End-User Programmers and their Communities: An Artifact-based Analysis

Kathryn T. Stolee, Sebastian Elbaum, and Anita Sarma
University of Nebraska–Lincoln

{kstolee, elbaum, asarma}@cse.unl.edu

September 22, 2011

This work is supported by the NSF GRFP under CFDA#47.076, NSF Award #0915526, and AFOSR Award #9550-10-1-0406.

Introduction

End User Programmers

People who engage in programming activities to support their hobbies and work.

Introduction

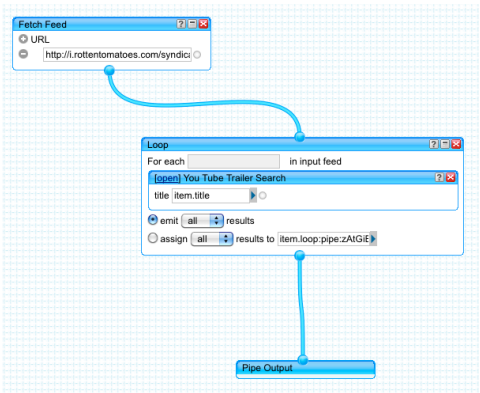
End User Programmers

People who engage in programming activities to support their hobbies and work.

	Professionals	**End Users**
Number in U.S.	3 million	13 million
Typical Education	C.S. Degree	Other Degree
Role of Programming	It's their job	It supports their job

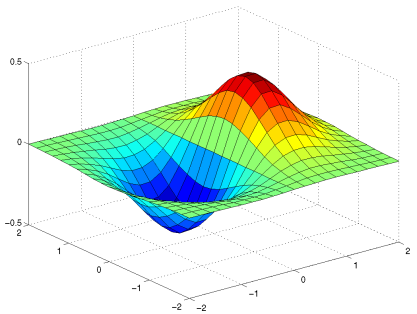
Many Domains and Applications

Web Mashups:



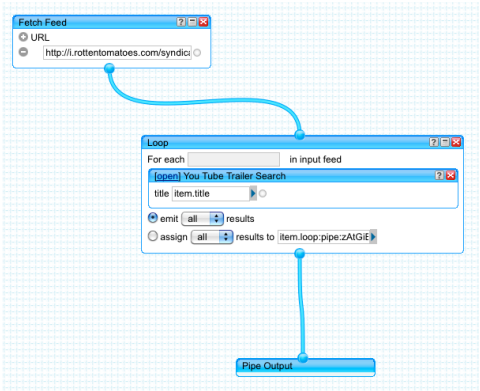
Educational Games:

Scientific Computing:



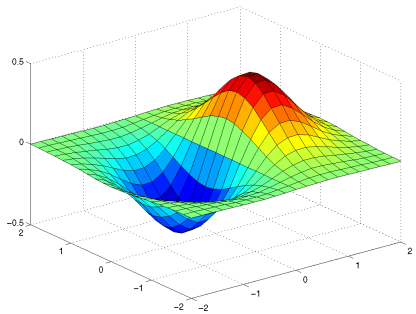
Many Domains and Applications

Web Mashups:



Educational Games:

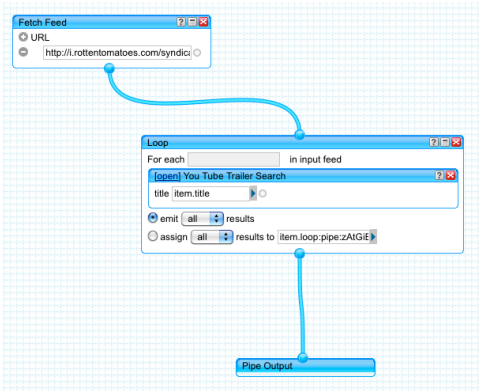
Scientific Computing:



Environment	Yahoo! Pipes	Scratch	MATLAB
# Artifacts	100,000	700,000	13,717
# Participants	90,000	500,000	5,356

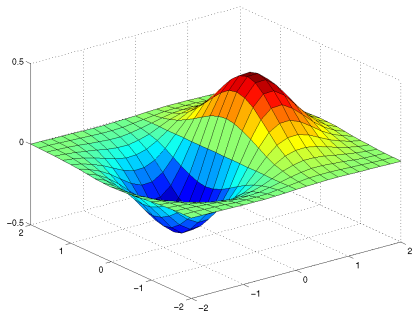
Many Domains and Applications

Web Mashups:



Educational Games:

Scientific Computing:



Environment	Yahoo! Pipes	Scratch	MATLAB
# Artifacts	100,000	700,000	13,717
# Participants	90,000	500,000	5,356

. . . yet we know little about **how** the repositories are utilized

Empirical Study Details

- **Research Goal**
- Study Context
- Research Questions
- Variables and Metrics
- Methods
- Results

Research Goal

To better understand *end-user programmer* communities

- Learn how communities and artifact repositories evolve
- Uncover needs for support in: development, maintenance, search, program understanding, . . .

Empirical Study Details

Goal: To better understand *end-user programmer* communities

- Research Goal
- **Study Context**
- Research Questions
- Variables and Metrics
- Methods
- Results

Why Mashup Communities?

Web Mashups

Applications that compose and manipulate existing data sources or services to create new data or service.

Why study mashups?

- Many environments (e.g., Apatar, DERI Pipes, IBM Mashup Center, Kivati, Yahoo! Pipes, . . .)
- Potential impact (many users, growth)

Why Mashup Communities?

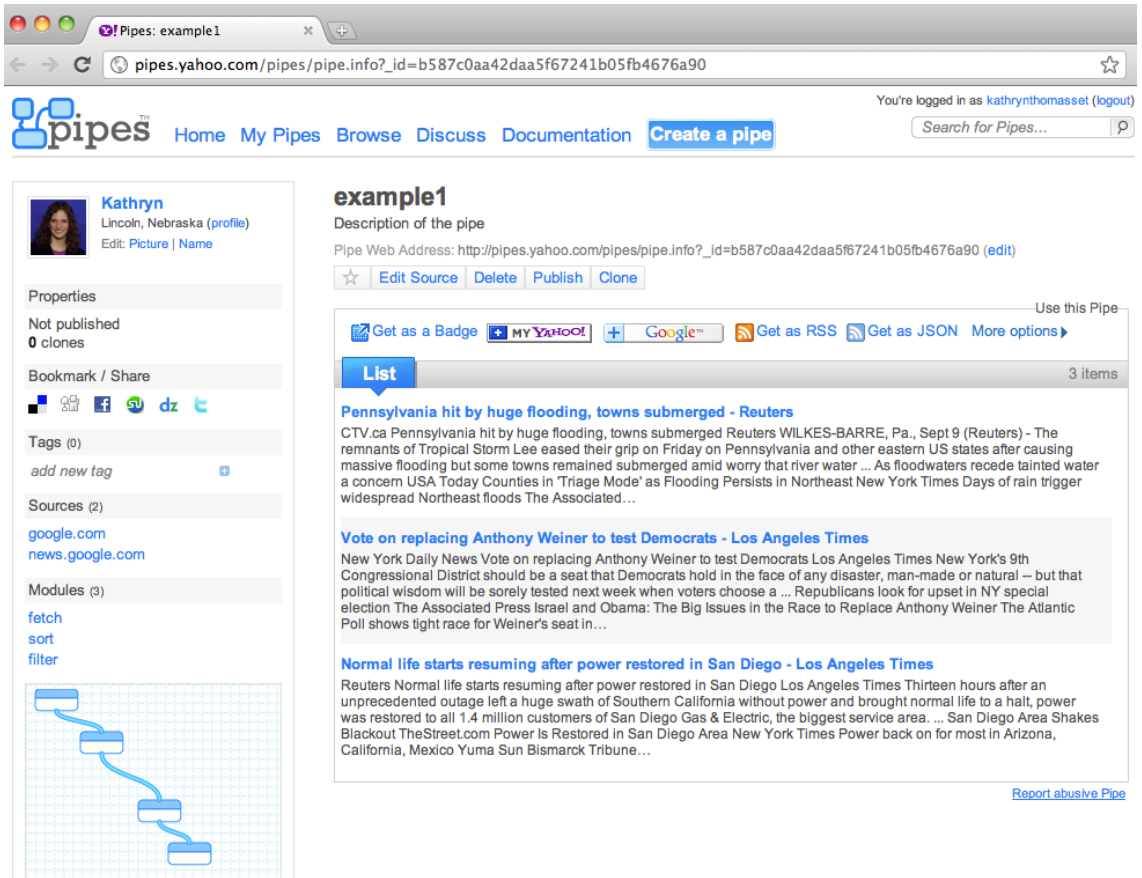
Web Mashups

Applications that compose and manipulate existing data sources or services to create new data or service.

Why study mashups?

- Many environments (e.g., Apatar, DERI Pipes, IBM Mashup Center, Kivati, Yahoo! Pipes, . . .)
- Potential impact (many users, growth)

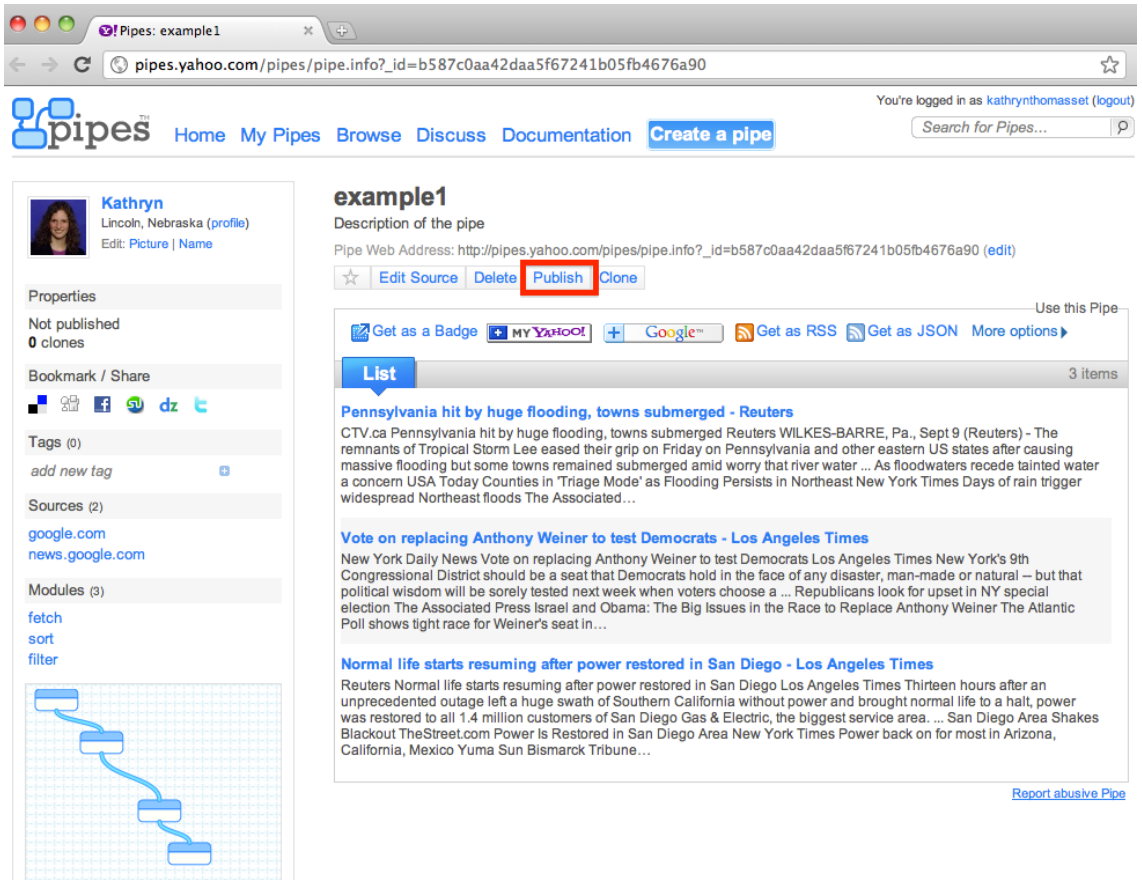
About Yahoo! Pipes



This example mashup fetches and filters news from `news.google.com`

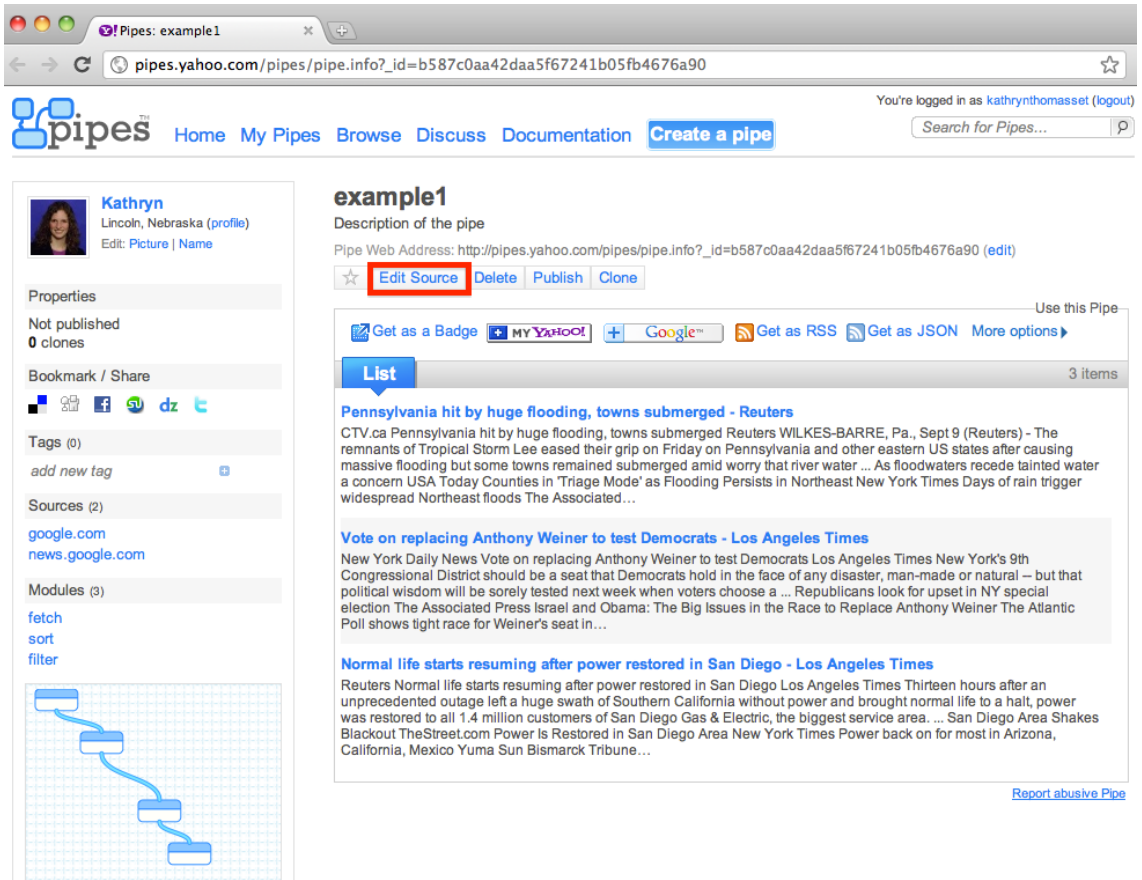
Information page shows the pipe output and descriptive information

