

Comparing Repositories Visually with RepoGrams

<http://repograms.net>

Daniel Rozenberg, [Ivan Beschastnikh](#), Fabian Kosmale,
Valerie Poser, Heiko Becker, Marc Palyart, Gail C. Murphy



University of British Columbia



Saarland University

Big (SE) data



GitHub



Bitbucket

- Millions of projects
- Open APIs
- Meticulously tracked and archived activity

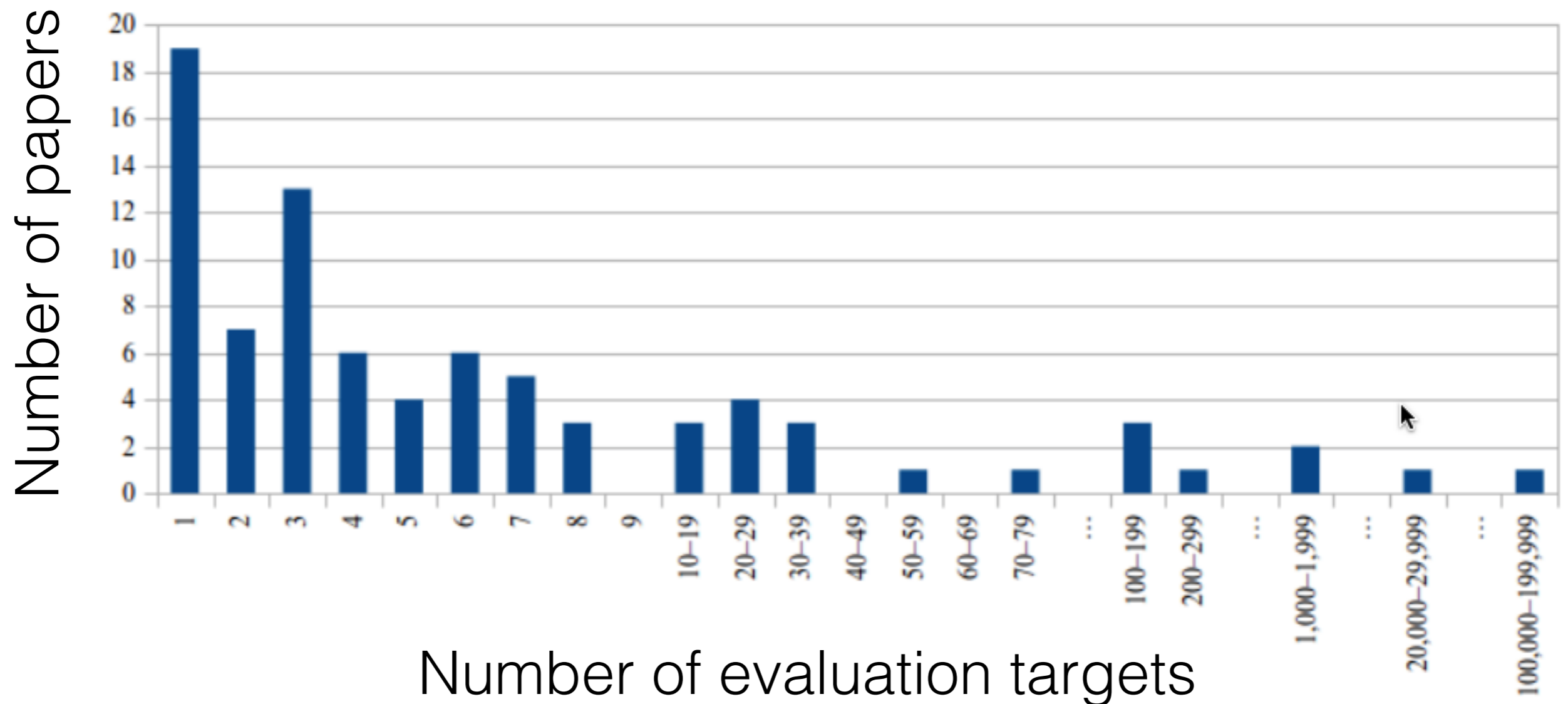


- Huge opportunity for researchers
- Each open source project is a potential evaluation target!

