

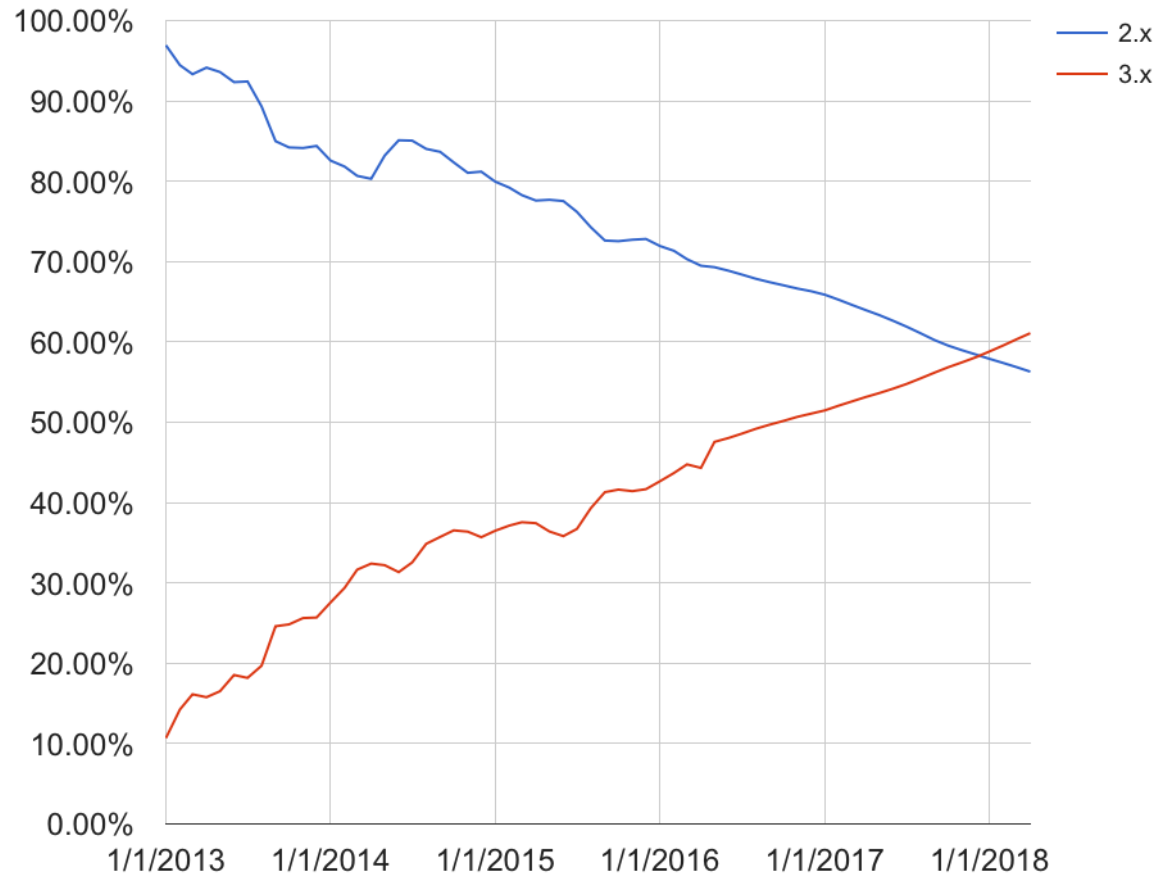


PyCharm Support for PEP 484

Andrey Vlasovskikh
PyCharm Community Lead

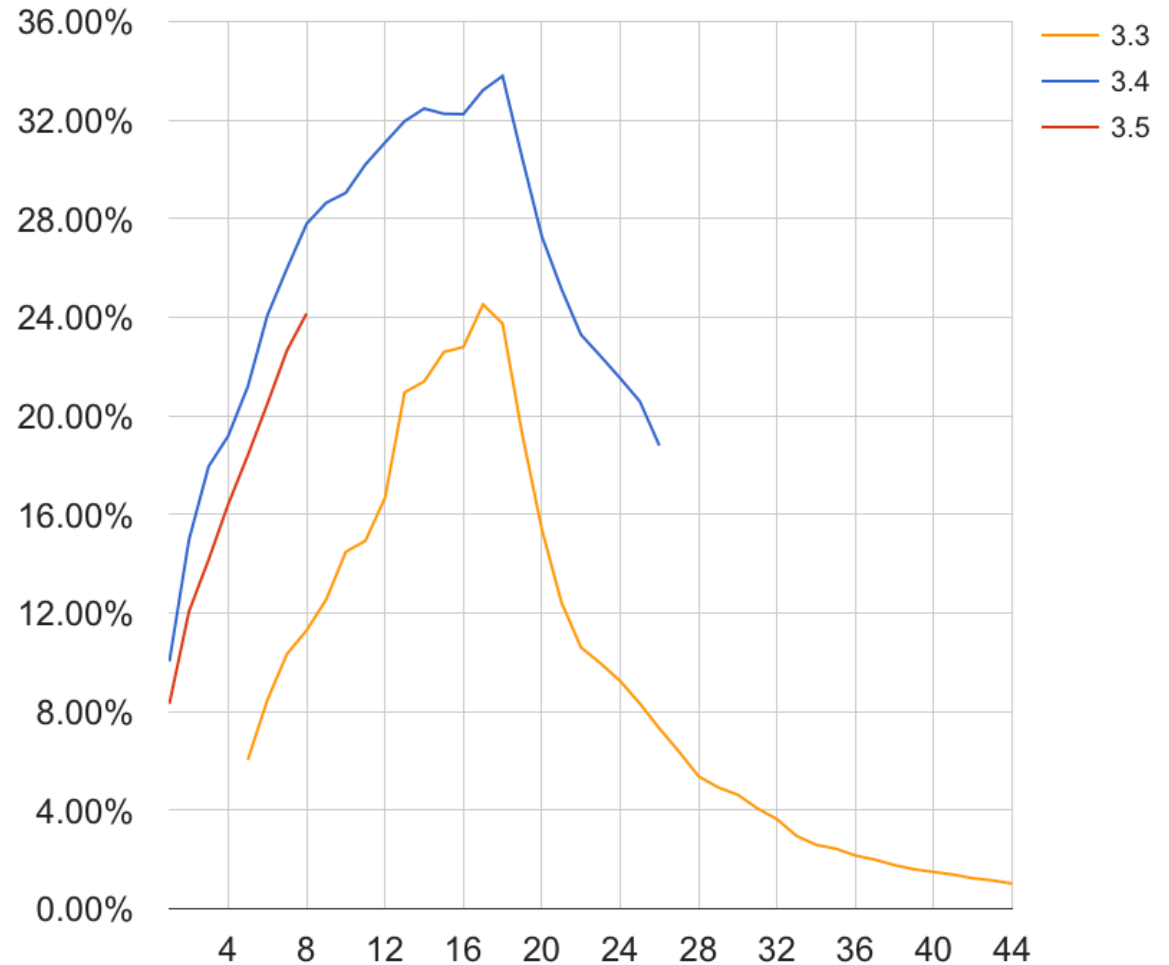
Python 2 vs 3

- Predicted equal usage in December 2017
- Up to 20% of people use both Python 2 and 3



3.x Adoption

- Python 3.5 adoption rate is 4% slower than Python 3.4



Switch to PEP 484

- PyCharm used to have its own type system similar to PEP 484
- It wasn't hard for us to add support most of PEP 484
- Some pieces of PEP 484 are still not supported



Feedback from Users

- Adoption rate
 - PyCharm supports PEP 484 since November 2015
 - People started to use type hints since March 2016
 - No comprehensive statistics yet
- Problems
 - Some people are confused with Optional vs default arguments
 - People wanted PEP 484 for Python 2 (solved in the updated PEP)



Not Supported Yet

- Strict union checks, including Optional checks
- Type variance
- Python version and platform checks
- New `typing.Type[C]`
- Use stubs over original `*.py` files
 - We combine data from both sources, will switch to stubs-only
- Typed repository
 - Incomplete stubs, versioning, update policy



Text / Binary Types for Python 2 + 3

- Important for Python 2 to 3 migration
- Guido added `typing.Text` in 3.5.2
- We sent two proposals to python-ideas
 - Track ASCII values to check implicit conversions (see Mypy#1141)
 - Give up on strict text / binary checking in Python 2 (rejected)
- Let's work on text / binary types for 2 + 3 **at the PyCon sprints!**
 - I'm [@vlasovskikh](https://twitter.com/vlasovskikh) on Twitter

