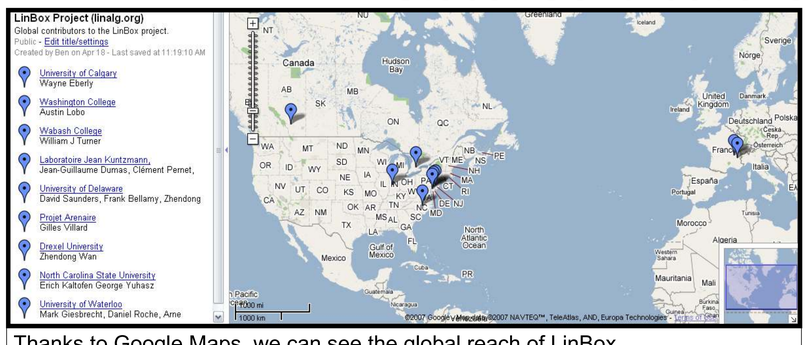
Some branches, such as the one on the far left, can be long-lived; others, such as the one on the right, are removed (*pruned*) after the required features have been implemented. Files may be *merged* from any branch to any other branch, or the trunk, and back.

For more information about Subversion's wide array of capabilities, feel free to browse the free and well-written Book of Subversion, listed below.



A tree, given some imagination?

Global Contributions to Project LinBox



Thanks to Google Maps, we can see the global reach of LinBox.

Each blue placemark signifies at least one significant contributor to LinBox.

Back row, from the left: David Saunders, Qing Xiang, William J. Turner, Carl Devore, Zhendong Wan, Ahmet Duran.
Front row, left to right: Gilles Villard, Pascal Giorgi, Dmitri Morozov, Erich Kaltofen, Mark Giesbrecht, Arne Storjohann

Useful URLs

http://subversion.tigris.org/	Home of Subversion, with links to related utilities
http://cvs2svn.tigris.org/	Very slick tool for converting CVS repositories to Subversion. Maintains history and everything!
http://svnbook.red-bean.com/	The Book of Subversion. Live it, love it, learn it.
http://trac.edgewall.org/	Home of Trac, a simple but functional issue tracker, with svn integration
http://linalg.org/	Home of LinBox and related software.

All the software mentioned on this poster is freely available under an Open Source certified license near you!