Overview

• Introduction (community statistics)
• Past: Where have we been
• Present: Where are we
• Future: Where are we going?
There has been marked growth in the community of users around scientific computing in Python in the past year.
Mailing List Activity

Monthly Mailing List Traffic (KBytes) August 2001 - August 2008

scipy-dev  scipy-user  numeric-discussion  1 year moving avg.
The SciPy Community is Global
Visits to scipy.org since Sept. 2008

<table>
<thead>
<tr>
<th>Rank</th>
<th>Country</th>
<th>Visits</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>United States</td>
<td>252,208</td>
<td>38.12%</td>
</tr>
<tr>
<td>2</td>
<td>Germany</td>
<td>59,515</td>
<td>9.00%</td>
</tr>
<tr>
<td>3</td>
<td>United Kingdom</td>
<td>41,063</td>
<td>6.21%</td>
</tr>
<tr>
<td>4</td>
<td>France</td>
<td>40,354</td>
<td>6.10%</td>
</tr>
<tr>
<td>5</td>
<td>Canada</td>
<td>24,688</td>
<td>3.73%</td>
</tr>
<tr>
<td>6</td>
<td>Japan</td>
<td>19,378</td>
<td>2.93%</td>
</tr>
<tr>
<td>7</td>
<td>Italy</td>
<td>15,151</td>
<td>2.29%</td>
</tr>
<tr>
<td>8</td>
<td>Australia</td>
<td>13,591</td>
<td>2.05%</td>
</tr>
<tr>
<td>9</td>
<td>China</td>
<td>13,452</td>
<td>2.03%</td>
</tr>
<tr>
<td>10</td>
<td>Netherlands</td>
<td>11,799</td>
<td>1.78%</td>
</tr>
</tbody>
</table>
Student Sponsorship

<table>
<thead>
<tr>
<th>Student Name</th>
<th>Country</th>
</tr>
</thead>
<tbody>
<tr>
<td>David Cournapeau</td>
<td>Japan</td>
</tr>
<tr>
<td>Stefan van der Walt</td>
<td>South Africa</td>
</tr>
<tr>
<td>Rahul Garg</td>
<td>Alberta</td>
</tr>
<tr>
<td>Seshadri Tirunillai</td>
<td>USC</td>
</tr>
<tr>
<td>Amir Khosrowshahi</td>
<td>Berkeley</td>
</tr>
<tr>
<td>Dav Clark</td>
<td>Berkeley</td>
</tr>
<tr>
<td>Gabriel Gellner</td>
<td>Toronto</td>
</tr>
<tr>
<td>Min Ragan-Kelley</td>
<td>Berkeley</td>
</tr>
<tr>
<td>Ryan May</td>
<td>OU</td>
</tr>
<tr>
<td>David Warde-Farley</td>
<td>Toronto</td>
</tr>
</tbody>
</table>

Many thanks to the sponsoring organizations!
Short History

- 1994 — Python Matrix object (Jim Fulton)
- 1995 — Numeric born (Jim Hugunin, Konrad Hinsen, Paul Dubois, David Ascher, Jim Fulton)
- 2000 — Numeric moves to sourceforge (Project registered as numpy)
- 2001 — SciPy born (Pearu Peterson, Travis Oliphant, Eric Jones)
- 2001 — IPython born (Fernando Perez)
- 2002 — SciPy '02 - Python for Scientific Computing Workshop
- 2003 — matplotlib born (John Hunter)
- 2003 — Numarray (Perry Greenfield, J. Todd Miller, Rick White, Paul Barrett)
- 2006 — NumPy 1.0 Released
• Now for something completely different
NumPy Book Announcement

Trelgol Publishing

Setting information free

0110101

Products

<table>
<thead>
<tr>
<th>Name</th>
<th>Author</th>
<th>Unit Price</th>
<th>Buy Now</th>
<th>Total Time</th>
<th>May Sell Rights</th>
<th>World Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>Generic layout file</td>
<td>Travis E. Oliphant</td>
<td>$5.00</td>
<td>PayPal Buy Now</td>
<td>3 yrs Aug 21, 2008</td>
<td>N</td>
<td>$500</td>
</tr>
</tbody>
</table>

1 You may still donate this amount, but the book is now in the public domain.
2 The final actual sold "world price" was about $105,000. Much of this money went to fund students.