About Yahoo! Pipes



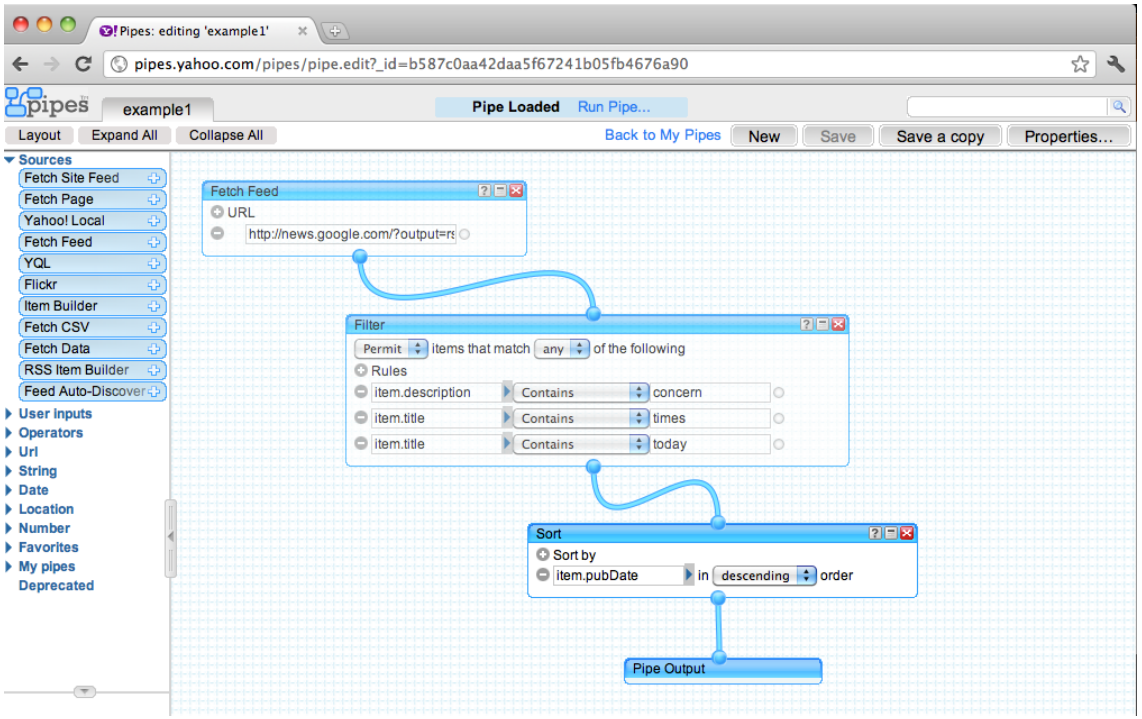
Clicking Publish adds the pipe to the public repository

About Yahoo! Pipes



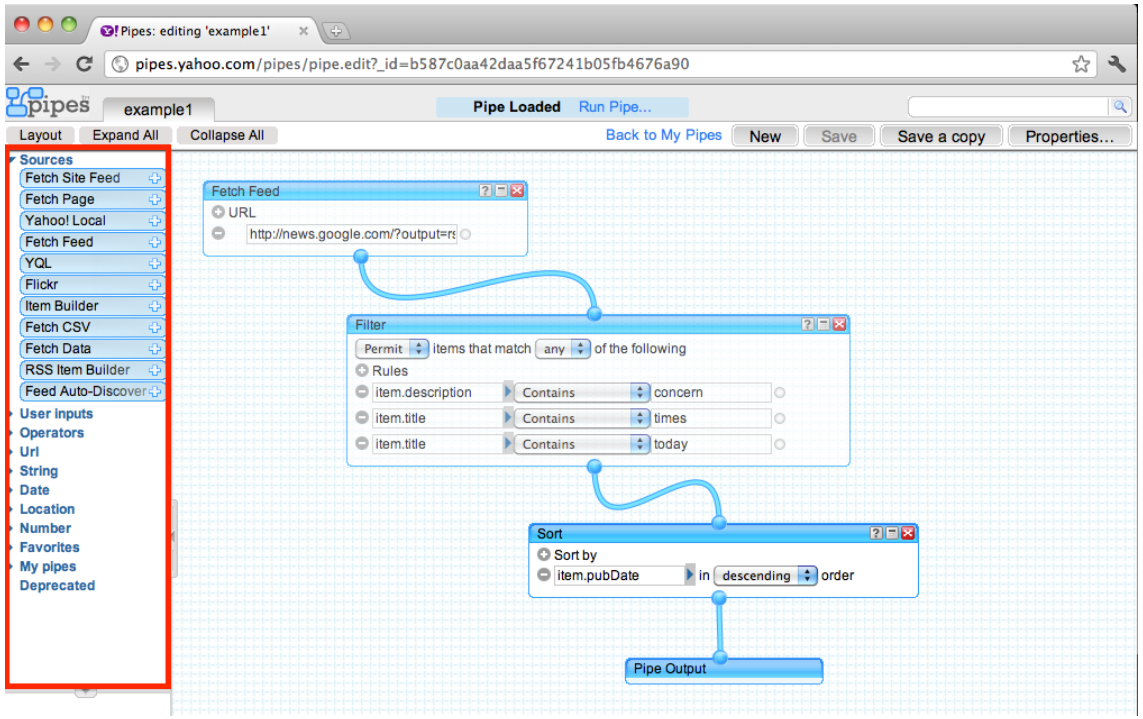
Clicking Edit Source loads the Pipes Editor

About Yahoo! Pipes



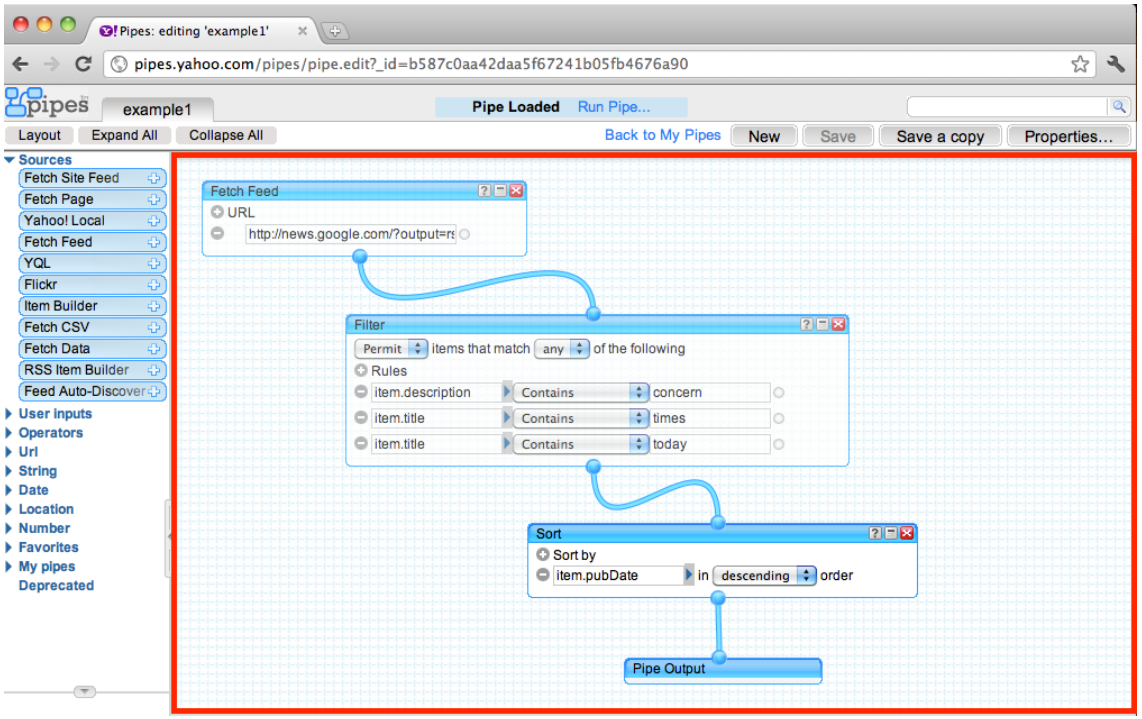
- Visual mashup creation environment
- Within a browser
- Drag and drop interface

About Yahoo! Pipes



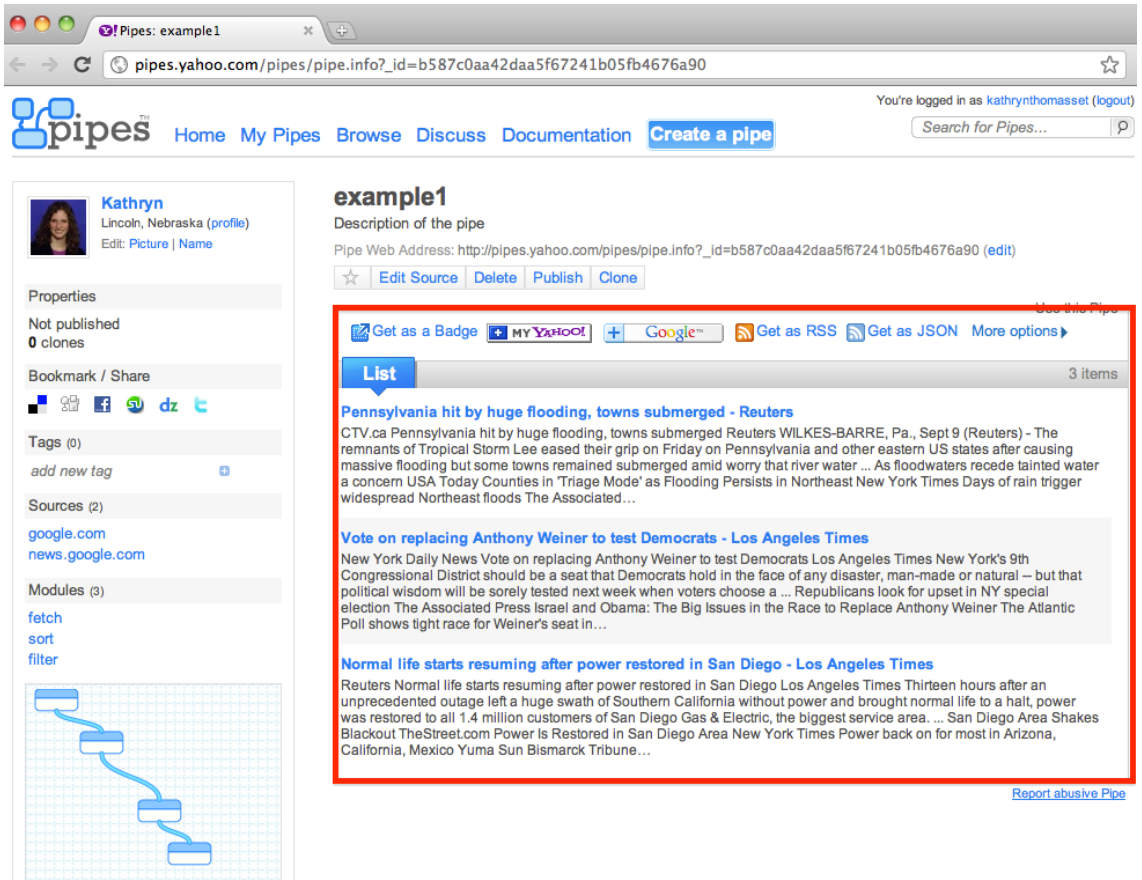
- Visual mashup creation environment
- Within a browser
- Drag and drop interface

About Yahoo! Pipes



- Visual mashup creation environment
- Within a browser
- Drag and drop interface

About Yahoo! Pipes



- Visual mashup creation environment
- Within a browser
- Drag and drop interface

Empirical Study Details

Goal: To better understand *end-user programmer* communities

- Research Goal
- Study Context
- **Research Questions**
- Variables and Metrics
- Methods
- Results

Research Questions

RQ1: What are the characteristics of Yahoo! Pipes community?

1a,b: author attrition and author contributions

1c: artifact sharing, abstraction, complexity, and degree of overlap among pipes in the repository

Research Questions

RQ1: What are the characteristics of Yahoo! Pipes community?

- 1a,b: author attrition and author contributions
- 1c: artifact sharing, abstraction, complexity, and degree of overlap among pipes in the repository

RQ2: How do pipe attributes change as authors gain experience?

- 2a: experience measured by time
- 2b: experience measured by total contributions

Research Questions

RQ1: What are the characteristics of Yahoo! Pipes community?

- 1a,b: author attrition and author contributions
- 1c: artifact sharing, abstraction, complexity, and degree of overlap among pipes in the repository

RQ2: How do pipe attributes change as authors gain experience?

- 2a: experience measured by time
- 2b: experience measured by total contributions

RQ3: What are the characteristics of most prolific authors?

- 3a: author activities
- 3b: author skills
- 3c: awareness of the community

Research Questions

RQ1: What are the characteristics of Yahoo! Pipes community?

- 1a,b: author attrition and author contributions
- 1c: artifact sharing, abstraction, complexity, and degree of overlap among pipes in the repository

RQ2: How do pipe attributes change as authors gain experience?

- 2a: experience measured by time
- 2b: experience measured by total contributions

RQ3: What are the characteristics of most prolific authors?

- 3a: author activities
- 3b: author skills
- 3c: awareness of the community

Empirical Study Details

Goal: To better understand *end-user programmer* communities

- Research Goal
- Study Context
- Research Questions
- **Variables and Metrics**
- Methods
- Results

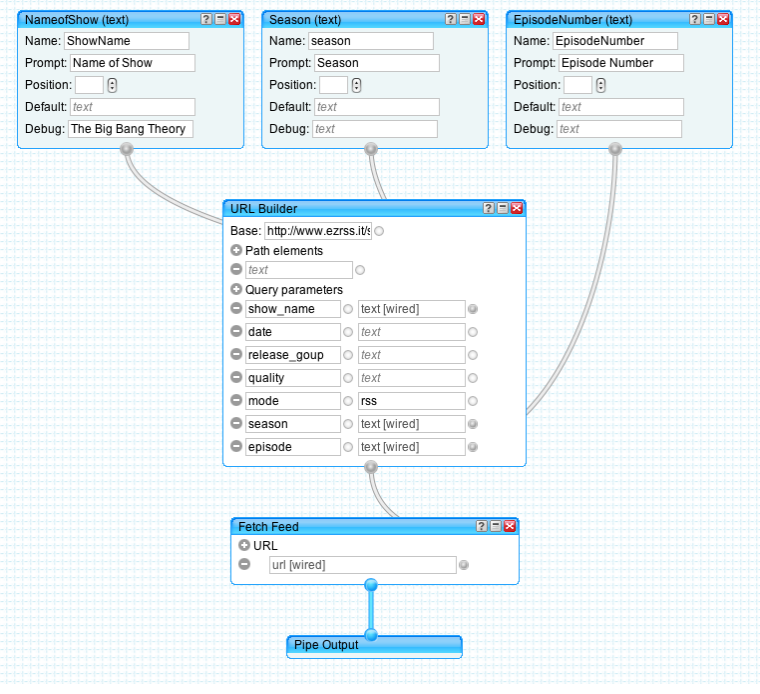
Study Details

Concept to Capture	Variable
artifact sharing/impact	popularity
abstraction	configurability
complexity	size
overlap of artifacts in repository	diversity

