Do Java Programmers Write Better Python?

Studying Off-language Code Quality on GitHub

Siegfried Horschig, Toni Mattis, Robert Hirschfeld

Software Architecture Group
Hasso Plattner Institute, University of Potsdam, Germany

PX/18 10 Apr. 2018, Nice, France
semicolon

terminating lines since ~1958
Java
C++
C
JavaScript
C#

Python?

separator: foo.x(); foo.y()

works, but discouraged!
foo.x(); foo.y()

semicolon terminating lines since ~1958
So, how many **Java programmers** accidentally write...

```
foo.x();
foo.y()
```

... in Python?

---

**semicolon**

terminating lines since ~1958
Finding Off-language Programmers

» How to find programmers that
  › primarily work with **Java**, but
  › occasionally switch to **Python**?

» Idea: Open source contributors on GitHub
  › Find **user accounts** that commit many LOC to *.java* files
  › ... and sometimes *.py* files
  › check their **Python** code

» We have a copy of GitHub, based on GHTorrent*
  › ~10TB of commits and user data in PostgreSQL
  › ~250,000 full git repositories on disk

*) [http://ghtorrent.org/](http://ghtorrent.org/)
Candidate Selection

- **Java contributions**: LOC changed in *.java files
- **Python contributions**: LOC changed in * .py files

\[ LOC_{Java} \geq 5 \times LOC_{Python} \]
\[ LOC_{Python} \geq 150 \]

84 Candidates of 14,380,149 users
SELECT author_id,
    sum(pycommit.changes) as pychanges,
    sum(jcommit.changes) as jchanges
FROM
    (SELECT author_id, sha FROM commits) author
JOIN
    (SELECT sha, changes FROM raw_patches WHERE name LIKE '%.py') pycommit
ON author.sha = pycommit.sha
JOIN
    (SELECT sha, changes FROM raw_patches WHERE name LIKE '%.java') jcommit
ON author.sha = jcommit.sha
GROUP BY author_id
HAVING pychanges > 150
AND jchanges > (pychanges * 5);
Candidates and Projects

84 Java Candidates
40 Projects
5 out of 1000 LOC (480,875 LOC in total)

91 C++ Candidates
33 Projects
24 out of 1000 LOC (175,402 LOC in total)

100 Control Group
(of 1800 Candidates)
380 Projects
1 out of 1000 LOC (1,335,220 LOC in total)

Counting end-of-line semicolons...

Horschig, Mattis, Hirschfeld | PX/18, Nice, 2018 | Software Architecture Group, HPI Potsdam
**PyLint**: Few/Many Methods per Class

**Hypothesis**: Java/C++ programmers are forced into class-based OOP. They should excel at writing classes.

<table>
<thead>
<tr>
<th>Language</th>
<th>Classes with too many Methods</th>
<th>Classes with too few Methods (Data Class)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Java</td>
<td>0.32 of 1000 LOC</td>
<td>5.39 of 1000 LOC</td>
</tr>
<tr>
<td>C++</td>
<td>0.18 of 1000 LOC</td>
<td>3.09 of 1000 LOC</td>
</tr>
<tr>
<td>Python</td>
<td>0.69 of 1000 LOC</td>
<td>9.22 of 1000 LOC</td>
</tr>
</tbody>
</table>
Some Tentative Conclusions

» Knowing a language can influence your code style in another language...
  › **positively** regarding generalizable knowledge (e.g. OOP)
  › **negatively** regarding peculiarities (e.g. line endings, indentation, built-in names, ...)

› **Consequence:** The order in which we learn/teach languages likely influences our/students' success at another language

» The **GHTorrent** dataset allows to study such effects with little effort compared to user studies
Correlation | Causation | Coincidence

» **Common unobserved factor** that attracts both semicolons and C++ developers
Insights limited

» **Random variation** independent of language
Limited control via "p < 0.05"

» **Selection Bias**
< 0.002% of all GitHub users
Pipeline

GHTorrent → CSV → Documents (MongoDB) → GitHub

GitHub Dataset

Commits → Candidate users → Project IDs

Repositories

Projects → PyLint