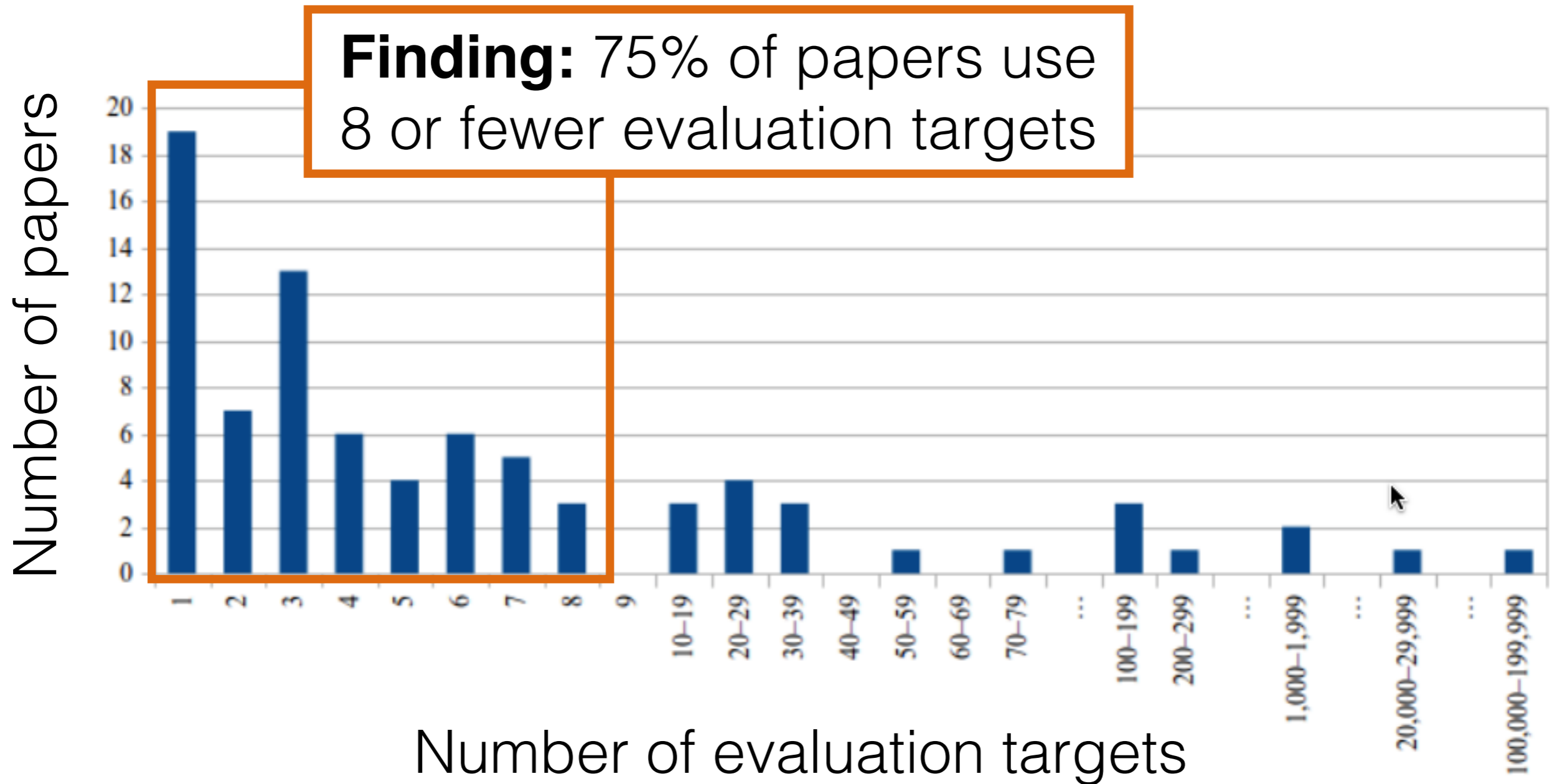
How many projects do paper authors use in their evaluation?