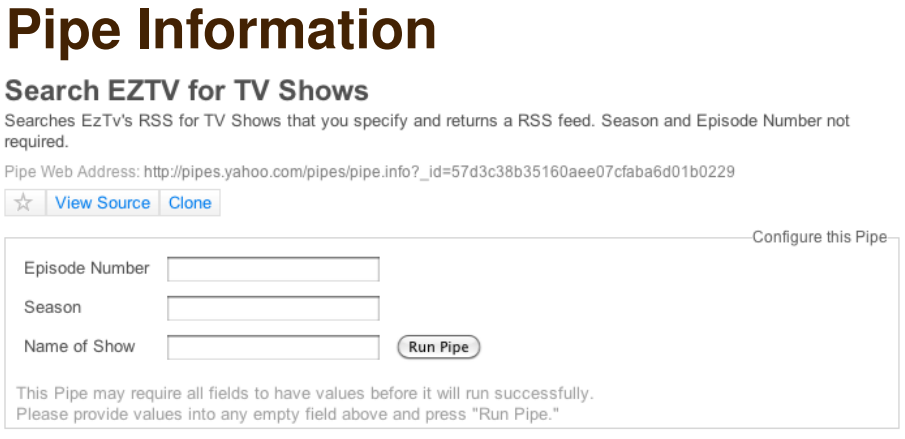
Study Details

Variables: size, configurability, popularity, diversity

Pipe Source



Pipe Information



Study Details

Variables: **size**, configurability, popularity, diversity

Pipe Source

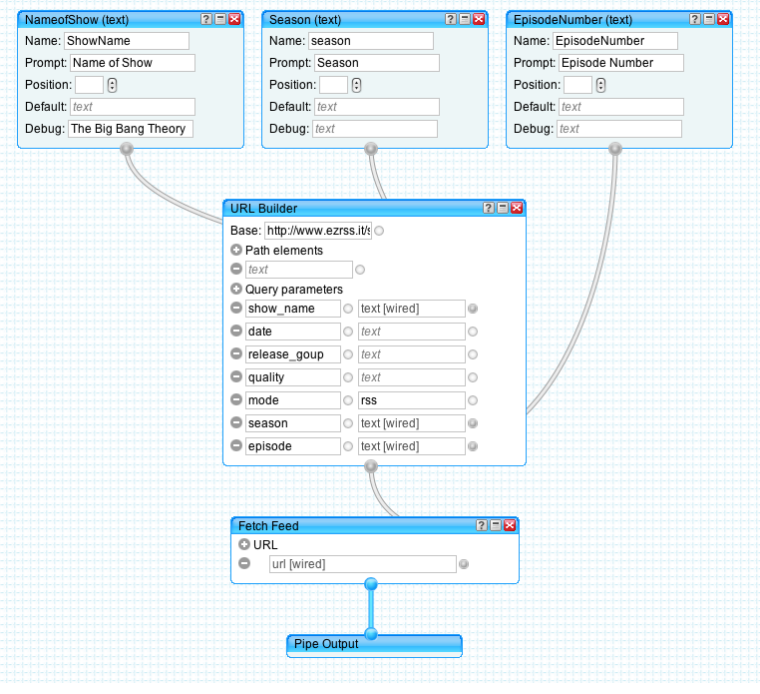
6 modules

Significance: Size is related to complexity

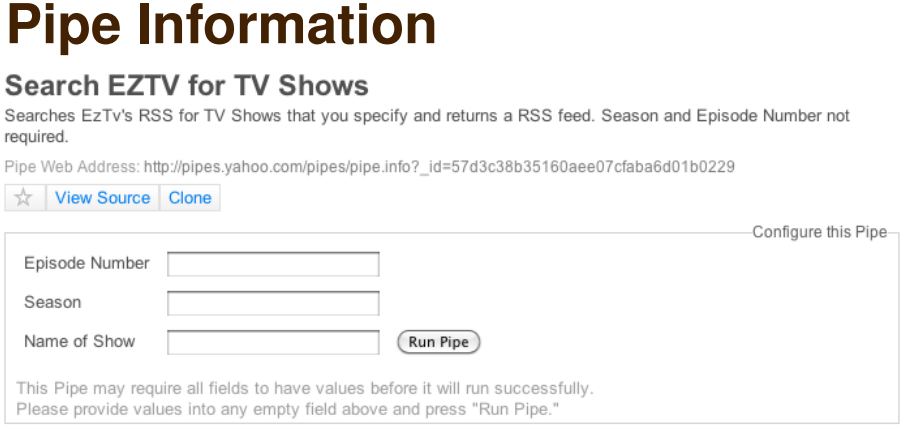
Study Details

Variables: size, **configurability**, popularity, diversity

Pipe Source



Pipe Information



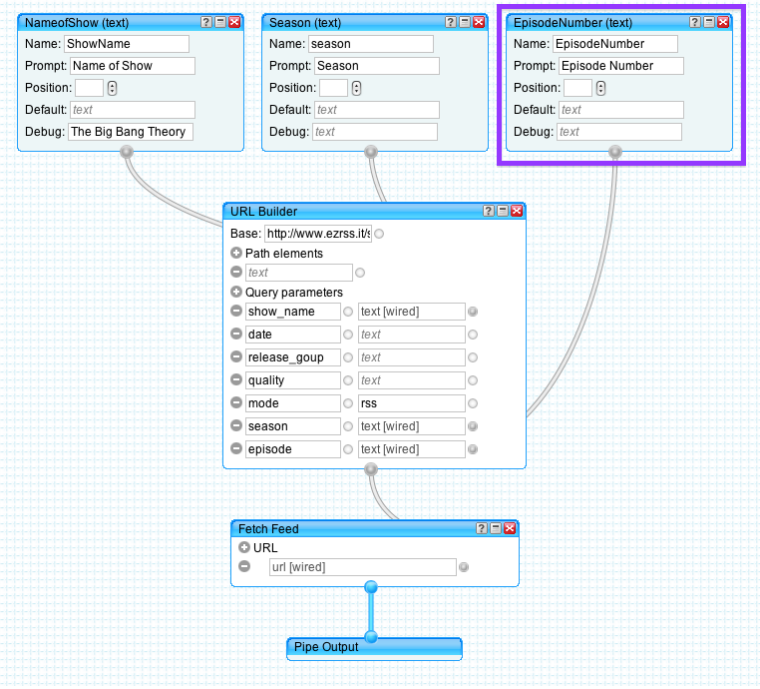
3 modules

Significance: Configurability is related to abstraction and language mastery

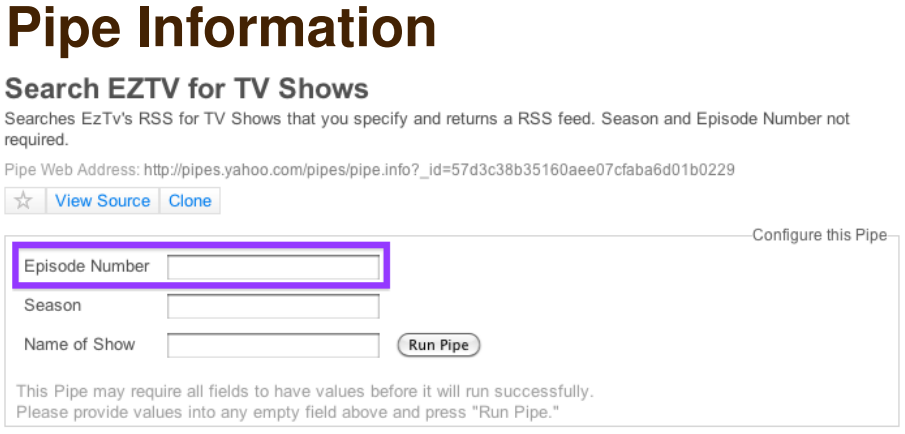
Study Details

Variables: size, **configurability**, popularity, diversity

Pipe Source



Pipe Information



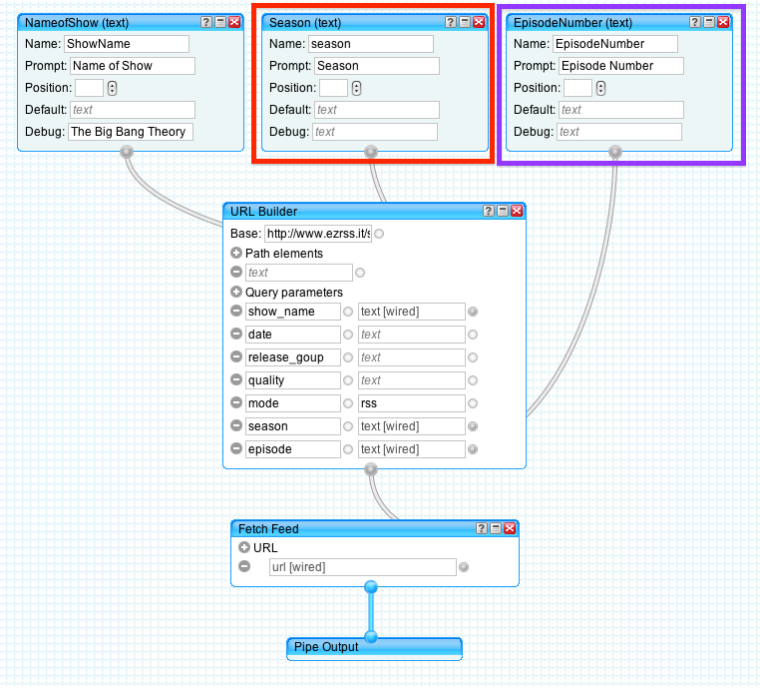
3 modules

Significance: Configurability is related to abstraction and language mastery

Study Details

Variables: size, **configurability**, popularity, diversity

Pipe Source



Pipe Information

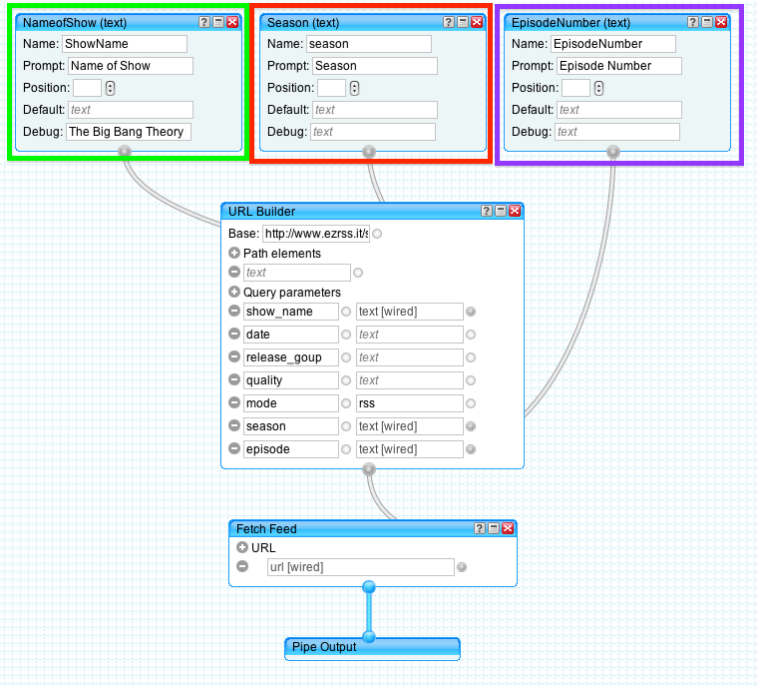
3 modules

Significance: Configurability is related to abstraction and language mastery

Study Details

Variables: size, **configurability**, popularity, diversity

Pipe Source



Pipe Information

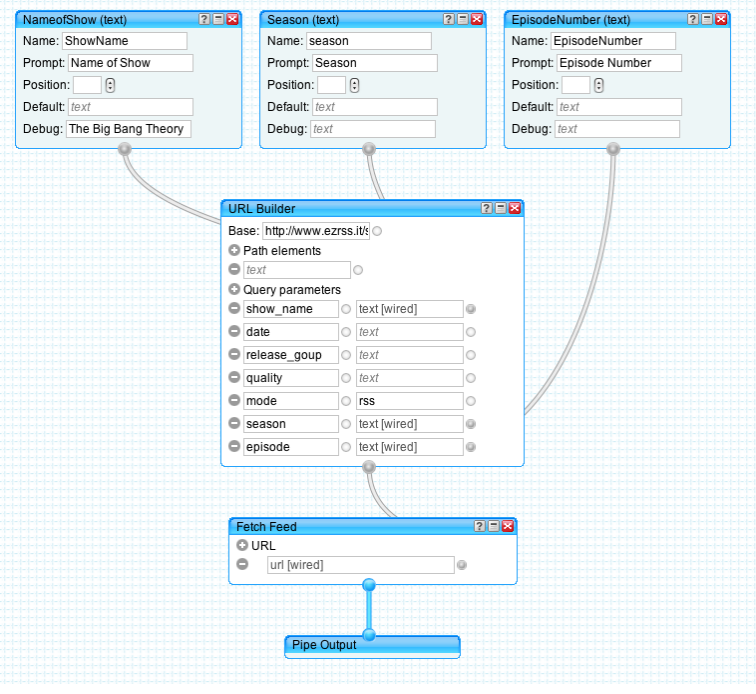
3 modules

Significance: Configurability is related to abstraction and language mastery

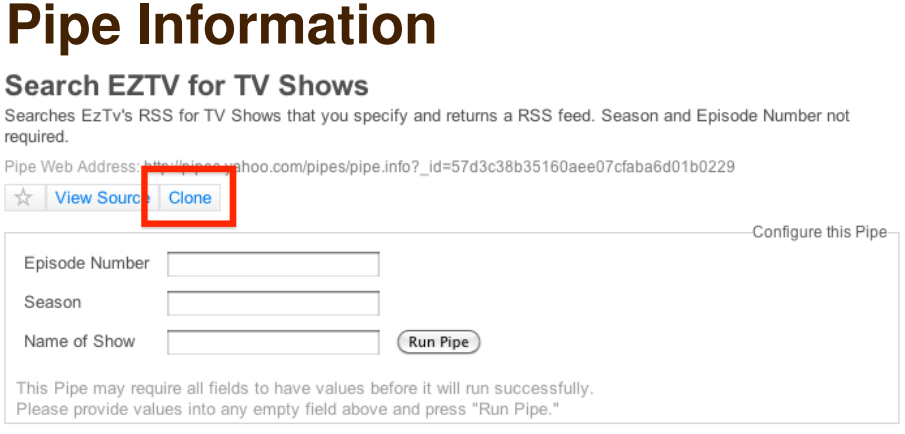
Study Details

Variables: size, configurability, **popularity**, diversity

Pipe Source



Pipe Information



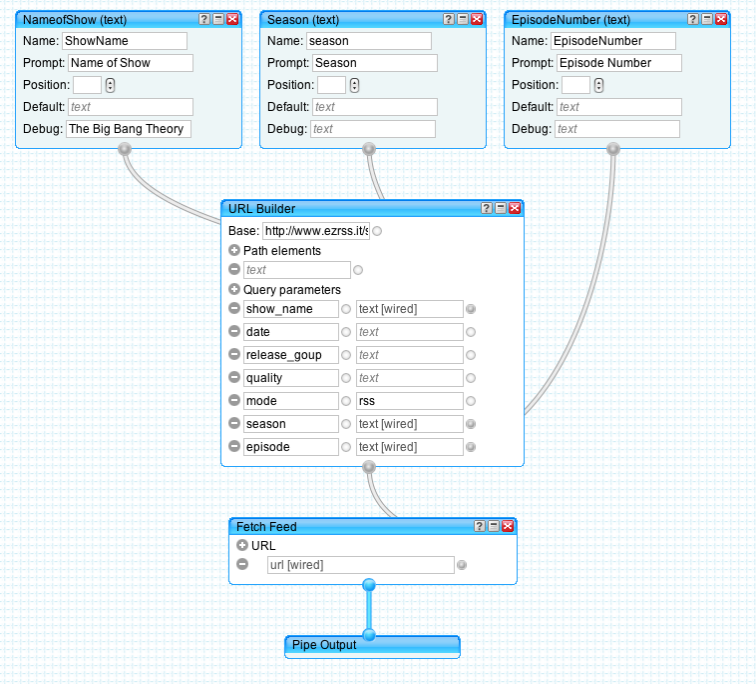
190 clones

Significance: Popularity is related to impact on community

Study Details

Variables: size, configurability, popularity, **diversity**

Pipe Source



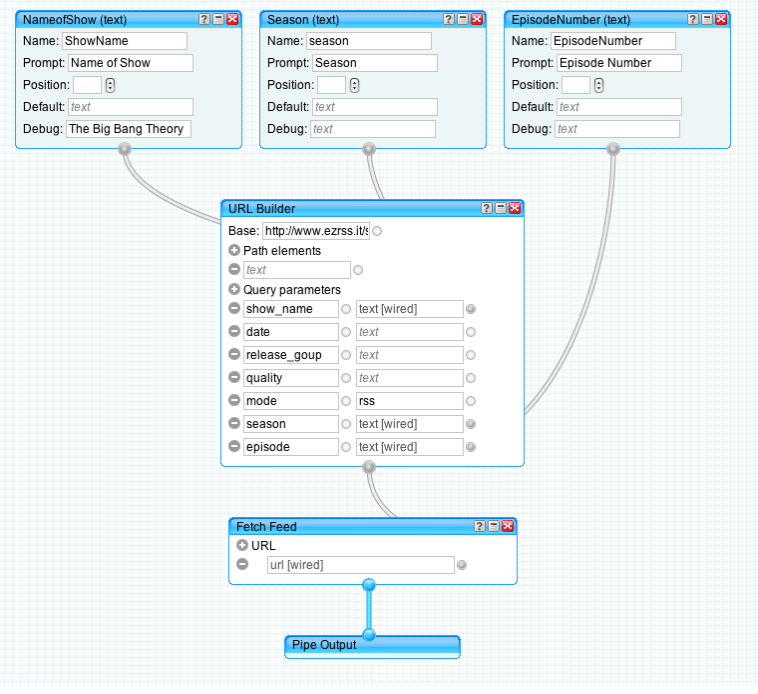
1. Same structure, fields, content
2. Same structure, field counts
3. Same structure
4. Same bag of modules
5. Same set of modules
6. Same type bag
7. Same size
8. No match

