pandas?

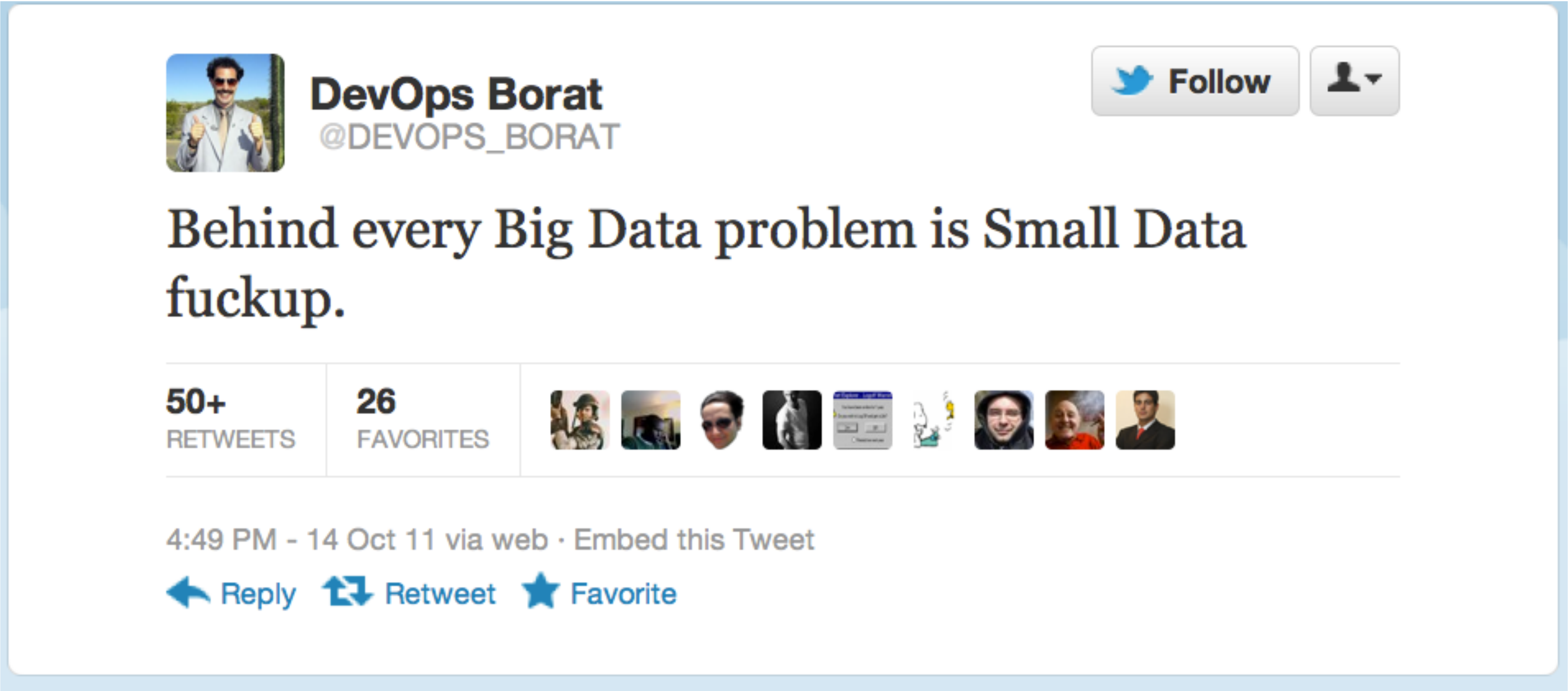
- http://pandas.pydata.org
- Rich relational data tool built on top of NumPy
  - Like R's `data.frame` on steroids
  - Excellent performance
  - Easy-to-use, highly consistent API
- A foundation for data analysis in Python

pandas

- In heavy production use in the financial industry, among others
- Generally much better performance than other open source alternatives (e.g. R)
- Hope: basis for the “next generation” statistical computing and analysis environment

Simplifying data wrangling

- Data munging / preparation / cleaning / integration is slow, error prone, and time consuming
- Everyone already <3's Python for data wrangling: pandas takes it to the next level



DevOps Borat
@DEVOPS_BORAT

Behind every Big Data problem is Small Data fuckup.