- **Experiment:** selected 114 papers from ICSE, FSE, ASE, MSR, ESEM (years 2012-2014)
- Recorded number of targets that the authors claim to evaluate

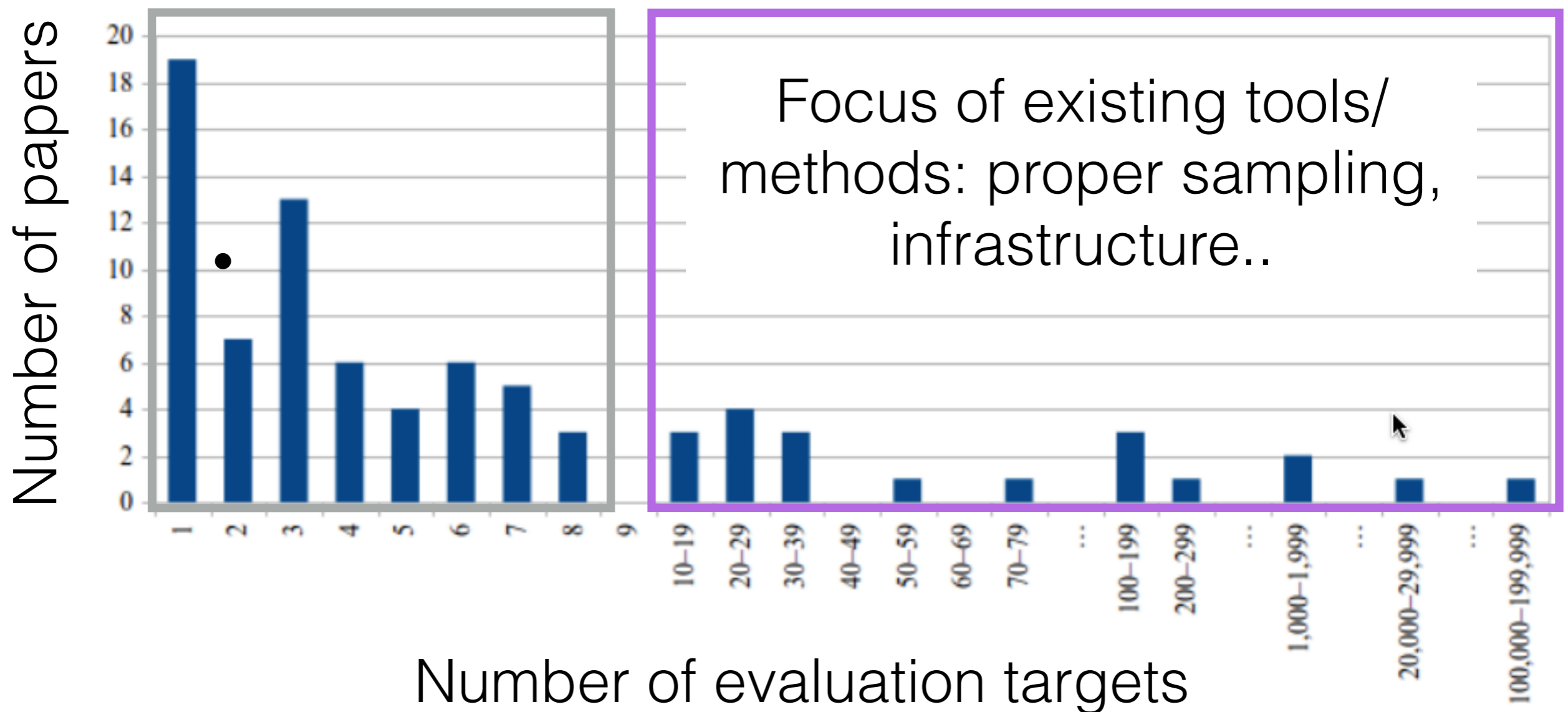
How many projects do paper authors use in their evaluation?