Significance: Diversity is related to contribution novelty

Study Details

Variables: size, configurability, popularity, **diversity**

Pipe Source



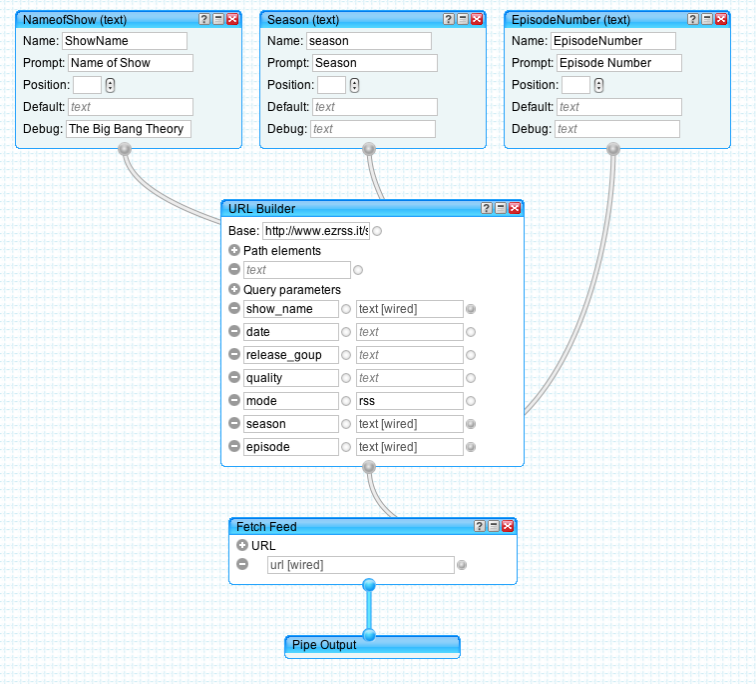
1 Same structure, fields, content
2 Same structure, field counts
3 **Same structure**
4 Same bag of modules
5 Same set of modules
6 Same type bag
7 Same size
8 No match

Significance: Diversity is related to contribution novelty

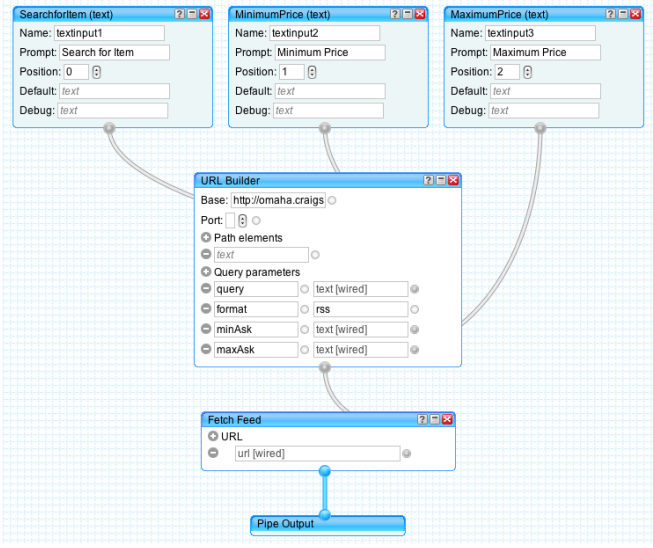
Study Details

Variables: size, configurability, popularity, **diversity**

Pipe Source



3 Same structure

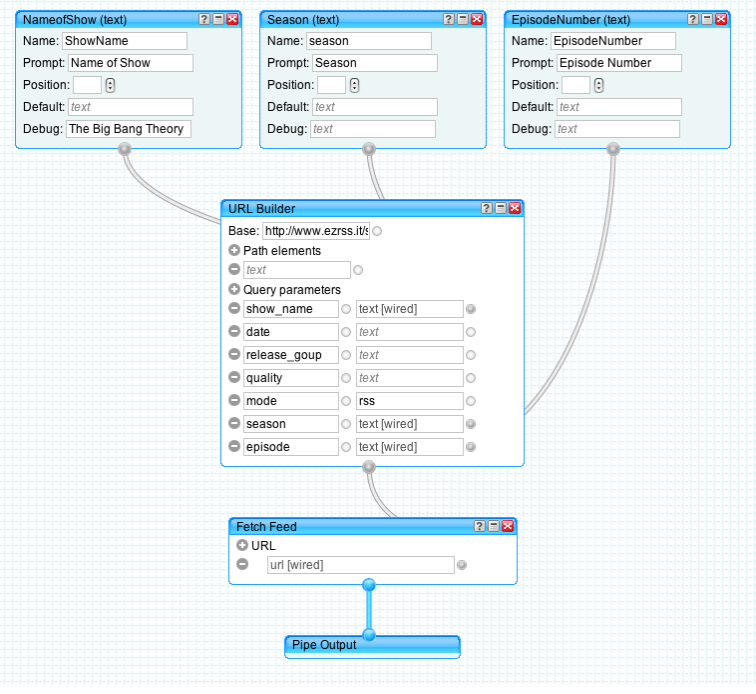


Significance: Diversity is related to contribution novelty

Study Details

Variables: size, configurability, popularity, **diversity**

Pipe Source



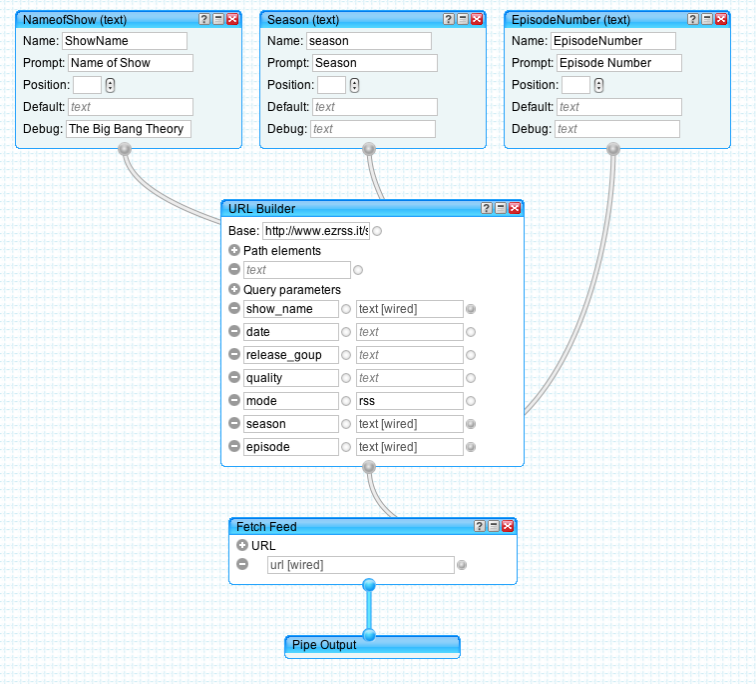
1. Same structure, fields, content
2. Same structure, field counts
3. Same structure
4. Same bag of modules
5. Same set of modules
6. Same type bag
7. Same size
8. No match

Significance: Diversity is related to contribution novelty

Study Details

Variables: size, configurability, popularity, **diversity**

Pipe Source



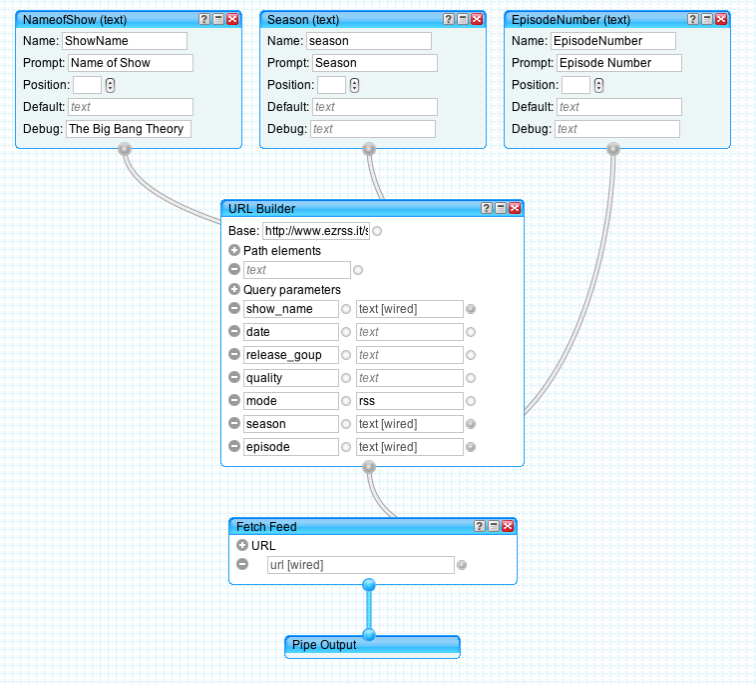
1. Same structure, fields, content
2. Same structure, field counts
3. Same structure
4. Same bag of modules
5. **Same set of modules**
6. Same type bag
7. Same size
8. No match

Significance: Diversity is related to contribution novelty

Study Details

Variables: size, configurability, popularity, **diversity**

Pipe Source



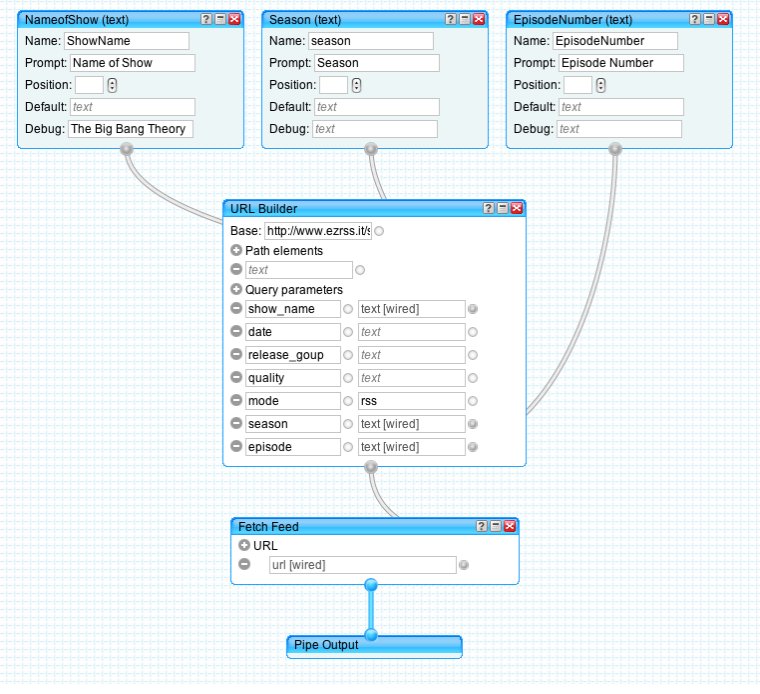
5 Same set of modules

Significance: Diversity is related to contribution novelty

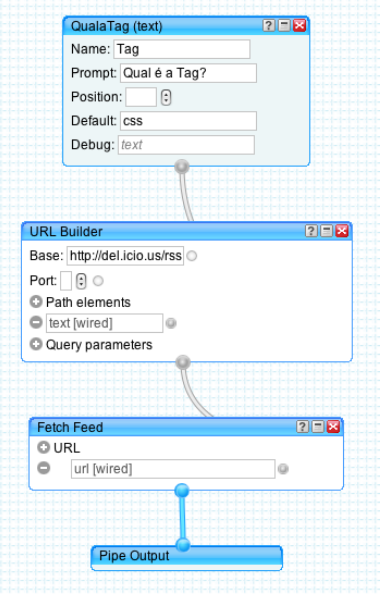
Study Details

Variables: size, configurability, popularity, **diversity**

Pipe Source



5 Same set of modules

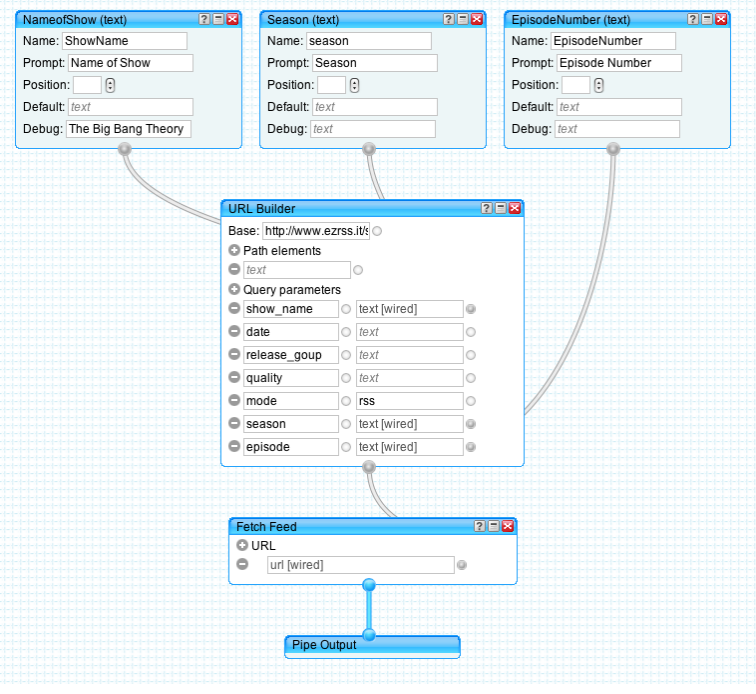


Significance: Diversity is related to contribution novelty

Study Details

Variables: size, configurability, popularity, **diversity**

Pipe Source



1. Same structure, fields, content
2. Same structure, field counts
3. Same structure
4. Same bag of modules
5. Same set of modules
6. Same type bag
7. Same size
8. No match

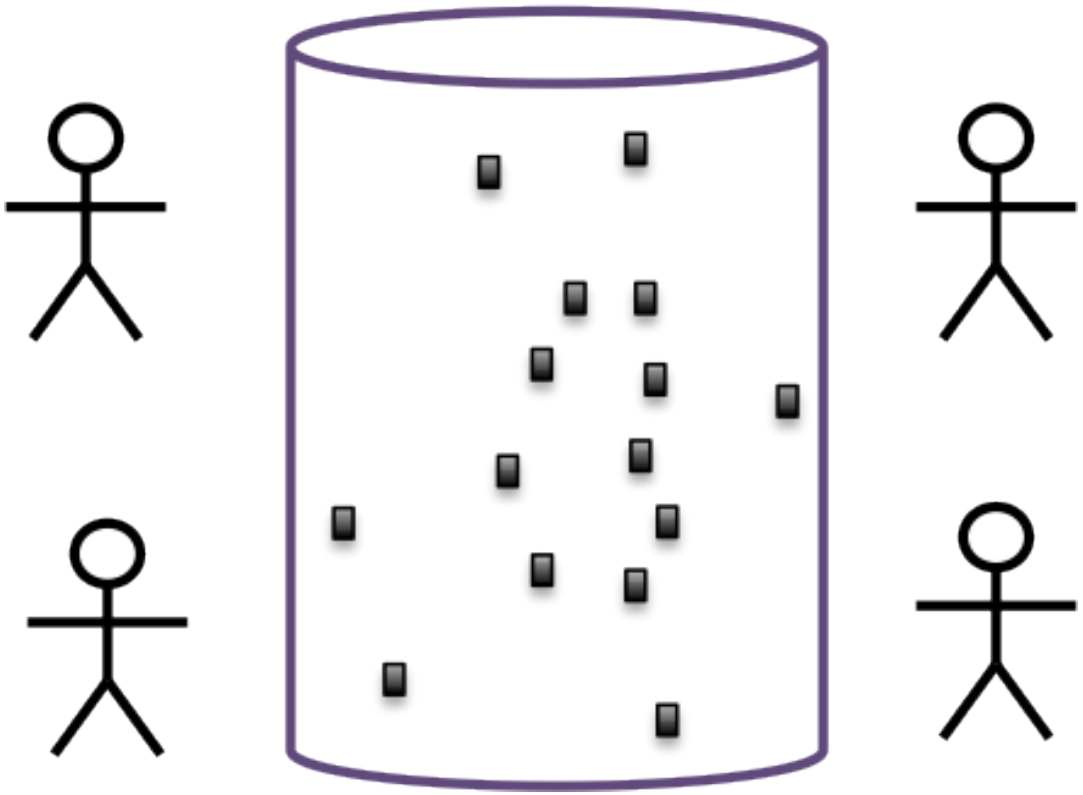
Significance: Diversity is related to contribution novelty