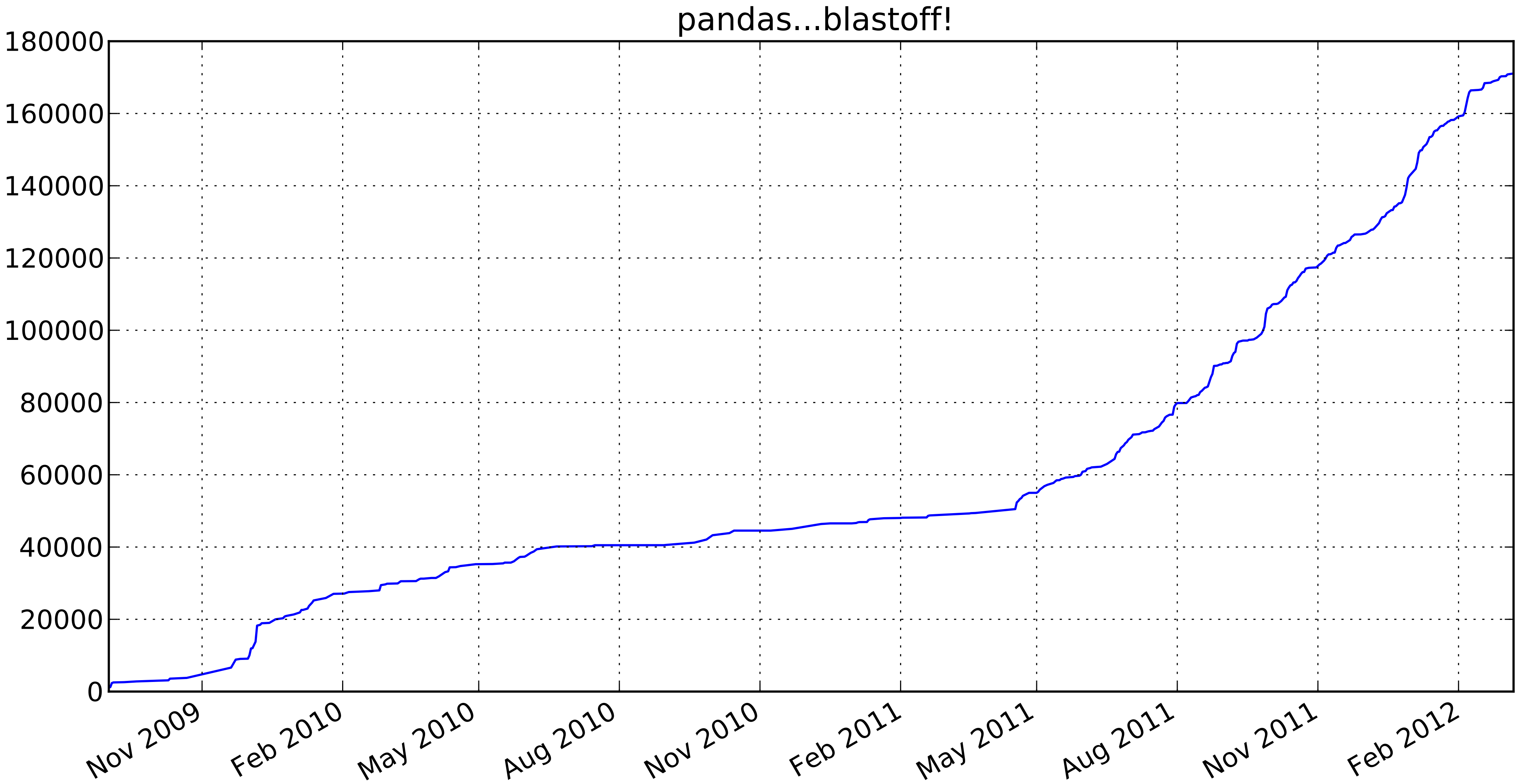
50+ RETWEETS

26 FAVORITES

4:49 PM - 14 Oct 11 via web · Embed this Tweet

Explosive pandas growth

- 10 significant releases since 9/2011
- Hugely increased user base



Battle tested

- > 98% line coverage as measured by coverage.py
- v0.3.0 (2/19/2011): 533 test functions

