Measuring Open-Source Software as an Intangible, Digital Asset using GitHub

Sixth World KLEMS Conference

Digital Economy Session March 16, 2021

Carol Robbins, NCSES

National Center for Science and Engineering Statistics
Social, Behavioral and Economic Sciences
National Science Foundation
Collaborators

<table>
<thead>
<tr>
<th>Name</th>
<th>Position</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gizem Korkmaz</td>
<td>Associate Professor, Biocomplexity Institute, UVA</td>
</tr>
<tr>
<td>Ledia Guci</td>
<td>Senior Analyst, NCSES, NSF</td>
</tr>
<tr>
<td>Bayoán Santiago Calderón</td>
<td>Postdoctoral Research Associate, Biocomplexity Institute, UVA</td>
</tr>
<tr>
<td>Brandon Kramer</td>
<td>Postdoctoral Research Associate, Biocomplexity Institute, UVA</td>
</tr>
</tbody>
</table>

**Disclaimer:** The views expressed in this paper are those of the authors and not necessarily those of their respective institutions.

**Acknowledgments:** This material is based on work supported by U.S. Department of Agriculture (58-3AEU-7-0074) and the National Science Foundation (Contract #49100420C0015)
Open-Source Software: an Intangible Digital Asset

“Open Source Software (OSS) is a computer software, with its source code made available with a license, in which the copyright holder provides the rights to study, change, and distribute the software to anyone and for any purpose.” (Open Source Initiative)

Developed, maintained and extended by:

- **universities** (e.g., Stanford, MIT, UC, Berkeley)
- **businesses** (e.g., Microsoft, Google)
- **government research institutions** (e.g., Sandia National Lab)
- **Nonprofits**
- **individuals**
Where is it coming from and who is creating it?
Overview

- Motivation
- Knowledge outputs and the System of National Accounts
- Data Discovery
- Quantity/Volume
- Sector and Country
- Where we are headed: time series investment and capital stock
NCSES's mandate is the collection, interpretation, analysis, and dissemination of objective data on the science and engineering enterprise.

NCSES’s mission:
• Research and Development
• The science and engineering workforce
• U.S. competitiveness in science, engineering, technology, and R&D
• The condition and progress of STEM education in the United States

Data Products include:
• Workforce Statistics
• R&D Statistics
• Business Innovation Statistics
• Indicators of Research, Invention, and Innovation
2018 Oslo Manual Promotes Bringing Innovative Knowledge into the SNA

- Integrating Innovation Data with SNA sources
- 2018 Revision of Oslo Manual
  - SNA framework recommended for collection of innovation statistics
  - Use SNA terminology where applicable
  - Innovation in all SNA sectors should follow SNA
    o Business
    o General government
    o Non-profit institutions serving households
    o Household
- Not going to happen all at once
  o Universities
Inspiration

• Corrado, Hulten and Sichel: measuring intangibles
  "Measuring Capital and Technology: An Expanded Framework", in Measuring Capital in the New Economy, 2005

• von Hippel: motivations of open-source software developers
  "Open Source Software Projects as user Innovation Networks - No Manufacturer Required."

• Greenstein and Nagel: measuring Apache servers as substitutes

• Sichel and von Hippel: measure household innovation based on time spent doing it.
  "Household Innovation and R&D: Bigger than You Think.“ Review of Income and Wealth.
Data Development Questions

• How much is created each year? (flow measure)
• How much open-source software is in use? (stock measure)
• Who creates it? (Sectors: Business, Government, Academia, Households, Nonprofits, Foreign)
• What data can be used to develop a volume measure?
• What depreciation rates and deflators are appropriate?
Prototype for one Programming Language