Empirical Study Details

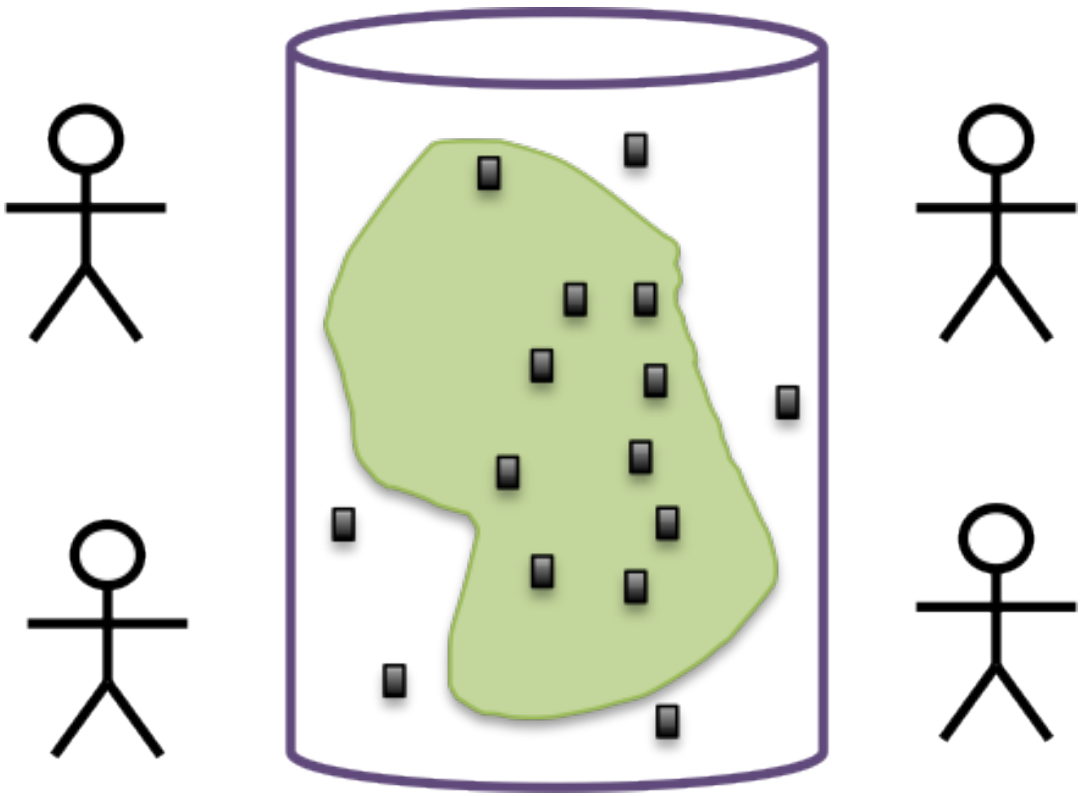
Goal: To better understand *end-user programmer* communities

- Research Goal
- Study Context
- Research Questions
- Variables and Metrics
- **Methods**
- Results

Data Collection

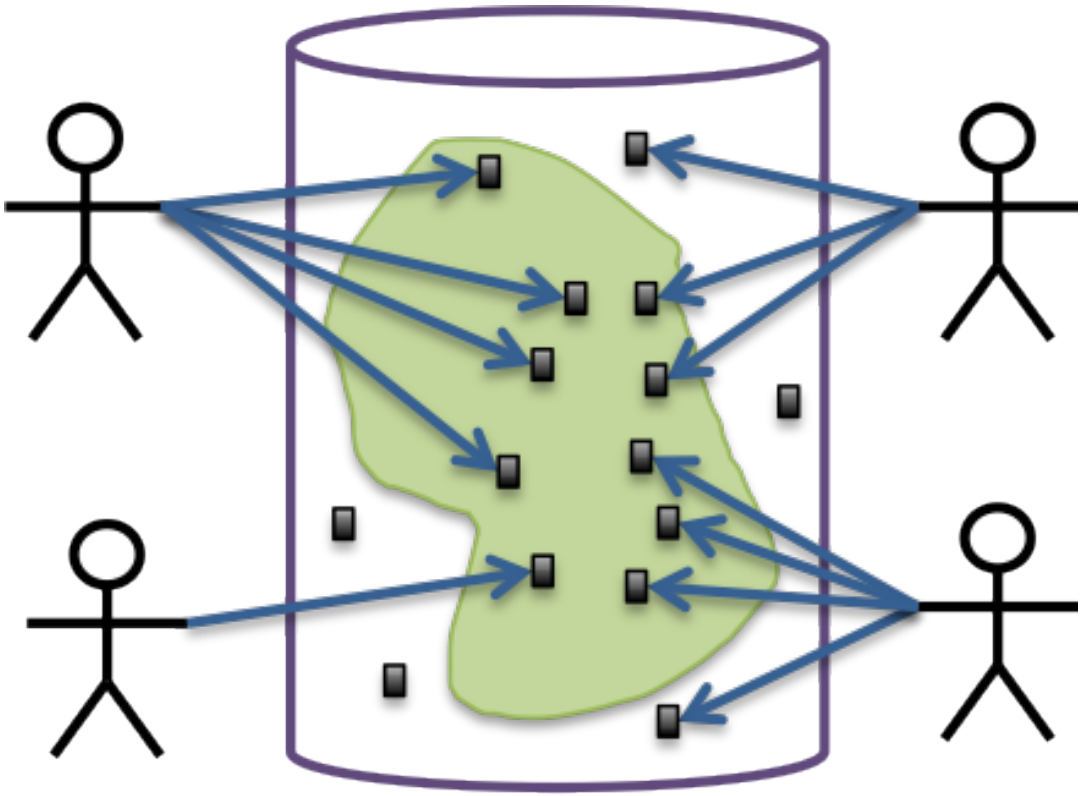


Data Collection



Artifacts: 32,887

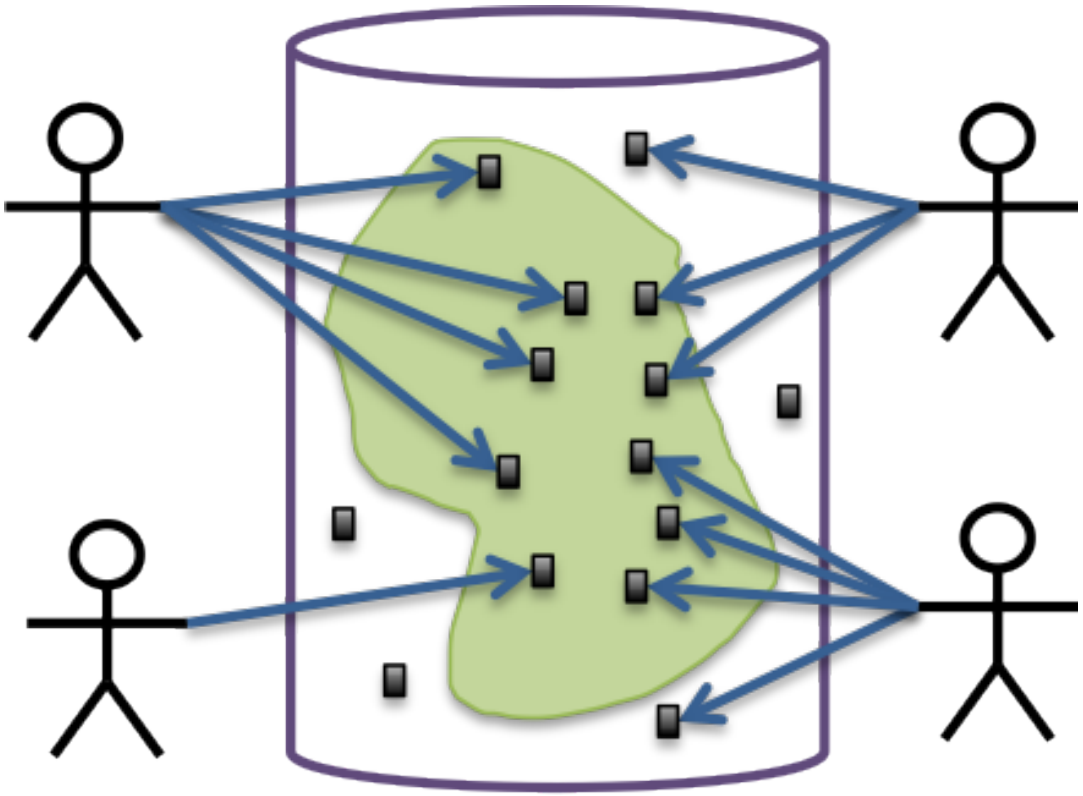
Data Collection



Artifacts: 32,887

Authors: 20,313

Data Collection



Artifacts: 32,887

Authors: 20,313

Threats: public repository offers limited visibility (internal); sampling bias (external); generalizability to other domains (external)

Empirical Study Details

Goal: To better understand *end-user programmer* communities

- Research Goal
- Study Context
- Research Questions
- Variables and Metrics
- Methods
- **Results**

Research Questions

RQ1: What are the characteristics of Yahoo! Pipes community?

1a,b: author attrition and author contributions

1c: artifact sharing, abstraction, complexity, and degree of overlap among pipes in the repository

RQ1: Characteristics of Yahoo! Pipes Community

Summary

Metric	Average
Size	8.20 modules per pipe
Configurability	0.65 modules per pipe
Popularity	5.67 clones per pipe
Diversity	3.62 cluster level

RQ1: Characteristics of Yahoo! Pipes Community

Summary

Metric	Average
Size	8.20 modules per pipe
Configurability	**0.65 modules per pipe**
Popularity	5.67 clones per pipe
Diversity	3.62 cluster level

34% of pipes are configurable

RQ1: Characteristics of Yahoo! Pipes Community

Summary

Metric	Average
Size	8.20 modules per pipe
Configurability	0.65 modules per pipe
Popularity	**5.67 clones per pipe**
Diversity	3.62 cluster level

54% of pipes have been cloned

RQ1: Characteristics of Yahoo! Pipes Community

Summary

Metric	Average
Size	8.20 modules per pipe
Configurability	0.65 modules per pipe
Popularity	5.67 clones per pipe
Diversity	**3.62 cluster level**

5% of pipes are exact duplicates, yet 46% have a match if field values are relaxed

RQ1: Characteristics of Yahoo! Pipes Community

Take Aways:

- There is a lot of reuse of shared pipes
- Participants often submit pipes that are highly similar to other pipes in the repository

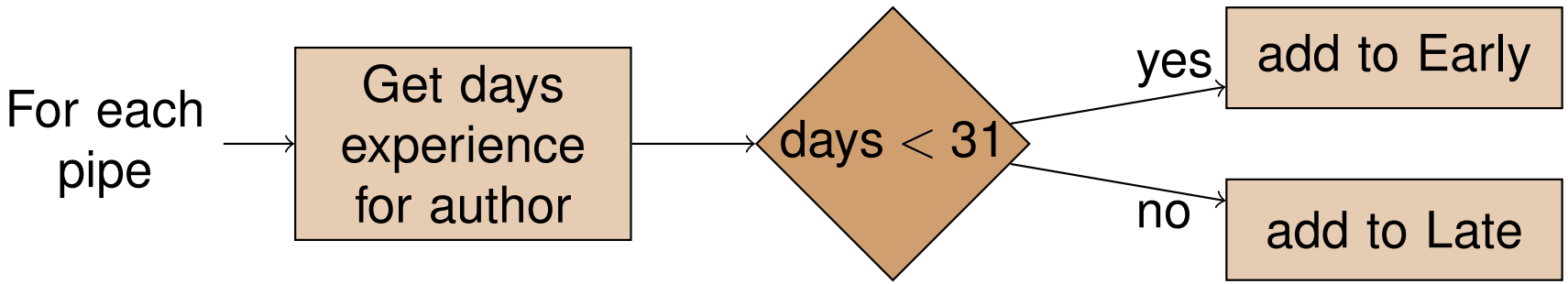
Research Questions

RQ2: How do pipe attributes change as authors gain experience?

- 2a: measures experience in terms of time
- 2b: measures experience in terms of total contributions

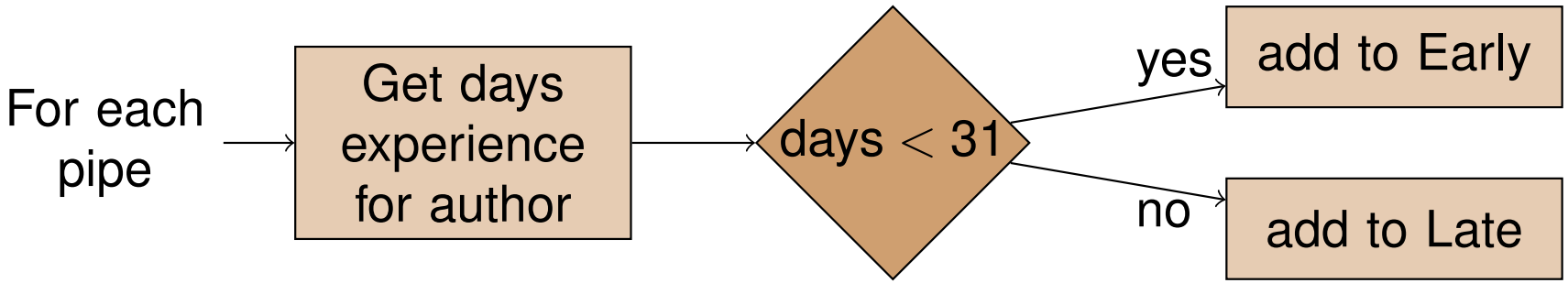
RQ2: Analysis of artifacts as authors gain experience

Comparisons based on experience (time)



RQ2: Analysis of artifacts as authors gain experience

Comparisons based on experience (time)



Characteristic	μ_{early}	μ_{late}
# of Pipes	27,555	5,332
Diversity***	3.519	4.126
Popularity***	4.984	9.254
Configurability***	0.614	0.838
Size***	7.919	9.587

$H_0 : \mu_{early} > \mu_{late}$
$H_a : \mu_{early} \leq \mu_{late}$

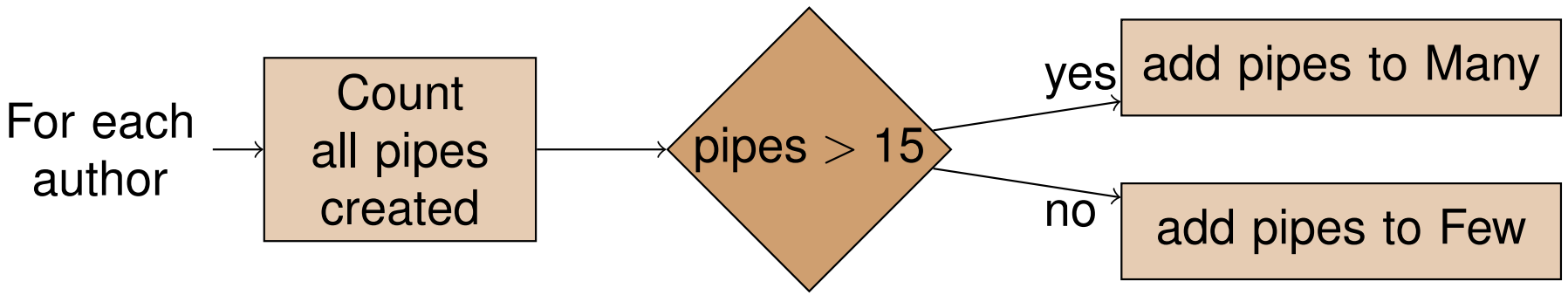
Signif. codes:
*** 0.001 ** 0.01

RQ2: Analysis of artifacts as authors gain experience

Take Away: More experience results in pipes that are larger, more popular, more configurable, and more diverse