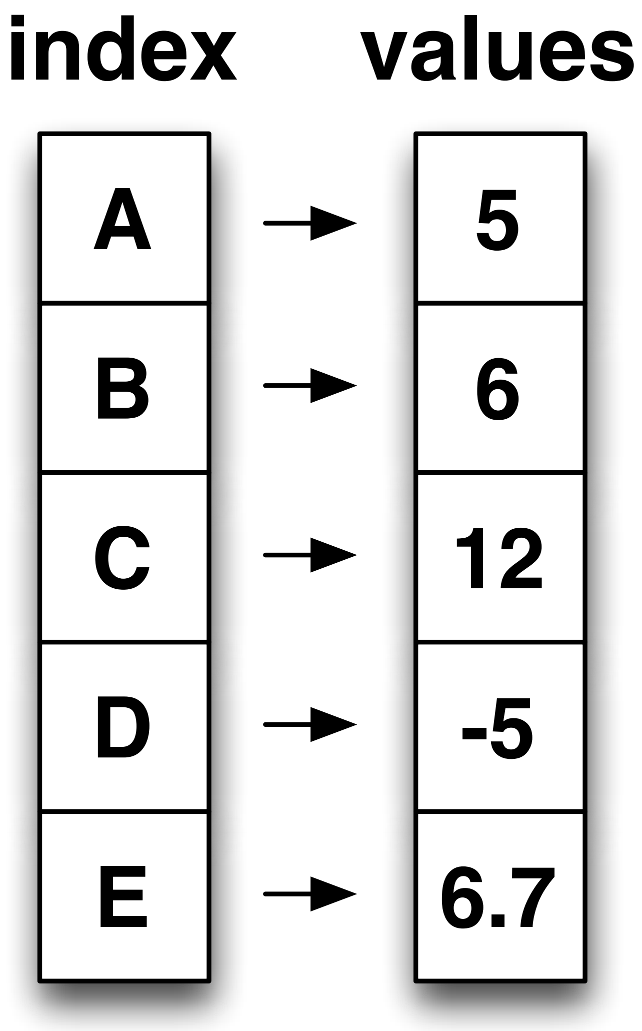
Battle tested

- > 98% line coverage as measured by coverage.py
- v0.3.0 (2/19/2011): 533 test functions
- v0.7.3dev (3/27/2012): >1500 test functions

IPython

- Simply put: one of the hottest Python projects out there
- Tab completion, introspection, interactive debugger, command history
- Designed to enhance your productivity in every way. I can’t live without it
- IPython HTML notebook is #winning