World Price Briefly Explained

This represents the total price the author requires before freely releasing the content. No one is required to individually pay this amount.
Present: Where we are

- NumPy/SciPy packages
- Community involvement
- Larger ecosystem
2007-2008 NumPy/SciPy Releases

- SciPy 0.6.0 (September 2007)
- NumPy 1.0.4 (November 2007)
- NumPy 1.1.0 (May 2008)
- NumPy 1.1.1 (August 2008)
Major Changes

- New MaskedArrays
- IO code overhaul
- Improved test coverage
- NumScons
- Focus on coding style
Release Management

- Better packaging
- Release numbering
- Buildbot
Community Involvement

- Blogs (http://planet.scipy.org)
- IRC (#scipy on freenode)
- Mailing lists
- Sprints
- Conferences
• Average daily message count:
  – numeric-discussion, scipy-user, scipy-dev: 44 messages/day
  – Adding in ipython and matplotlib yields another 15 per day
  – If you monitor enthought-dev you’ll be reading another 31 per day
• Total: 90 messages per day
• Yes, it’s hard to keep up.
Sprints

- August 2007 — SciPy 2007 sprint
- December 2007 — SciPy sprint at UCB
- February 2008 — Sage days 8 at Enthought
- March 2008 — NIPY/IPython sprint in Paris
- April 2008 — SciPy sprint at UCB
- July 2008 — MayaVi sprint at Enthought
Conferences

- PyCon 2008
- 2008 SIAM meeting
- EuroSciPy 2008
- SciPy 2008
The Larger Ecosystem

- Major releases
- Distribution
- New Tools
Major Releases

- matplotlib 0.98
- ETS 3
- MayaVi 2
Distribution

- Python Package Index
- Python(x,y)
- EPD
- Sage
Future: Where are we going?

- NumPy/SciPy packages
- The Larger Ecosystem
- Getting involved
NumPy/SciPy releases

• Upcoming releases
• Release management
NumPy 1.2

- Sphinx-based documentation!!
- Nose-based testing framework (GSoC project)
- Changes to histogram and median
- Arrayterators
NumPy 1.2 open issues

- ABI compatibility
- Generalized UFuncs
- C-API cleanups
- Release date
SciPy 0.7

- Sparse matrices improvements
- New functionality: Radial basis functions, hierarchical clustering, constants
- Nose-based testing framework
- Sandbox removed
Release Management

- Time-based releases
- Code Review
- Enhancement Proposals
- Distributed Version Control
**Feature-based**
A release cycle under this model is driven by deciding what features will be in the next release. Once all the features are complete, the code is stabilized and finally a release is made. Obviously this makes it relatively easy to predict what features will be in the next release, but extremely difficult to determine when the release will occur.

**Time-based**
A release cycle under this model is driven by deciding when the next release will be. This, of course, makes predicting when the release will be out extremely easy, but makes it difficult to know exactly what features will be included in the release.
Time-based Release Issues

- Branching
- Reviewing
- Testing
- Reverting
- Postponing
- Releasing
In order to be able to release on schedule requires that the mainline of development (the trunk) is extremely stable. This requires that a significant amount of work be conducted on branches.
Another important way to improve the quality of project and keep the trunk in shape is to require peer code review and consensus among the core developers on which branches are ready to be merged.
A full test suite is also essential for being able to regularly release code.
Time-based Releases: Reverting

Sticking to release schedule requires occasionally reverting commits.
Time-based Releases: Postponing

It also requires postponing branch merges until the branch is ready for release.
Since time-based release management relies on a regular releases, the cost of making a release needs to be minimized. In particular, we need to make it much, much easier to create the packages, post the binaries, create the release notes, and send out the announcements.
Release Management Practices

- Code review
- Enhancement proposals
- Distributed version control
Issue 2946: Update NumPy API format to support updates that don’t break binary compatibility

**Description**

The current NumPy API number, stored as NPY_VERSION in the header files, needs to be incremented every time the NumPy C-API changes. The counter tells developers with exactly which revision of the API they are dealing. NumPy does some checking to make sure that it does not run against an old version of the API. Currently, we have no way of distinguishing between changes that break binary compatibility and those that don’t.

The proposed fix breaks the version number up into two counters — one that gets increased when binary compatibility is broken, and another when the API is changed without breaking compatibility.