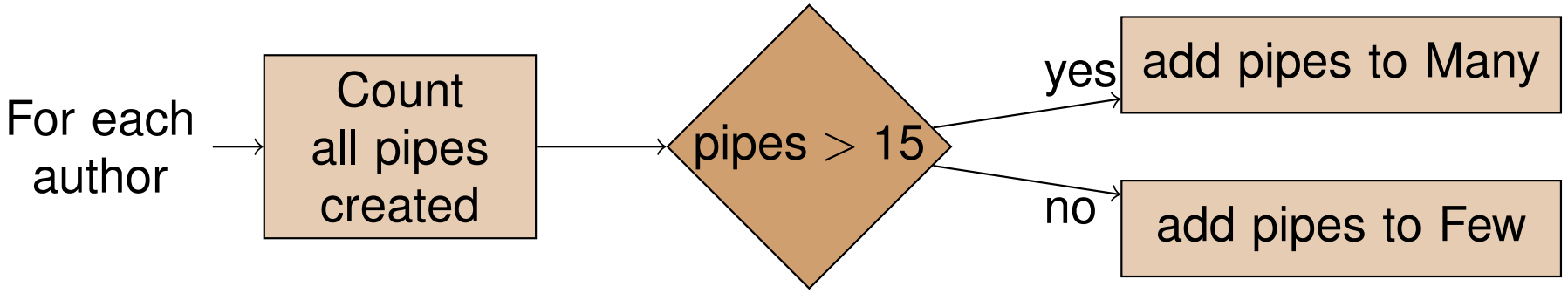
RQ2: Analysis of artifacts as authors gain experience

Comparisons based on contributions



RQ2: Analysis of artifacts as authors gain experience

Comparisons based on contributions



Characteristic	μ_{few}	μ_{many}
# of Pipes	30,503	2,384
Diversity	3.639	3.355
Popularity***	4.302	23.250
Configurability***	0.644	0.729
Size**	8.194	8.136

$H_0 : \mu_{few} > \mu_{many}$
$H_a : \mu_{few} \leq \mu_{many}$

Signif. codes:
*** 0.001 ** 0.01

RQ2: Analysis of artifacts as authors gain experience

Take Away: The most prolific authors create pipes that are larger, more popular, and more configurable

. . . what about diversity?

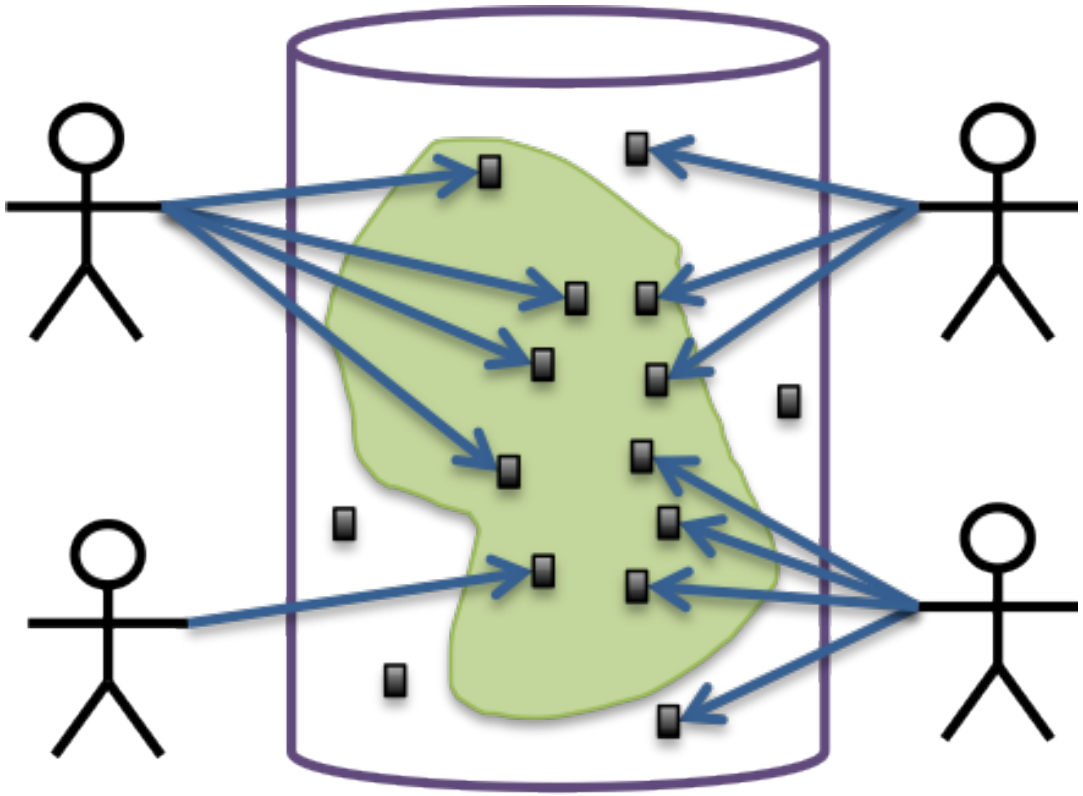
Research Questions

RQ3: What are the characteristics of most prolific authors?

- 3a: author activities
- 3b: author skills
- 3c: awareness of the community

RQ3: Characteristics of most prolific authors

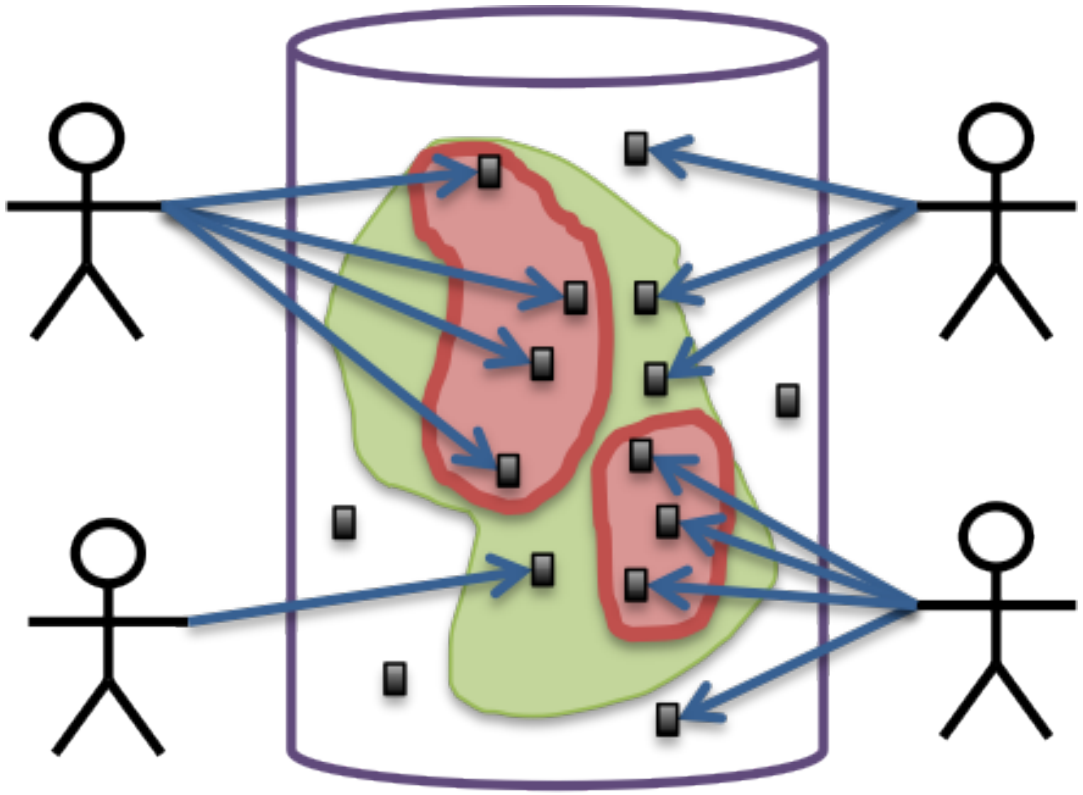
Study Set-up



Authors: 20,313

RQ3: Characteristics of most prolific authors

Study Set-up

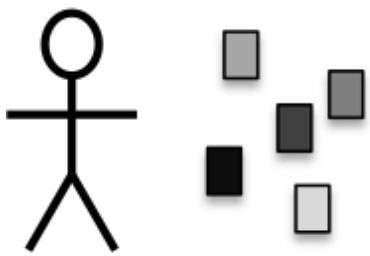


Authors: 20,313

Prolific Authors: 81

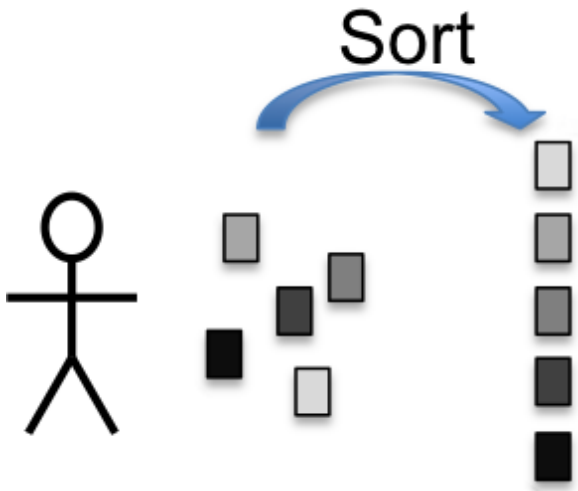
RQ3: Characteristics of most prolific authors

Rolling Cluster Analysis



RQ3: Characteristics of most prolific authors

Rolling Cluster Analysis

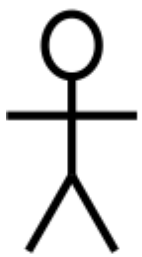


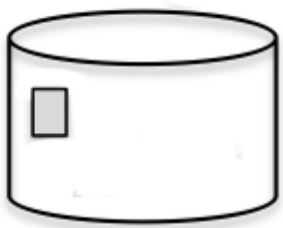
RQ3: Characteristics of most prolific authors

Rolling Cluster Analysis

RQ3: Characteristics of most prolific authors

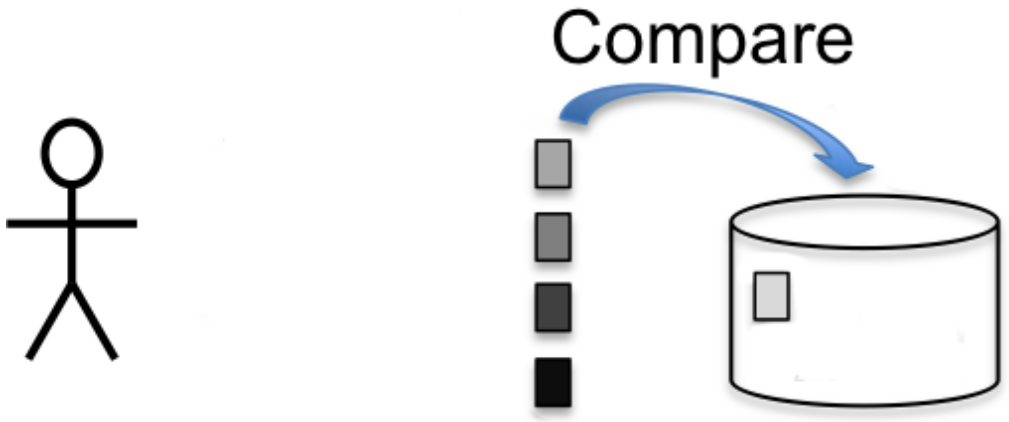
Rolling Cluster Analysis





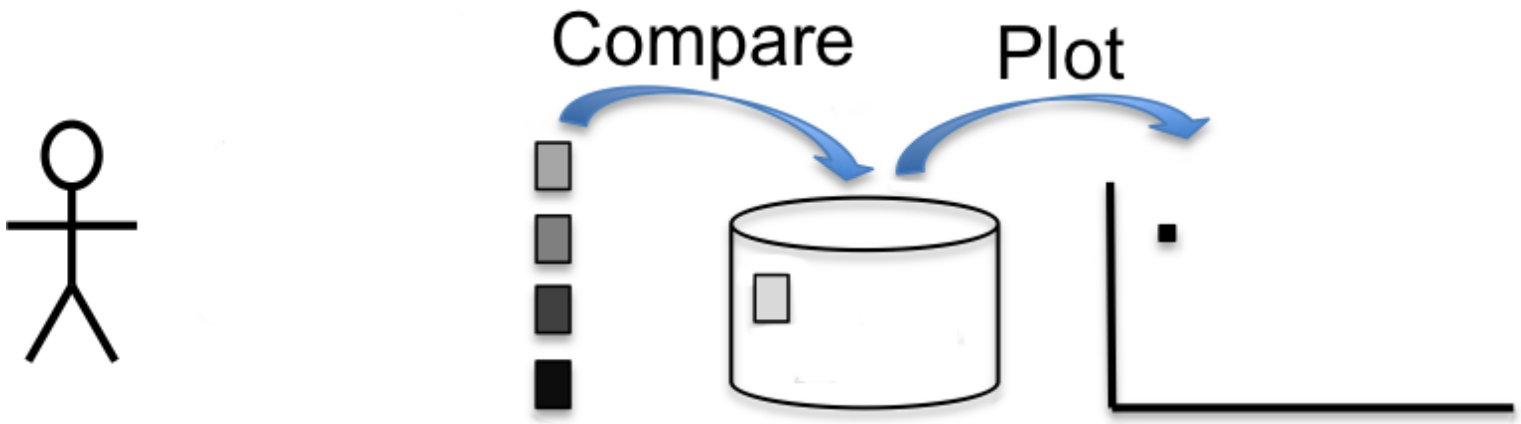
RQ3: Characteristics of most prolific authors

Rolling Cluster Analysis



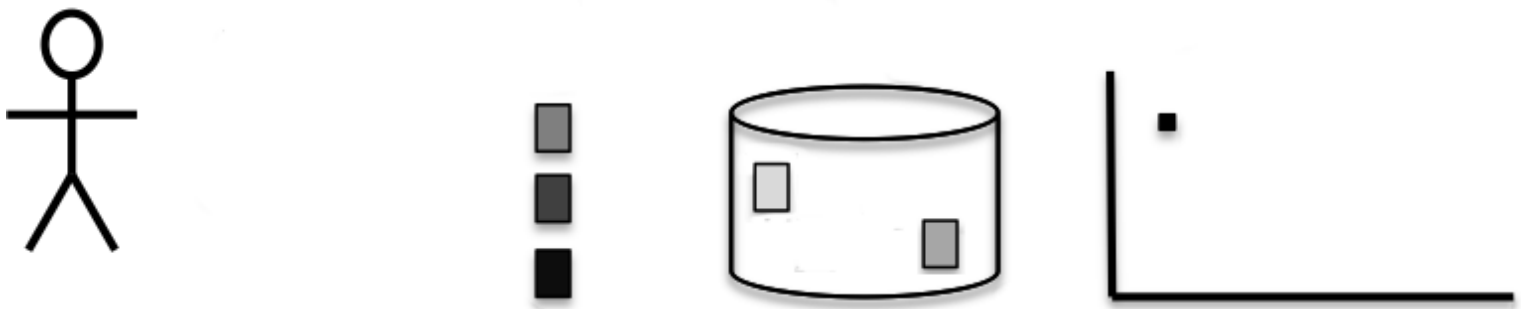
RQ3: Characteristics of most prolific authors