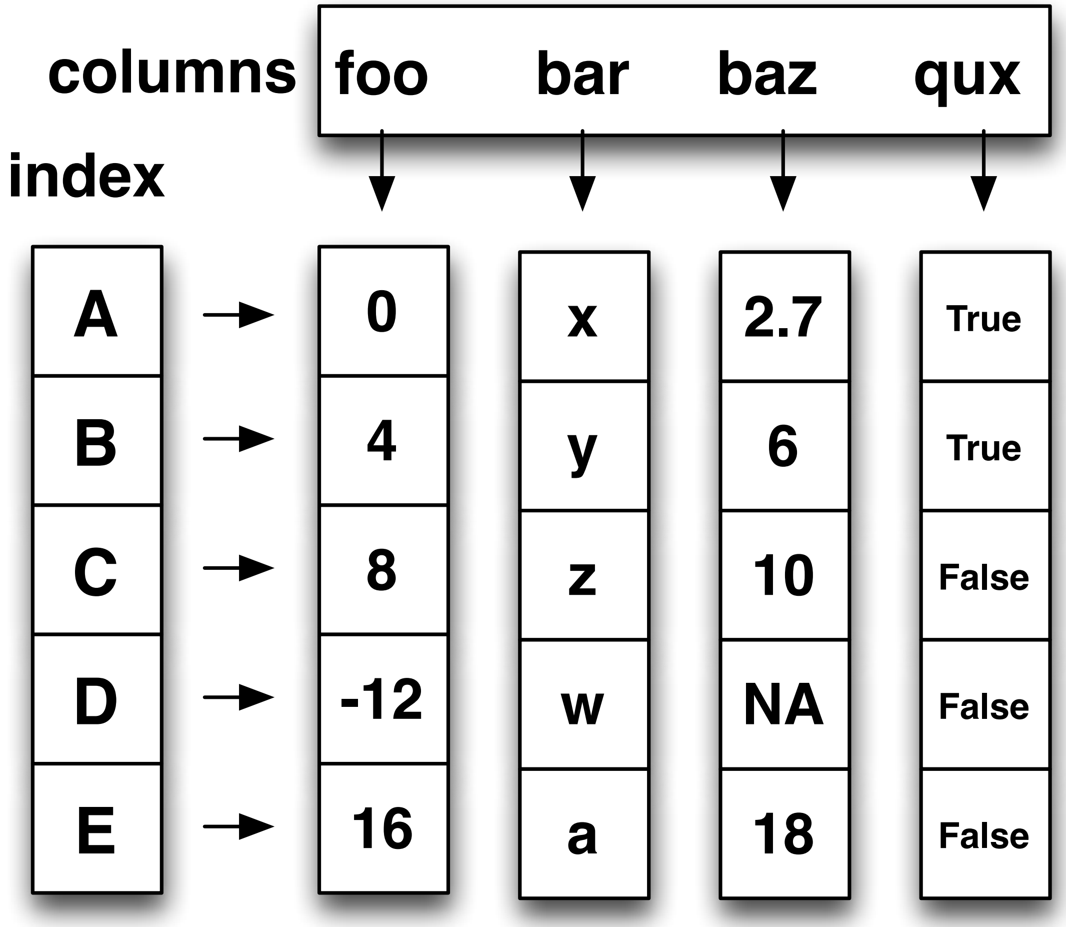
Series



index		values
A	→	5
B	→	6
C	→	12
D	→	-5
E	→	6.7

- Subclass of `numpy.ndarray`
- Data: any type
- Index labels need not be ordered
- Duplicates are possible (but result in reduced functionality)

DataFrame



columns

index

index	foo	bar	baz	qux
A	0	x	2.7	True
B	4	y	6	True
C	8	z	10	False
D	-12	w	NA	False
E	16	a	18	False

- NumPy array-like
- Each column can have a different type
- Row and column index
- Size mutable: insert and delete columns

DataFrame

```
In [10]: tips[:10]
Out[10]:
    total_bill  tip  sex    smoker day time    size
1   16.99       1.01 Female No     Sun Dinner  2
2   10.34       1.66 Male   No     Sun Dinner  3
3   21.01       3.50 Male   No     Sun Dinner  3
4   23.68       3.31 Male   No     Sun Dinner  2
5   24.59       3.61 Female No     Sun Dinner  4
6   25.29       4.71 Male   No     Sun Dinner  4
7   8.770       2.00 Male   No     Sun Dinner  2
8   26.88       3.12 Male   No     Sun Dinner  4
9   15.04       1.96 Male   No     Sun Dinner  2
10  14.78       3.23 Male   No     Sun Dinner  2
```

DataFrame

- Axis indexing enable rich data alignment, joins / merges, reshaping, selection, etc.