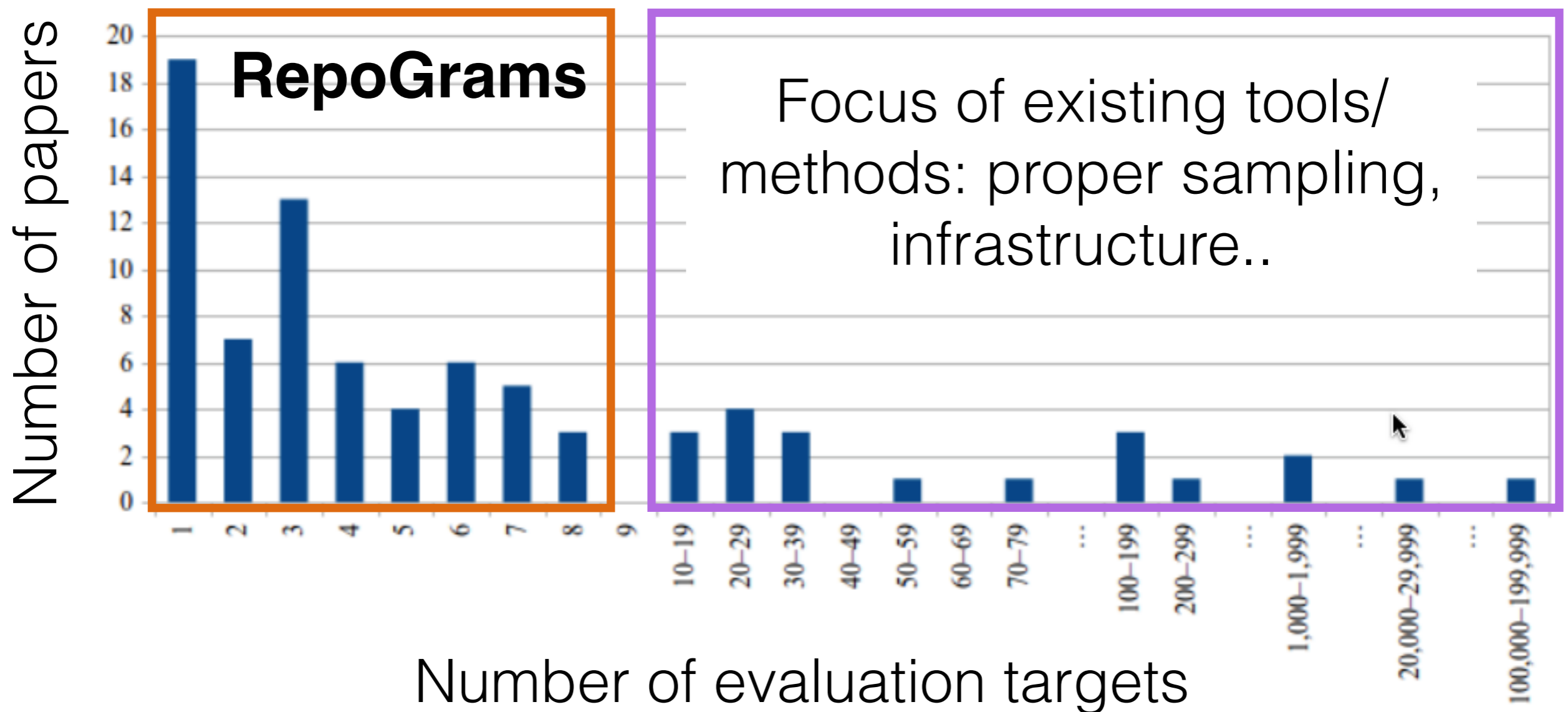
How many projects do paper authors use in their evaluation?



Existing tools focus on supporting scalable analysis



Existing tools focus on supporting scalable analysis



RepoGrams:

Qualitative repository analysis

Presents data in a way that can be observed but not measured

RepoGrams:

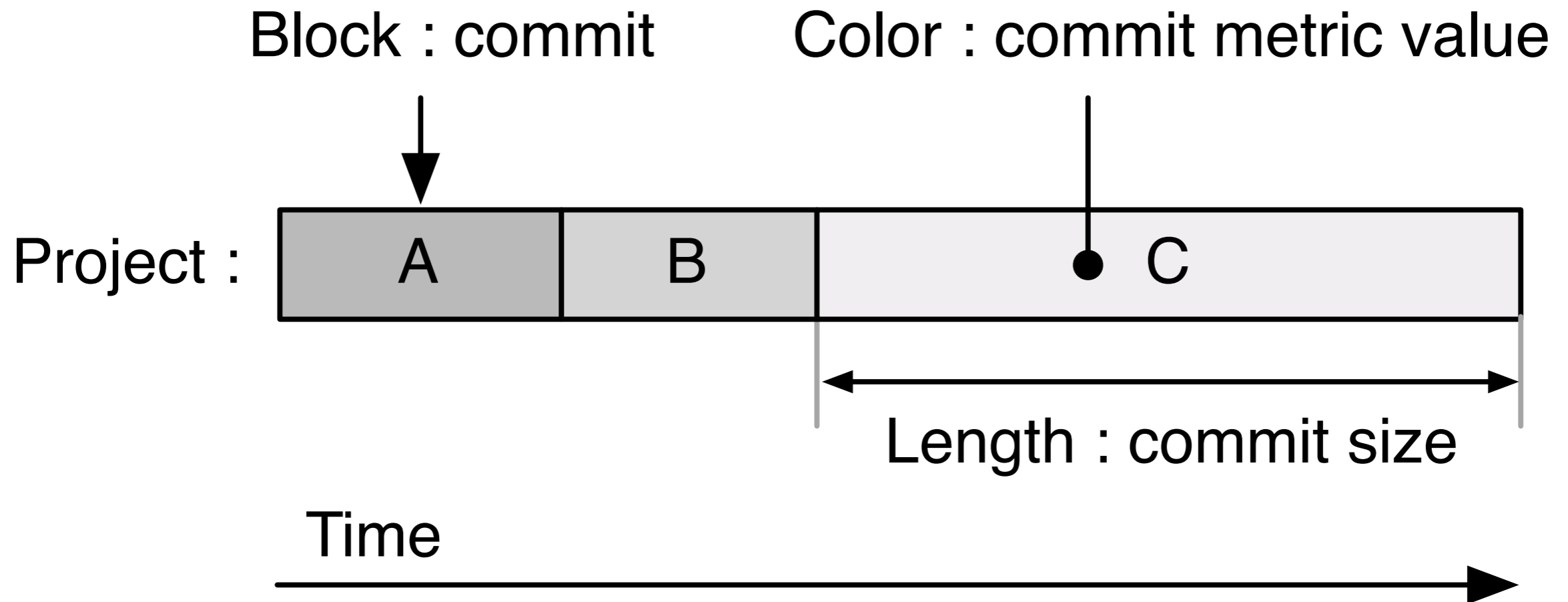
Qualitative repository analysis

Presents data in a way that can be observed but not measured

- Goal is not to provide an answer, but to **surface relevant information**
- Help the user think critically/contrast relevant features of a (small number of) projects
- Support curation of a small number of project (≤ 8)

Visualization: a natural fit for qualitative analysis & nuance

Core abstraction in RepoGrams: Repository “footprint”



Demo: the basics



Settings Load/save state Help

Group by metric Metric (block color): Commit Author Switch... Block length mode: Fixed Switch... Zoom: - x1 +
 Group by repository Normalization mode: Globally normalized x1 x100

Repositories Earliest commits Latest commits

Commit Author ? Legend: ... unique authors

Colors for unique branches and unique authors metric are incomparable between projects.

passenger-docker A horizontal bar chart representing the commit history for the 'passenger-docker' repository. The bar is composed of many small colored blocks. The colors are primarily yellow, with some purple, blue, and green blocks interspersed. The blocks are of uniform width, indicating a constant commit block width. The colors represent different authors, with yellow being the most frequent.

GIT clone URL, e.g. <https://github.com/githubtraining/hellogitworld.git> + Add Random

Commit author metric:
one unique color per
author

Constant commit
block width

Demo: comparing two metrics

Commit Author ?

Legend:  ... unique authors

Colors for unique branches and unique authors metric are incomparable between projects.



Branches Used ?

Legend:  master  ... other branches

Colors for unique branches and unique authors metric are incomparable between projects.



Branches used metric:
one unique color per
branch; master is always
red

Demo: we can represent many things with a footprint

Commit Author ?

Legend: ... unique authors

Colors for unique branches and unique authors metric are incomparable between projects.



Branches Used ?

Legend: master ... other branches

Colors for unique branches and unique authors metric are incomparable between projects.



Commit Age ?

Legend: Less than 1 minute 1-59 minutes 1-2 hours 2-12 hours 12-24 hours 1-2 days 2-7 days More than 7 days



Commit age metric:
elapsed time between
commit and its parent

Demo: block width can denote magnitude of change

Commit Author ?

Legend: ... unique authors

Colors for unique branches and unique authors metric are incomparable between projects.



Branches Used ?

Legend: master ... other branches

Colors for unique branches and unique authors metric are incomparable between projects.