Rolling Cluster Analysis



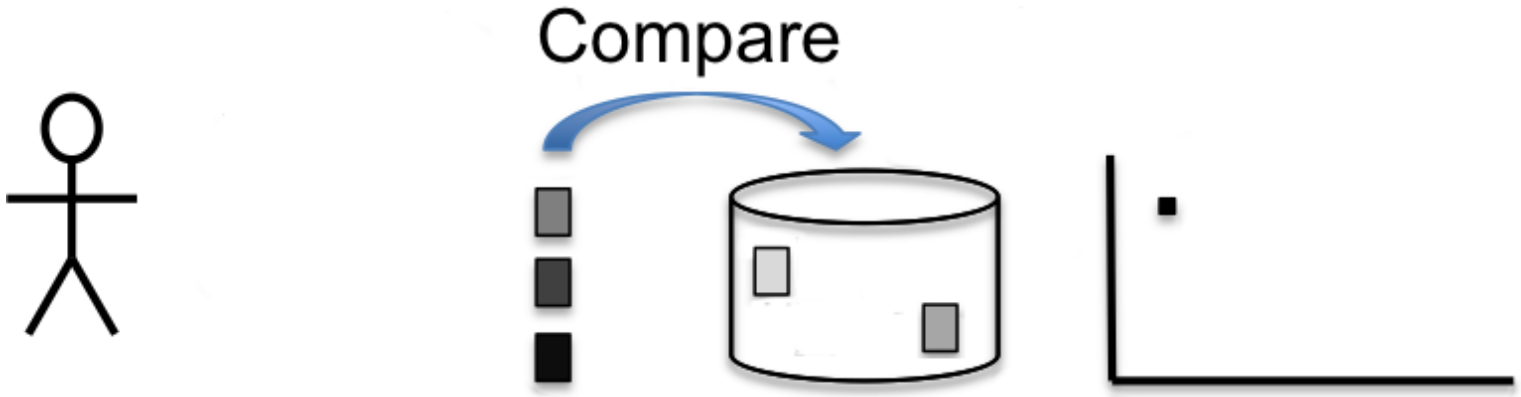
RQ3: Characteristics of most prolific authors

Rolling Cluster Analysis



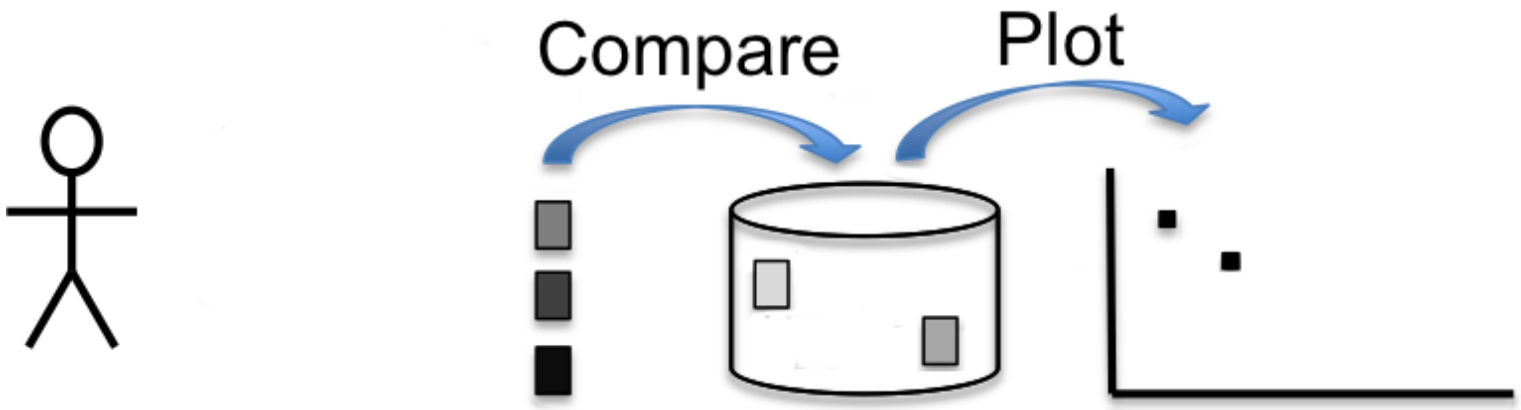
RQ3: Characteristics of most prolific authors

Rolling Cluster Analysis



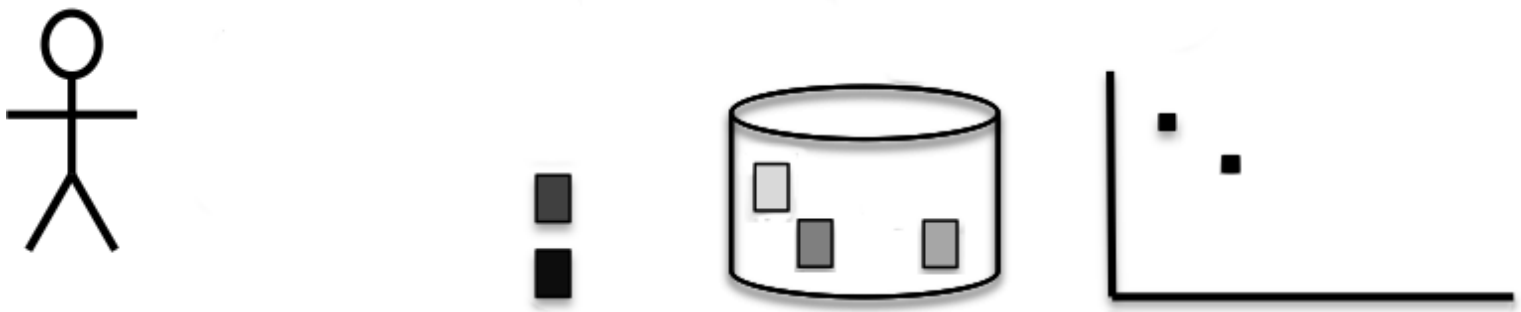
RQ3: Characteristics of most prolific authors

Rolling Cluster Analysis



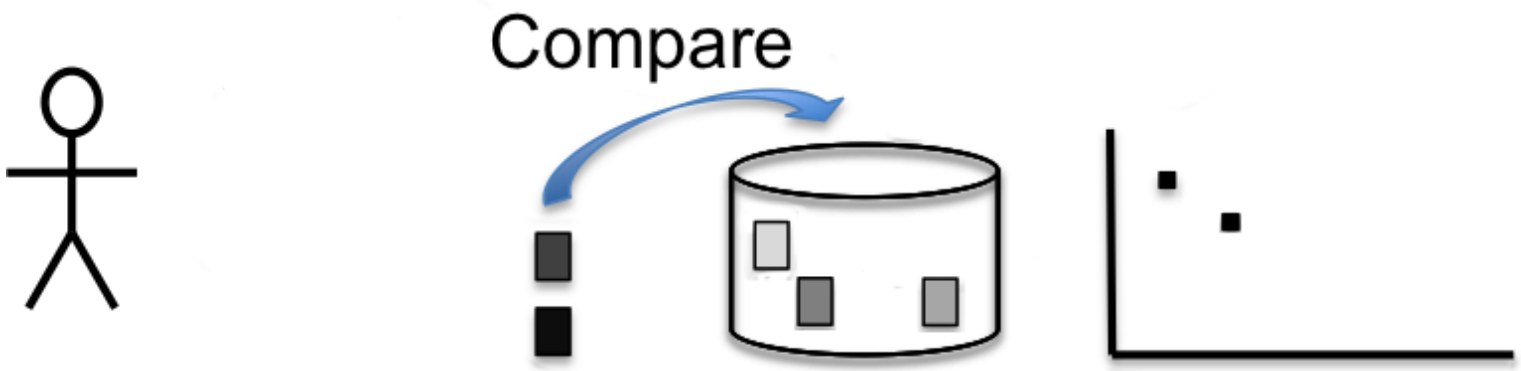
RQ3: Characteristics of most prolific authors

Rolling Cluster Analysis



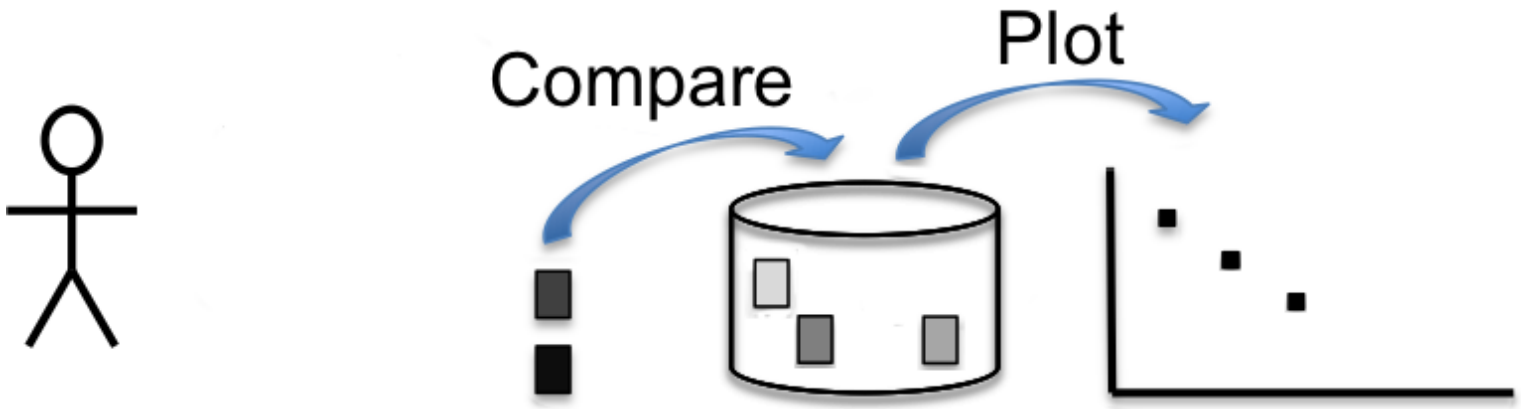
RQ3: Characteristics of most prolific authors

Rolling Cluster Analysis



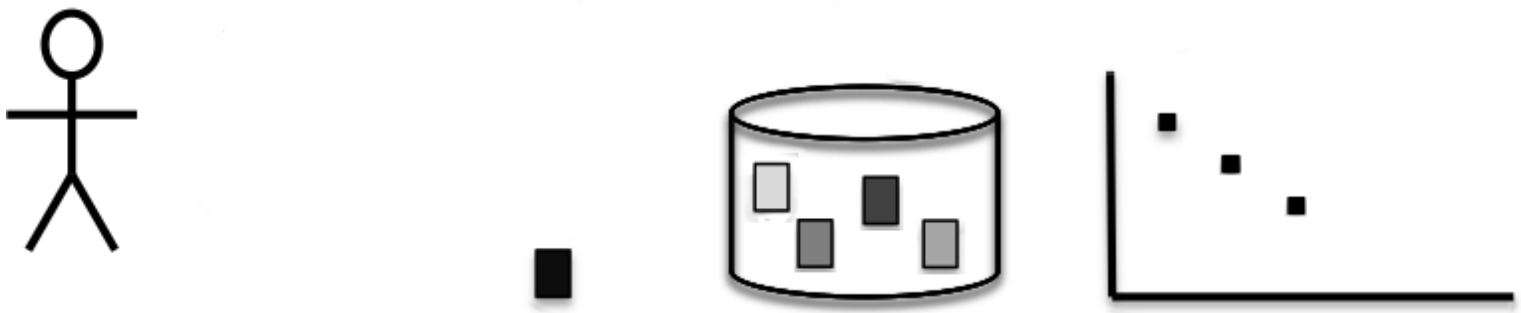
RQ3: Characteristics of most prolific authors

Rolling Cluster Analysis



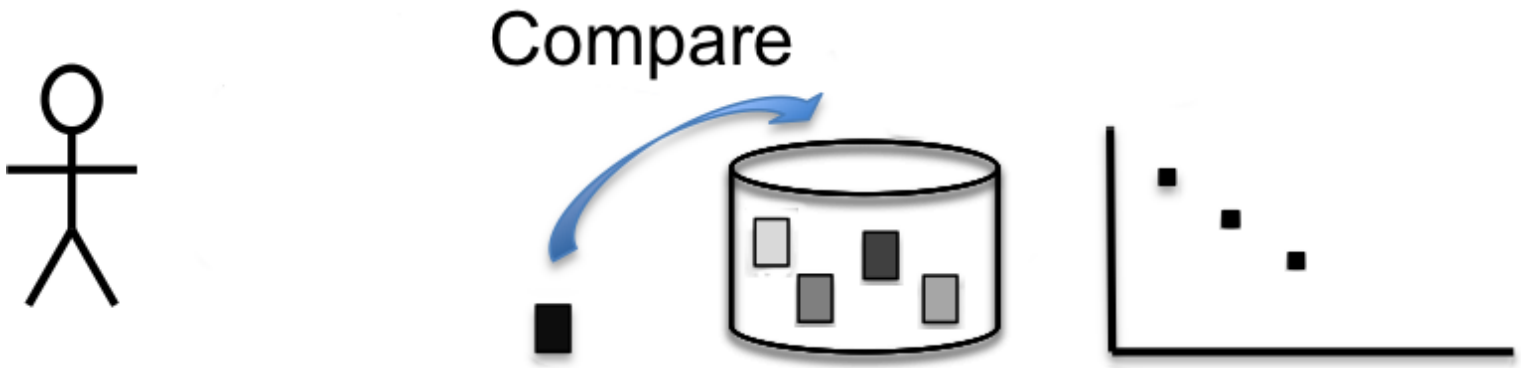
RQ3: Characteristics of most prolific authors

Rolling Cluster Analysis



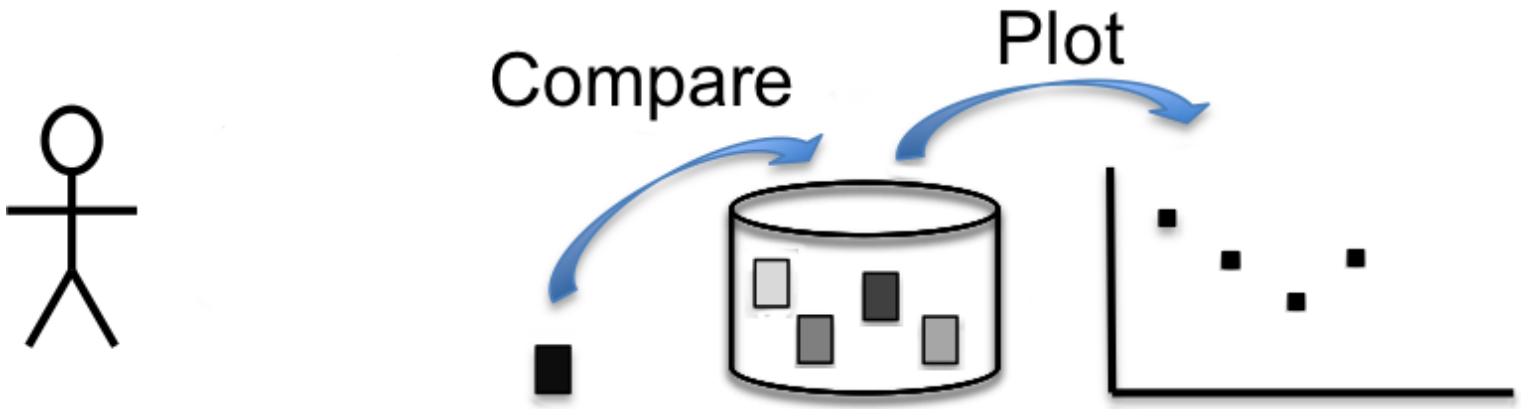
RQ3: Characteristics of most prolific authors

Rolling Cluster Analysis



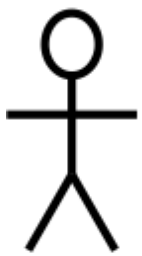
RQ3: Characteristics of most prolific authors