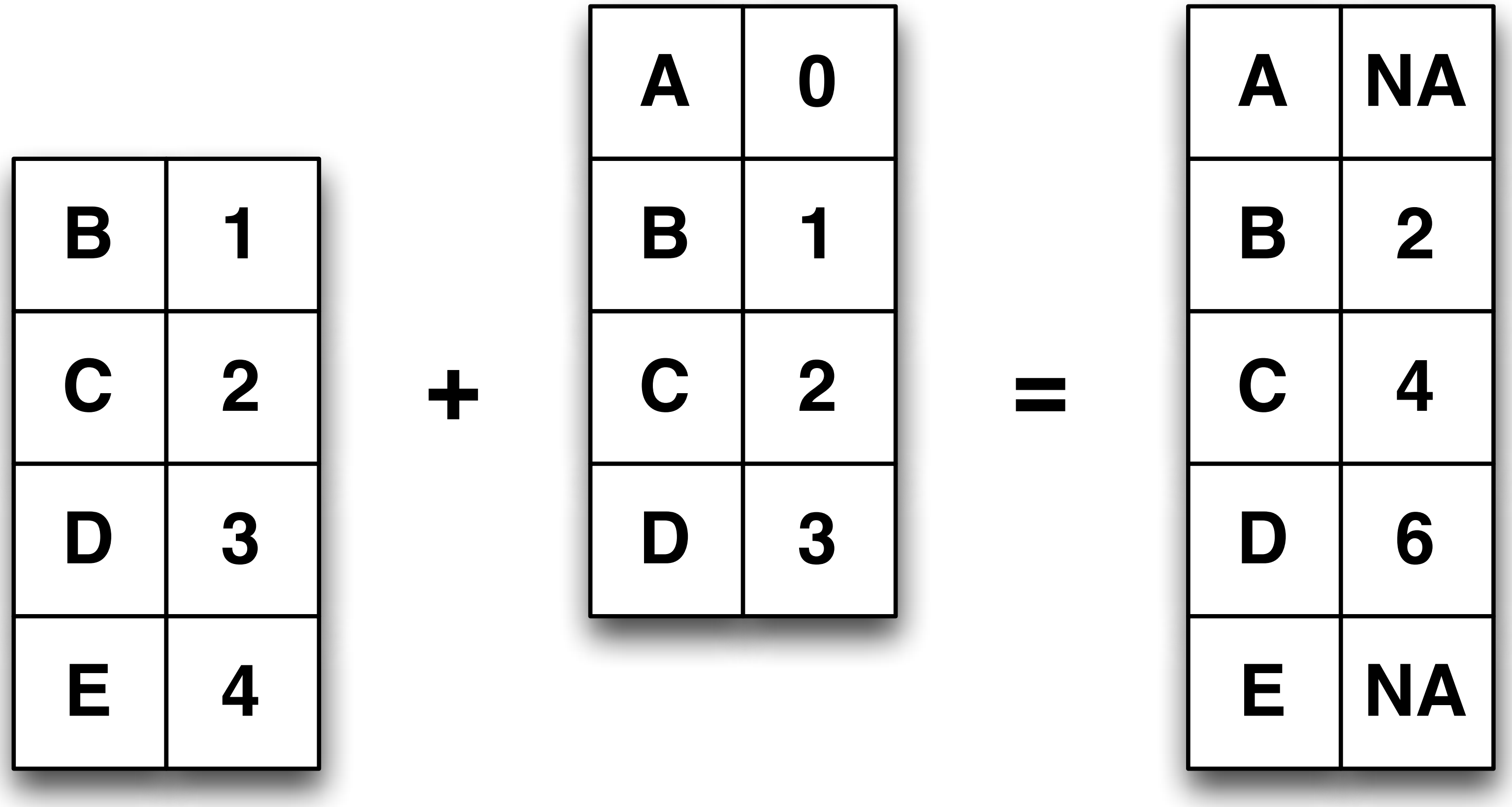
```
day            Fri    Sat    Sun    Thur
sex    smoker
Female No      3.125  2.725  3.329  2.460
       Yes     2.683  2.869  3.500  2.990
Male   No      2.500  3.257  3.115  2.942
       Yes     2.741  2.879  3.521  3.058
```

Axis indexing, the special pandas-flavored sauce

- Enables "alignment-free" programming
- Prevents major source of data munging frustration and errors
- Fast data selection
- Powerful way of describing reshape / join / merge / pivot-table operations

Data alignment

- Binary operations are joins!