Backward compatibility with packages such as Matplotlib is maintained by renaming NPY_VERSION to NPY_BINARY_VERSION.

**Patch Set 1**

Total comments: 4

<table>
<thead>
<tr>
<th>Raw unified diffs</th>
<th>Stats</th>
<th>Side-by-side diffs with inline comments</th>
<th>Delta from patch set</th>
<th>Patch</th>
</tr>
</thead>
<tbody>
<tr>
<td>numpy/core/code_generators/generate_numpy_api.py</td>
<td>1 chunk</td>
<td>20 lines</td>
<td>0 comments</td>
<td>2</td>
</tr>
<tr>
<td>numpy/core/include/numpy/ndarrayobject.h</td>
<td>1 chunk</td>
<td>31 lines</td>
<td>4 comments</td>
<td>2</td>
</tr>
<tr>
<td>numpy/core/src/multiarray/module.c</td>
<td>1 chunk</td>
<td>13 lines</td>
<td>0 comments</td>
<td>2</td>
</tr>
</tbody>
</table>

**Patch Set 2: Updated with Andrew’s suggestions.**

Created: 4 days, 16 hours ago

<table>
<thead>
<tr>
<th>Raw unified diffs</th>
<th>Stats</th>
<th>Side-by-side diffs with inline comments</th>
<th>Delta from patch set</th>
<th>Patch</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Code Review: Rietveld Tool, cont.

*/ There are several places in the code where an array of dimensions is allocated statically. This is the size of that static allocation.
*/ The array creation itself could have arbitrary dimensions but all the places where static allocation is used would need to be changed to dynamic (including inside of several structures)
*/
#define NPY_MAXNARGS 32
#define NPY_MAXDIMS 32

/* Used for Converter Functions "04" code in ParseTuple */
#define NPY_FAIL 0
#define NPY_NOSUCCEED 1

/* Helpful to distinguish what is installed */
#define NPY_VERSION 0x01000000

*/ There are several places in the code where an array of dimensions is allocated statically. This is the size of that static allocation.
*/ The array creation itself could have arbitrary dimensions but all the places where static allocation is used would need to be changed to dynamic (including inside of several structures)
*/
#define NPY_MAXNARGS 32
#define NPY_MAXDIMS 32

/* Used for Converter Functions "04" code in ParseTuple */
#define NPY_FAIL 0
#define NPY_NOSUCCEED 1

*/

Andrew Straw 2008/08/16 07:20:56 there's no need to change the actual value of NPY_
Stefan 2008/08/16 07:33:19 On 2008/08/16 07:20:56, Andrew Straw wrote: > ther

Andrew Straw 2008/08/16 07:20:56 Perhaps move NPY_VERSION below the NPY_BINARY_VERSION
Stefan 2008/08/16 07:33:19 On 2008/08/16 07:20:56, Andrew Straw wrote: > Perh

*/ Whenever (NPY_BINARY_VERSION & NPY_BINARY_COMPATIBLE) changes, external libraries that depend on NumPy need to be recompiled.
In other words, NPY_BINARY_VERSION consists of two parts:
0x00000000
where M gets increased whenever binary compatibility is broken, and K gets increased whenever the API is changed but binary compatibility is maintained.
*/
#define NPY_BINARY_VERSION 0x00010001
#define NPY_BINARY_COMPATIBLE 0xFFFF0000

/* Some platforms don't define bool, long long, or long double.
Handle that here.
*/

*/

*/ Some platforms don't define bool, long long, or long double.
Handle that here.
*/
Enhancement Proposals

- A Simple File Format for NumPy Arrays
- Implementing date/time types in NumPy
- Matrix Indexing
- Runtime Optimization
- Solvers Proposal
Launchpad

What is Launchpad?
Launchpad is a hosting service for open source projects that's big on collaboration. Take a tour... (new!)

Start here:
- Projects
- Distributions
- People and Teams

What's new?
- **Bugzilla and Trac plugins now in beta**
  Do you run a Bugzilla or Trac instance? Help beta test our new plugins and share bug comments with Launchpad.
- **Karma for commits**
  You'll now get karma for each commit you make to a branch that's registered in Launchpad.
- **New Launchpad user guide**
  If you're new to Launchpad, our new user guide will take you through each application step by step.
- **Help test Launchpad's new API!**
  Our new internet-services API is now in beta test.

Updated 2008-08-20. More...