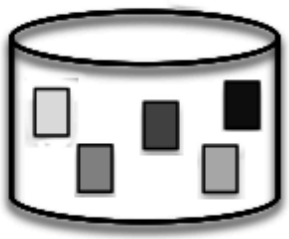
Rolling Cluster Analysis



RQ3: Characteristics of most prolific authors

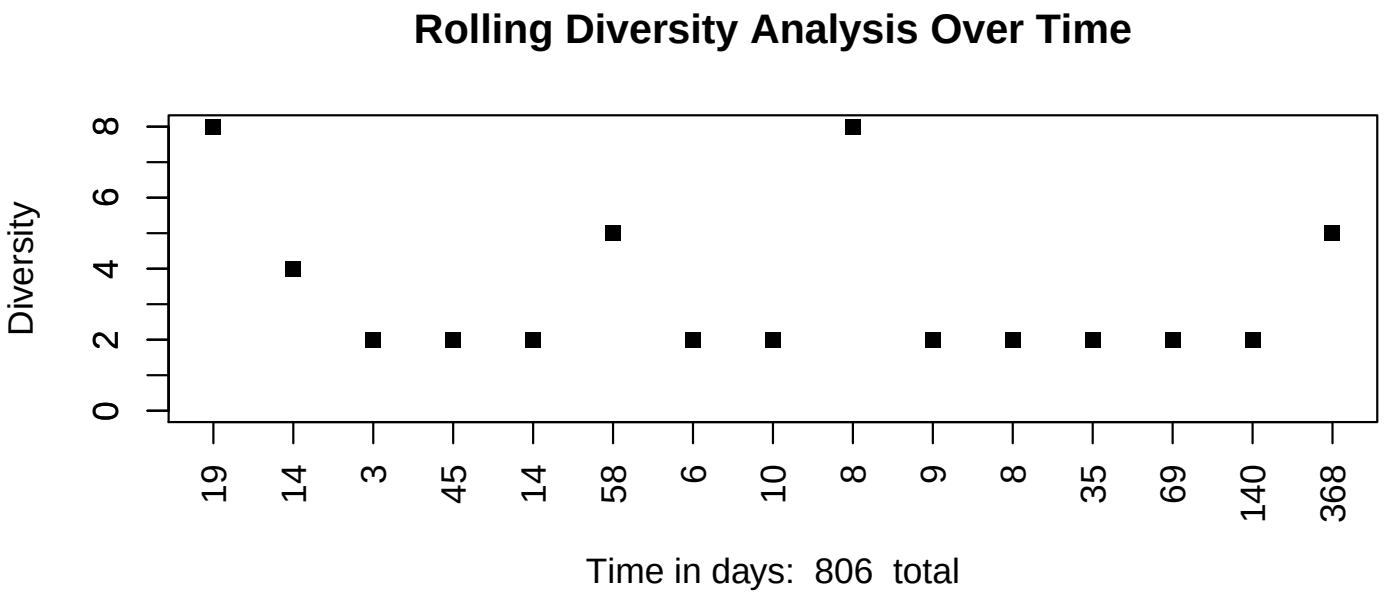
Rolling Cluster Analysis





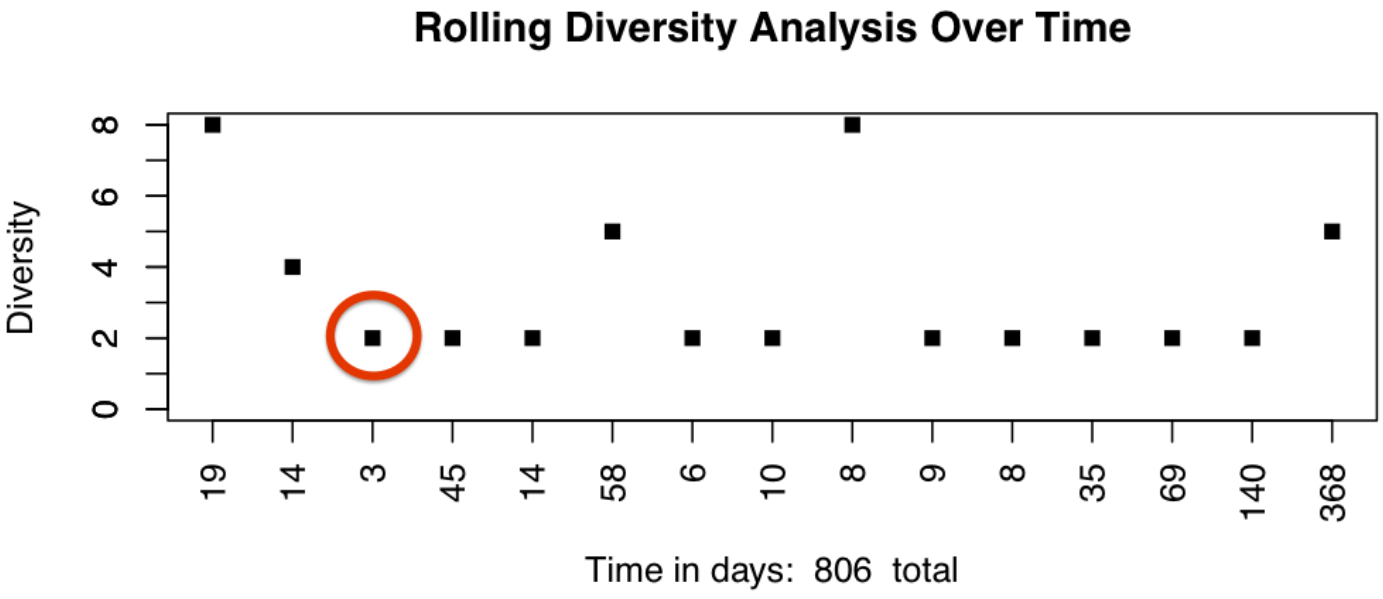
RQ3: Characteristics of most prolific authors

Author Activities



RQ3: Characteristics of most prolific authors

Author Activities



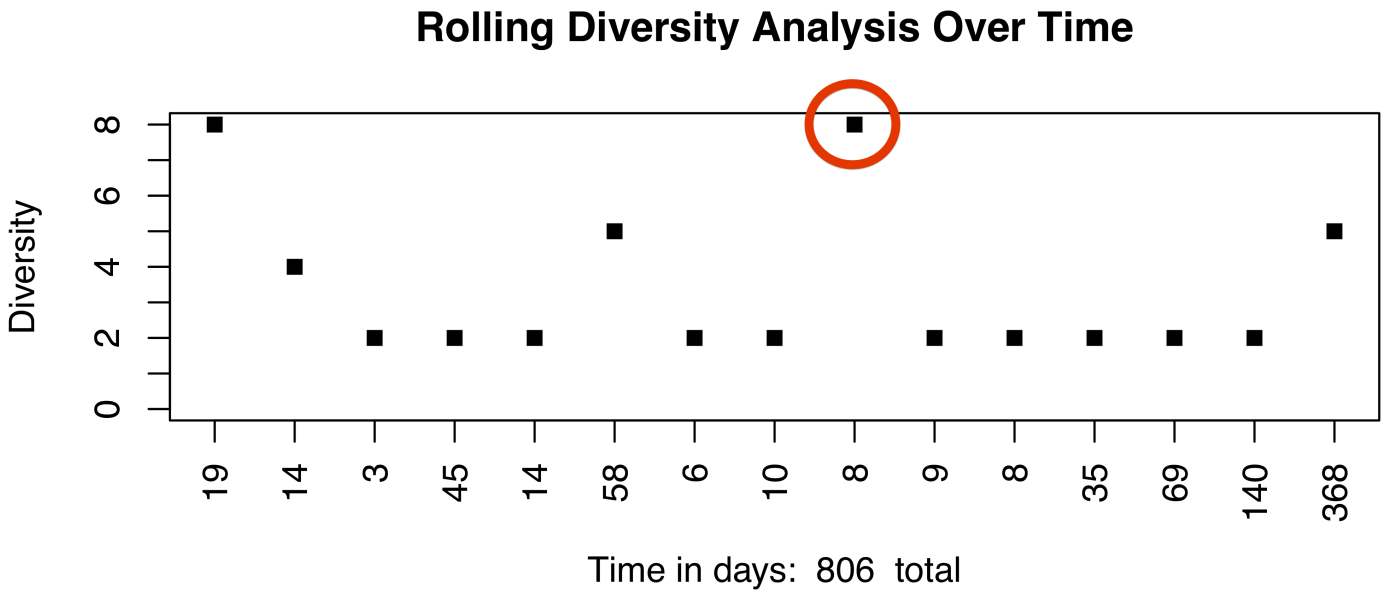
Level 2: Same structure and field counts; relax field values

43% of pipes submitted by prolific authors represent tweaks

For Example: Change a URL, filter criterion, sort order, . . .

RQ3: Characteristics of most prolific authors

Author Activities

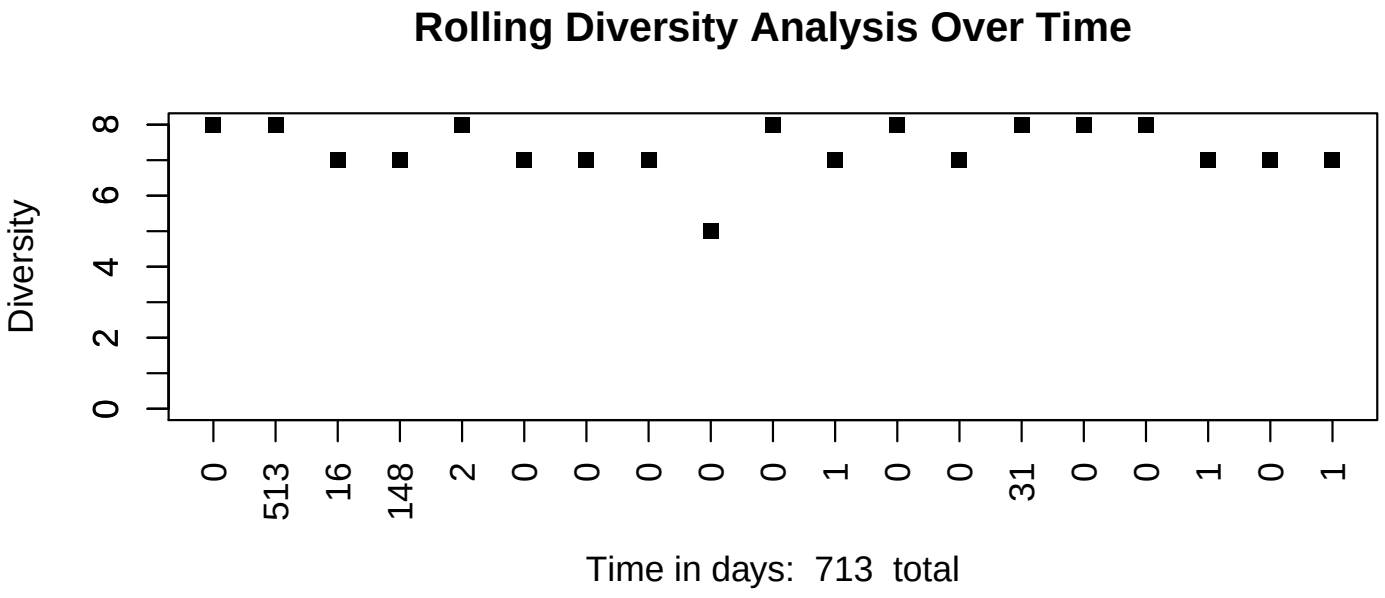


Level 8: No structural similarities

43% of pipes submitted by prolific authors represent tweaks
52% of pipes submitted by prolific authors represent new initiatives

RQ3: Characteristics of most prolific authors

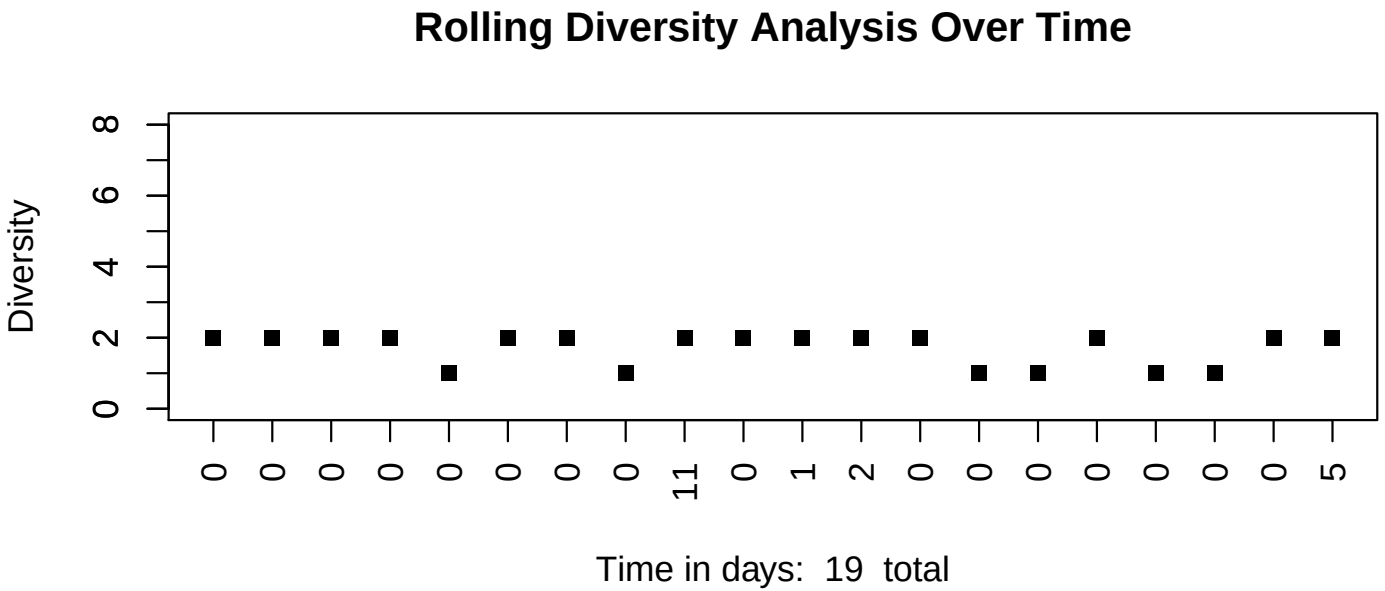
Author Activities



56% of prolific authors consistently submit new initiatives

RQ3: Characteristics of most prolific authors

Author Activities



27% of prolific authors consistently submit tweaks

RQ3: Characteristics of most prolific authors

Take Away #1: 1/2 of participants submit pipes that are novel to their previous contributions

Take Away #2: 1/4 of participants submit pipes that are tweaks of their other pipes

The *real* take away

End-user programmer communities may need . . .

- moderators.
  - → Repository is cluttered with highly similar artifacts (RQ1)

The *real* take away

End-user programmer communities may need . . .

- moderators.
  - → Repository is cluttered with highly similar artifacts (RQ1)
- more sophisticated repository search.
  - → Many pipes are very structurally similar to other pipes in the repository (RQ1)
  - → Early authors create less diverse pipes than later authors (RQ2)

The *real* take away

End-user programmer communities may need . . .

- moderators.
  - → Repository is cluttered with highly similar artifacts (RQ1)
- more sophisticated repository search.
  - → Many pipes are very structurally similar to other pipes in the repository (RQ1)
  - → Early authors create less diverse pipes than later authors (RQ2)
- artifact development support.
  - → Tweaks represent missed opportunities for parameterization (RQ3)
  - → Many shared pipes are tweaks on previously-committed pipes by the same author (RQ3)

Threats to Validity

- Internal
  - → History (the pipes were sampled at different times)
  - → Selection (the repository only provides public pipes)
- Construct
  - → Interaction of different factors
  - → Mono-method bias on diversity (only consider structural diversity, not semantic)
- External
  - → Generalizability (only studied one community)
  - → Sampling bias (could not control search results when sampling)

Conclusion

- Authors utilize the repository in different ways
- As authors gain experience in the environment, they tend to make more valuable contributions to the repository
- There is a need for better support to help end-user programmer communities continue to progress and grow
- To generalize the results, we are interested in extending the metrics to other languages and repositories

To facilitate replication, the data used in this analysis is available:
`http://cse.unl.edu/~kstolee/esem2011/artifacts.html`