B	1
C	2
D	3
E	4

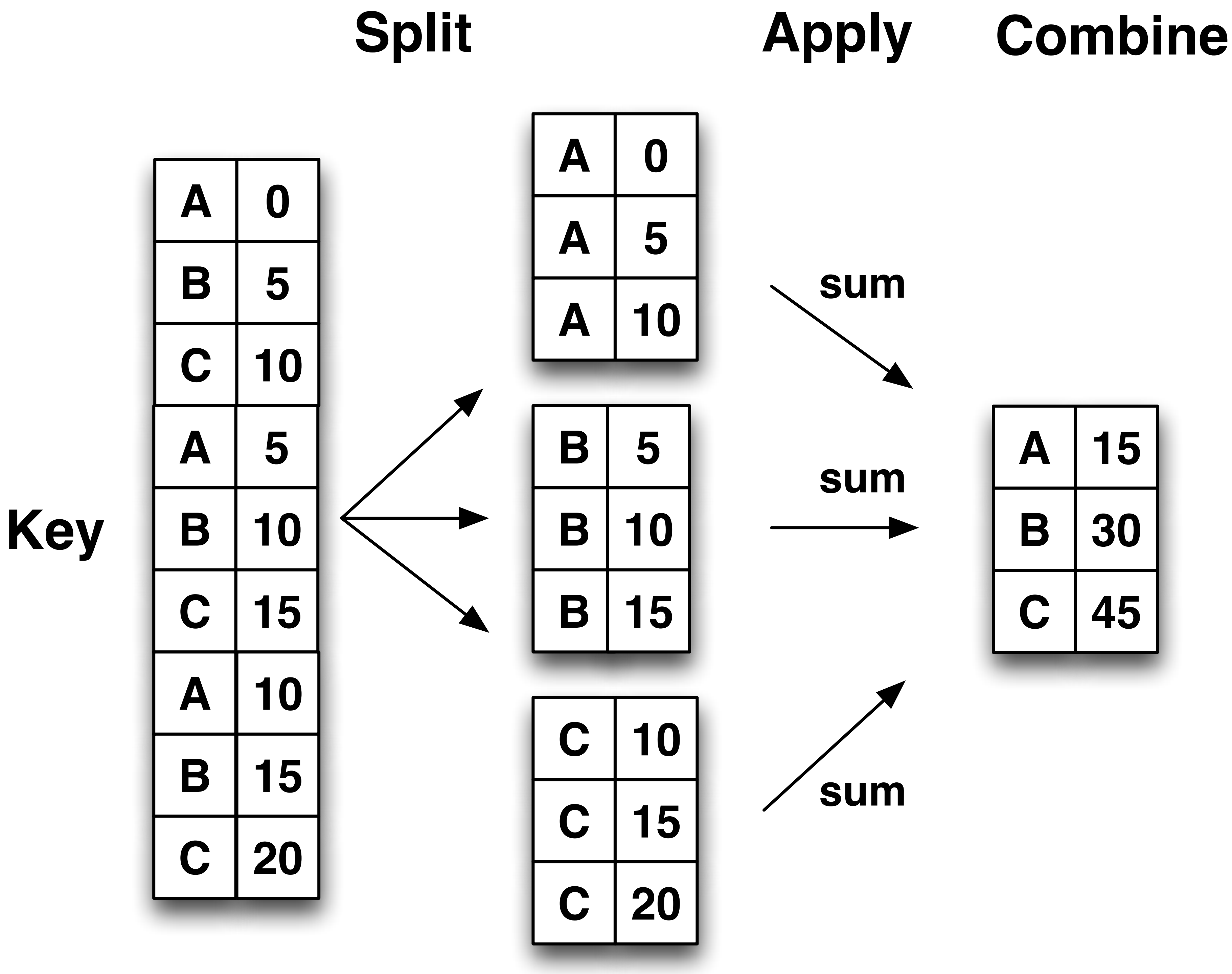
+

A	0
B	1
C	2
D	3

=

A	NA
B	2
C	4
D	6
E	NA

GroupBy



Split

Apply

Combine

Key

A	0
B	5
C	10
A	5
B	10
C	15
A	10
B	15
C	20

A	0
A	5
A	10

B	5
B	10
B	15

C	10
C	15
C	20

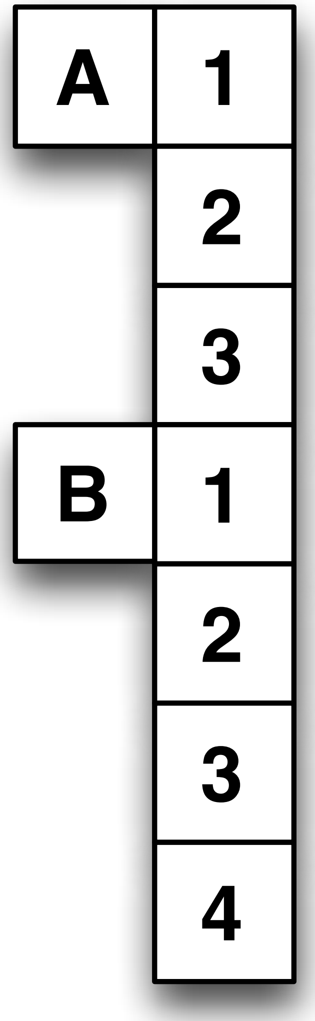
sum

sum

sum

A	15
B	30
C	45

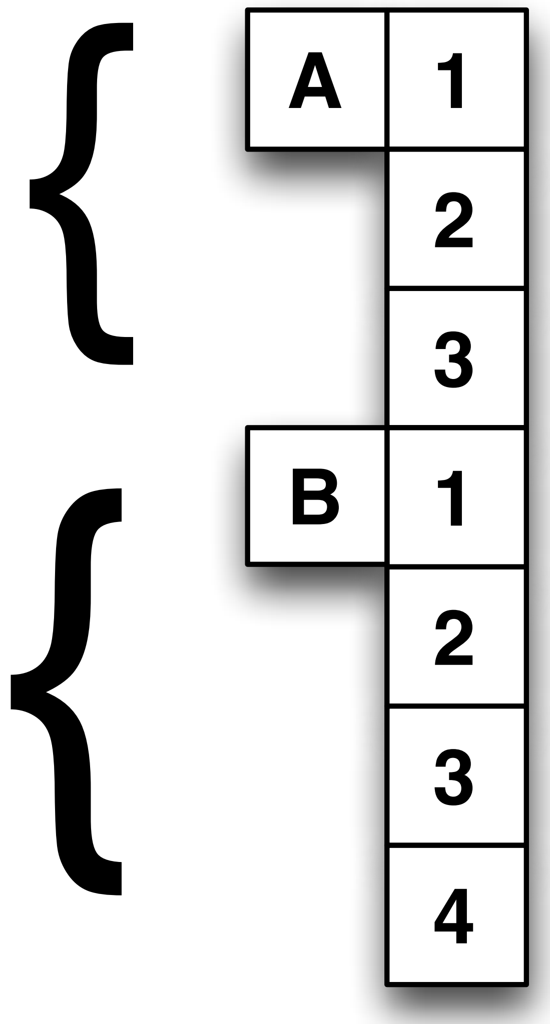
Hierarchical indexes



A	1
	2