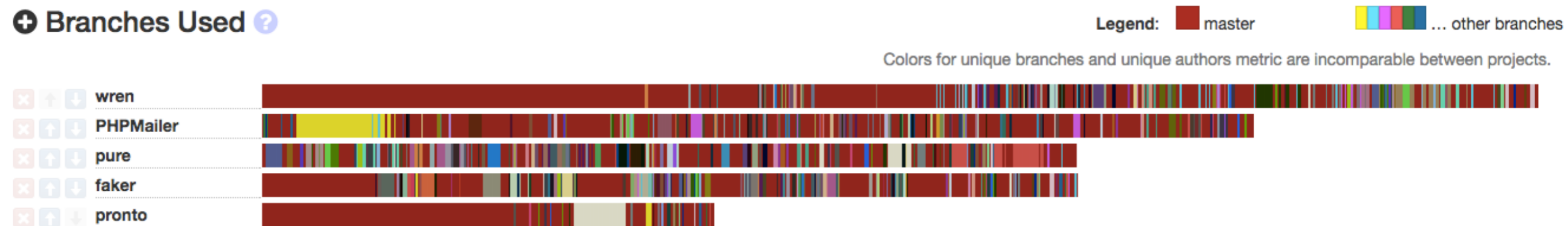
Commit Age ?

Legend: Less than 1 minute 1-59 minutes 1-2 hours 2-12 hours 12-24 hours 1-2 days 2-7 days More than 7 days



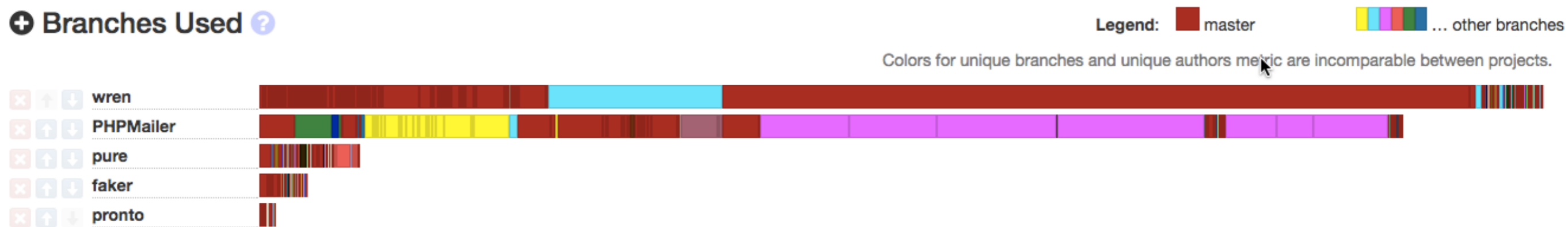
Block width: linear in the
LOC changed in commit

Demo: multiple projects



- *wren* has more commits than any other projects
- *wren*, *faker*, *pronto*, use **master** initially
- All projects eventually use a diversity of branches

Demo: multiple projects



- *wren* and *PHPMailer* have much larger commits
- PHPMailer has huge commits in the **purple** and **yellow** branches

Evaluation questions

RQ1: Can SE researchers use RepoGrams to understand and compare characteristics of a project's source repository?

RQ2: Will SE researchers consider using RepoGrams to select evaluation targets for experiments and case studies?

RQ3: How much effort is required to add metrics to RepoGrams?

Methodology

RQ1: Can SE researchers use RepoGrams to understand and compare characteristics of a project's source repository?

RQ2: Will SE researchers consider using RepoGrams to select evaluation targets for experiments and case studies?

RQ3: How much effort is required to add metrics to RepoGrams?

- 14 authors from MSR'14
- Tasks using RepoGrams
- Semi-struct. interviews

- 2 developers
- Each implemented 3 metrics

Evaluation highlights

RQ1: Can SE researchers use RepoGrams to understand and compare characteristics of a project's source repository?

RQ2: Will SE researchers consider using RepoGrams to select evaluation targets for experiments and case studies?

RQ3: How much effort is required to add metrics to RepoGrams?

- ◆ Successfully used RepoGrams for complex tasks

- ◆ Tools is of immediate use

- ◆ Researchers want custom metrics

- ◆ Setup: 1.5 hours

- ◆ Metric: avg/max = 40/52 min

- ◆ < 40 LOC total

Related work

- **Helping researchers with the selection process**
 - *Tools/Datasets:* GHTorrent, Boa, MetricMiner
 - *Methods:* “Diversity in software engineering research”, FSE13
- **Visualization**
 - *Tools:* CVSgrab, ConcernLines, Fractal Figures, Chronos, RelVis, Chronia, Evolution radar



RepoGrams



- * Lots of data, many potential evaluation targets!
- * But, proper **project selection is complex**
 - * Researcher must be highly aware of the features of the project that may influence the study results

- ◆ **RepoGrams: supports qualitative analysis of software repositories**
 - ◆ Presents data in a way that can be observed but not measured

Try our public deployment!

<http://repograms.net>