Lift your Speed Limits with Cython

Stefan Behnel <stefan.behnel@trustyou.com>
TrustScore® and Meta-Reviews
TrustYou Analytics

3 Mio. new text reviews every week
Why do we use Python?

- **Great** programming language
  - versatile, pragmatic, concise, readable
- **Community** is diverse, friendly and helpful
  - well supported and documented
- **Excellent** libraries and tools
  - big data, NLP, web services, automation, testing, ...
Tools we use at TrustYou

- NumPy – fast matrix/array computation
- SciPy – advanced scientific algorithms
- Scikit-learn – machine learning
- Pandas – data analysis
- NLTK, SpaCy – text analysis
- Tornado, asyncio – async web services
- lxml – XML / HTML parsing + processing
- ...
Tools we use at TrustYou

- NumPy
- SciPy
- Scikit-learn
- Pandas
- NLTK, SpaCy
- Tornado, asyncio
- lxml
- ...

Python Data ecosystem:
- NumPy integrates Data
- Cython integrates Code
Why Cython?

- Fastest way to integrate native code
- Production proven
- Widely used
- All about «getting things done»
  - Pragmatic programming language
  - Keeps focus on functionality, not boilerplate
  - Allows moving freely between Python and C/C++
  - As pythonic as you want, as native as you need
- "We write C so you don‘t have to"
Demo
Ideas for future features

- Comprehensions into C++ containers
  - `vector(i*2 for i in range(8))`
- Inline generators into loops
  - also in OpenMP parallel code
- `@total_ordering` decorator for extension types (C-implemented classes)
  - less code for fast object comparisons
- More sophisticated type inference
Thanks!
Any questions?

You can reach me at: stefan.behnel@trustyou.com

Oh, and visit www.trustyou.com
- check your next hotel
- we’re hiring :)