	3
B	1
	2
	3
	4

- Semantics: a tuple at each tick
- Enables easy group selection
- Terminology: “multiple levels”
- Natural part of GroupBy and reshape operations

Hierarchical indexes



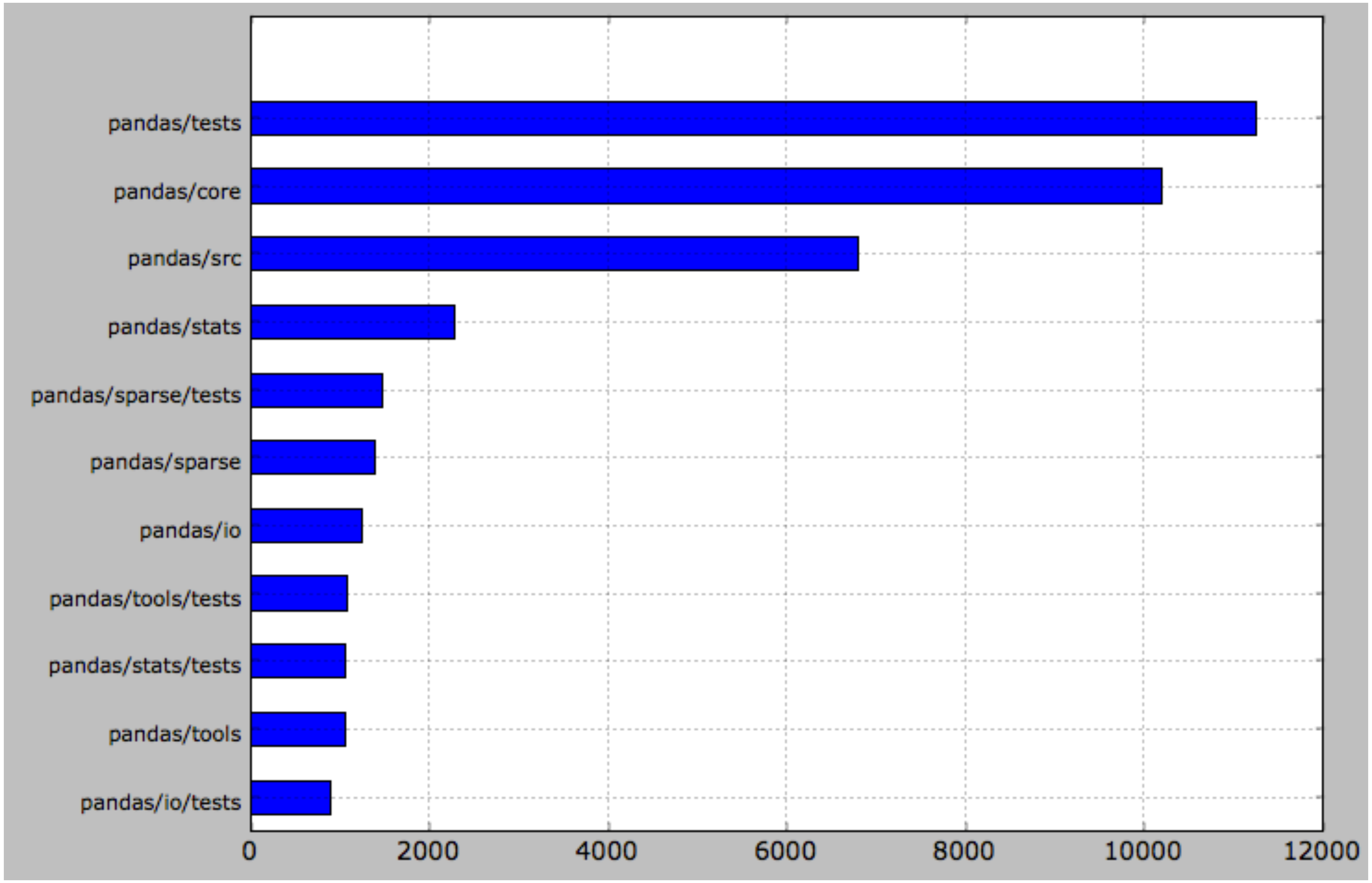
- Semantics: a tuple at each tick
- Enables easy group selection
- Terminology: “multiple levels”
- Natural part of GroupBy and reshape operations