Help improve Launchpad

© 2004-2008 Canonical Ltd. | Terms of use
Active code reviews for IPython

25 branches owned by 7 people and 2 teams, 159 commits by 18 people in the last month

3 active reviews, 0 approved merges

You can browse the source code for the development focus branch or get a copy of the branch using the command:

    bzr branch lp:python

There are download files available for IPython.

<table>
<thead>
<tr>
<th>Branch Merge Proposal</th>
<th>Requested By</th>
<th>Date Review Requested</th>
<th>Vote Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>lp:-laurent-dufrechouipthon/trunk-dev ⇒ lp:python</td>
<td>Fernando Perez</td>
<td></td>
<td>no votes (no comments)</td>
</tr>
<tr>
<td>lp:-fdo.perezipthon/trunk-dev ⇒ lp:python</td>
<td>Fernando Perez</td>
<td></td>
<td>no votes (no comments)</td>
</tr>
<tr>
<td>lp:-python-dev/python/python-daemon ⇒ lp:python</td>
<td>Fernando Perez</td>
<td></td>
<td>no votes (no comments)</td>
</tr>
</tbody>
</table>
Launchpad

~vcs-imports/numpy/trunk branch in Launchpad

Branch imported NumPy
Project: NumPy
Status: New
Get this branch by branch 1p:numpy
Branch format: Branch format B
Repository format: Packs containing knits without subtree support

Import details

Import Status: Reviewed
This branch is an import of the Subversion branch from http://svn.scipy.org/svn/numpy/trunk.
The next import is scheduled to run in 44 minutes.
Last successful import was 5 hours ago.

- Import started 5 hours ago on galapagos and finished 5 hours ago taking two minutes — see the log
- Import started 13 hours ago on neumayer and finished 13 hours ago taking two minutes — see the log
- Import started 19 hours ago on galapagos and finished 19 hours ago taking a minute
- Import started on 2008-08-20 on galapagos and finished on 2008-08-20 taking 50 seconds
- Import started on 2008-08-19 on neumayer and finished on 2008-08-19 taking 50 seconds
- Import started on 2008-08-19 on neumayer and finished on 2008-08-19 taking a minute
- Import started on 2008-08-19 on neumayer and finished on 2008-08-19 taking 50 seconds
- Import started on 2008-08-19 on galapagos and finished on 2008-08-19 taking 40 seconds
- Import started on 2008-08-18 on neumayer and finished on 2008-08-18 taking a minute
- Import started on 2008-08-18 on neumayer and finished on 2008-08-18 taking 50 seconds

No branches proposed for merging into this one.
Not proposed for merging
The Larger Ecosystem

- IPython
- Python
• IPython1 in trunk
• Graphical frontends
• More...
Python 2.6 and 3.0

Gmail – RELEASED Python 2.6b3 and 3.0b3

1 message

Barry Warsaw <barry@python.org>
Reply-To: python-list@python.org
To: python-announce@python.org, python-dev@python.org, python-3000@python.org, python-list@python.org

-----BEGIN PGP SIGNED MESSAGE-----
Hash: SHA1

On behalf of the Python development team and the Python community, I am happy to announce the third and last planned beta releases of Python 2.6 and Python 3.0.

Please note that these are beta releases, and as such are not suitable for production environments. We continue to strive for a high degree of quality, and these releases are intended to freeze the feature set for Python 2.6 and 3.0.

As these are the last planned beta releases, we strongly urge you to download these releases and test them against your code. Once we reach release candidates (currently planned for 03-Sep-2008), only highly critical bugs will be fixed before the final release.

If you find things broken or incorrect, please submit bug reports at

http://bugs.python.org

For more information and downloadable distributions, see the Python 2.6 website:

http://www.python.org/download/releases/2.6/
and the Python 3.0 website:

http://www.python.org/download/releases/3.0/

-----END PGP SIGNED MESSAGE-----
NumPy/SciPy and Python 2.6/3.0

- Improve test coverage
- Port to Python 2.6
- Improve test coverage
- Run 2to3 tool
- Fix issues
- ISSUE: No 2to3 tool for the C-API!
Getting Involved

- Documentation
- Bug-fixes
- Testing
- Code (review, cleanup, contributions)
- Mailing list participation
- Start a local SciPy group
- Code sprints
- Documentation/Bug-fix Days
- Web design
Thank You