<table>
<thead>
<tr>
<th>OSS Definition</th>
<th>Registry</th>
<th>Repository (GitHub)</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Open Source Initiative (OSI) -Approved License</td>
<td>• Language-specific package managers (e.g., CRAN, PyPI)</td>
<td>• Commit Data (who, what, when)</td>
</tr>
<tr>
<td>• Production Ready</td>
<td>• Contains metadata</td>
<td>• License</td>
</tr>
<tr>
<td>• Release for Current Ecosystem</td>
<td>• Continuous integration</td>
<td>• Profile of contributors</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Language</th>
<th>R</th>
<th>Python</th>
</tr>
</thead>
<tbody>
<tr>
<td>Package manager</td>
<td>CRAN</td>
<td>PyPI</td>
</tr>
<tr>
<td>Number of packages</td>
<td>13,719</td>
<td>164,836</td>
</tr>
<tr>
<td>Production ready</td>
<td>13,350</td>
<td>17,482</td>
</tr>
<tr>
<td>OSI-approved</td>
<td>13,143</td>
<td>15,043</td>
</tr>
<tr>
<td>Packages on GitHub (analysis)</td>
<td>4,358</td>
<td>9,773</td>
</tr>
</tbody>
</table>

- The registry data was collected using web harvest techniques.
- All CRAN and PyPI data as of July 2017, **14K R and Python packages** for analysis.


Defining the Scope of OSS in the US

- Software that is published under an Open Source Initiative **OSI-approved license**.
  - Licenses establish permissions (e.g., use, inspect, modify, distribute, attribution) and limitations (e.g., liability, warranty).
  - Most common licenses are: MIT, Apache, GPL.

From prototype to scale-up:

1. **Packages for programming languages R and Python**
   These are published codebases that are discoverable and installable through a registry and package manager.

2. **GitHub repositories**
   Repositories on GitHub, the world's largest remote hosting platform for Git version control.
Scale Up Data Collection: GitHub Repositories

- GHTorrent project data for additional user information (e.g., organization, company, location, email)
- Find public repositories with an OSI-approved license
- Collect information on development activity (e.g., commits, additions) and contributors using the GraphQL API.
- Obtained **7.75M repositories** (2009-2019) and **3.26M distinct contributors**

Quantity/Volume of Output: How much is that?

Project length and complexity determine effort.


- Effort is a nonlinear function of complexity and lines of code
  - Code lines measured per project
  - Historical software project factors

\[
Effort = 2.4(KLOC)^{1.05}
\]

Nominal development time = \(2.5(Effort)^{3.8}\)

Labor cost = Monthly wage \(\times\) Nominal development time
In Dollars, What Would that Imply?

Total resource cost = Resource cost (month) \times \text{Nominal development time}

Labor costs: wage and salary plus nonwage compensation
Intermediate input costs
Taxes on production
Gross operating surplus

Prototype: 14K open-source packages registered in PyPI and CRAN and hosted on GitHub: $2.4 \text{ billion} (in 2017 dollars)
Scaleup: 7.75M GitHub repositories with OSI-approved licenses in 2019 investment total: 2.6M repos, cost based on lines added: $512 \text{ billion} (2019)

We can directly attribute $33 \text{ billion} to US contributors in 2019.
Sectoral Contributions

- Multiple data sources and methods used to estimate *contribution of each sector* taking into account collaborations across sectors

Use company field and emails in GHTorrent data to map developers to sectors. Mapped 20.4% of GHTorrent users to sectors. 12% of the total activity is captured.

Country-level Contributions

- **Contribution of each country** taking into account international collaborations (e.g., fractional counting).

Using self-reported location information in GHTorrent to map developers to countries (ISO-2C country codes, regular expressions, major cities, spelling fixes)

**Mapped 19.7% users in GHTorrent to countries. 33% of the total activity is captured.**

US contributions are estimated as a third of the total contributions mapped to countries (35%)

## Software Investment in Economic Output

<table>
<thead>
<tr>
<th>Components of Software Investment</th>
<th>Private Sector</th>
<th>Public Sector</th>
<th>Household Sector</th>
<th>Rest of World</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prepackaged</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Custom</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Proprietary</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Open Source (OSS)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Own-account</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Proprietary</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Open Source (OSS)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

What we have learned so far

Quantities: Lines of Code and repositories

Contributors: by sector, academics will take more parsing

Countries: many contributors can be assigned
From Investment to Stock of Intangible Digital Assets

Next for us:
Annual Output/Volume based on own-account investment method, sum of costs
• Annual GitHub Volume: 2009-2019
• Price index: Own account software
• Depreciation rate: own account software

Measurement Questions
   Can this approach translate to the creation of software in other economies (I/O ratios consistent)?
   Does own-account software depreciate at the same or different rate than proprietary software?
National Center for Science and Engineering Statistics

https://ncses.nsf.gov