Let’s have a little fun

To the IPython Notebook!

What’s in pandas?

- A big library: 40k SLOC



Tests!

- Huge accumulation of use cases originating in real world applications
- 68 lines of tests for every 100 lines of code

I DON'T ALWAYS TEST MY CODE
BUT WHEN I DO I DO IT IN PRODUCTION
memegenerator.net

pandas.core

- Data structures
  - Series (1D)
  - DataFrame (2D)
  - Panel (3D)
- NA-friendly statistics
- Index implementations / label-indexing

pandas.core

- GroupBy engine
- Time series tools
  - Date range generation
  - Extensible date offsets
- Hierarchical indexing stuff

Elsewhere

- Join / concatenation algorithms
- Sparse versions of Series, DataFrame...
- IO tools: CSV files, HDF5, Excel 2003/2007
- Moving window statistics (rolling mean, ...)
- Pivot tables
- High level matplotlib interface

Hmm, pandas/src

- ~6000 lines of mostly Cython code
- Fast data algorithms that power the library and make it fast
- pandas in PyPy?

Ok, so why Python?

- Look around you!
- Build a superior data analysis and statistical computing environment
- Build mission-critical, data-driven production systems

Trolling #rstats

Hash tables, anyone?

The pandas roadmap

- Improved time series capabilities
- Port GroupBy engine to NumPy only
- Better integration with statsmodels and scikit-learn
- R integration via rpy2

The pandas roadmap

- Integration with JavaScript visualization frameworks: D3, Flot, others
- Alternate DataFrame “backends”
  - Memory maps
  - HDF5 / PyTables
  - SQL or NoSQL-backed
- Tighter IPython Notebook integration

ggplot2 for Python

- We **need** to build better a better interface for creating statistical graphics in Python
- Use pandas as the base layer !
- Upcoming project from Peter Wang: bokeh

pandas for “Big Data”

- Quite common to need to process larger-than-RAM data sets
- Alternate DataFrame backends are the likely solution
- Ripe for integration with MapReduce frameworks

Better time series

- Integration of scikits.timeseries codebase
- NumPy datetime64 dtype
- Higher performance, less memory

Better time series

- Fixed frequency handling
- Time zones
- Multiple time concepts
  - Intervals: 1984, or “1984 Q4”
  - Timestamps: moment in time, to micro- or nanosecond resolution