

Analytics with
GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks

Software Development Analytics with GrimoireLab

Jesus M. Gonzalez-Barahona

Universidad Rey Juan Carlos

@jgbarah <http://github.com/jgbarah/presentations>

Intl. Summer School on Visual Soft. Analytics
Leipzig (Germany), September 23rd 2019

Analytics with GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks

*It is difficult to improve
if you cannot measure
and track your improvement*

Our plan today

1 A bit of context

Dealing with dynamic complexity

2 Data sources

3 GrimoireLab

4 Cauldron Alpha

5 Case studies

Activity

Remaining code

Performance

Demographics

Diversity

6 Final remarks

Analytics with
GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks



Analytics with
GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks

Me and my two hats

Uni Rey Juan Carlos:

- Understanding free, open source software
- Data analytics approach
- Data visualization in XR

<http://gsyc.es/jgb>



Me and my two hats

Bitergia:

- From research to the real world
- Understanding software development
- Data analytics approach

<http://bitergia.com>



Analytics with
GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks

Recommendations

- Open your laptop
- Download the slides (they have links)
- Visit Alpha.Cauldron.io and produce your own dashboard
- Play with the dashboards
- Understand the interpretations behind the numbers

<https://alpha.cauldron.io>

Analytics with GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

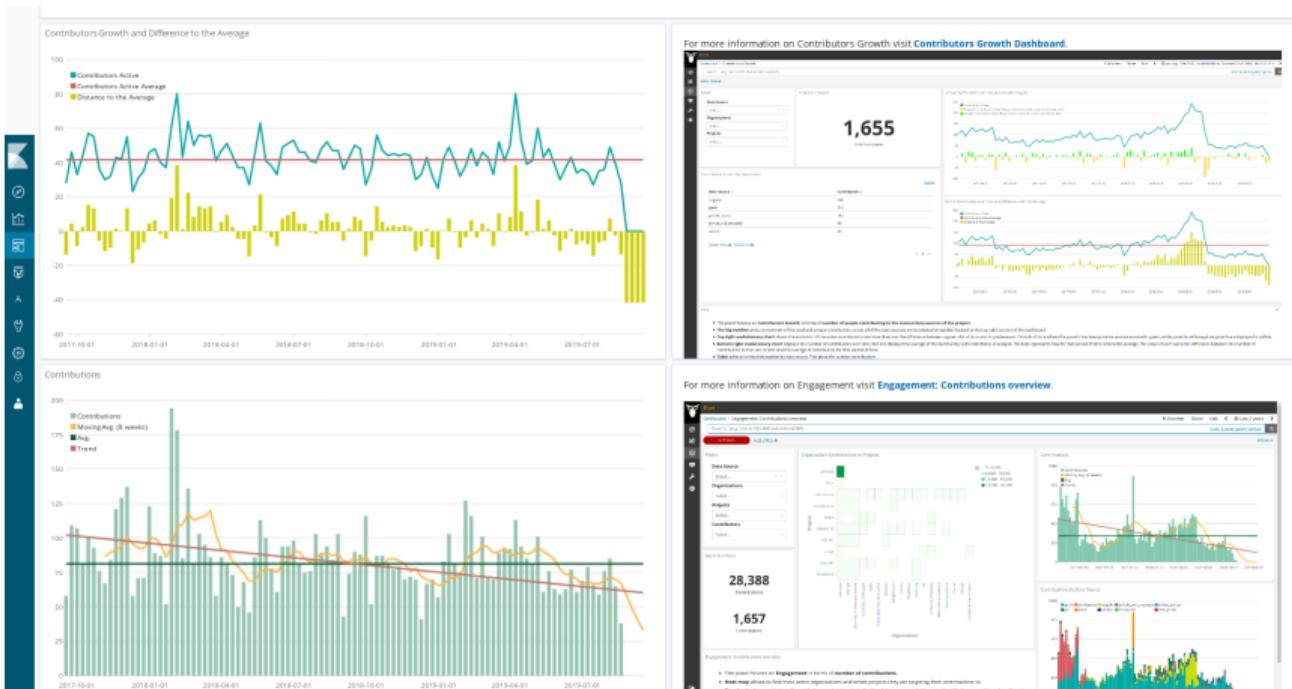
Performance

Demographics

Diversity

Final remarks

Cauldron Alpha



Analytics with GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks

A bit of context

Dealing with dynamic complexity

Analytics with
GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks

Development projects may be large and complex



Projects may be large and complex...
and dynamic

It's difficult to...

- ...track what's happening
- ...understand why it's happening
- ...react quickly
- ...evaluate results of reaction

If data is available
analytics may come to the rescue

A continuous process

Figure out your interest

Find out available data

Define key parameters

Monitor, understand, detect deviations

Act to correct, improve

Track results

Measure → Monitor → Act

A continuous process

Case example: Overall development activity

Interest: activity

Data: changes to code, tickets

Parameters: commits, tickets closed

Monitoring: charts, numbers

Observation: numbers declining

Action: allocate more developer effort

Analytics with
GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks



Analytics with
GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

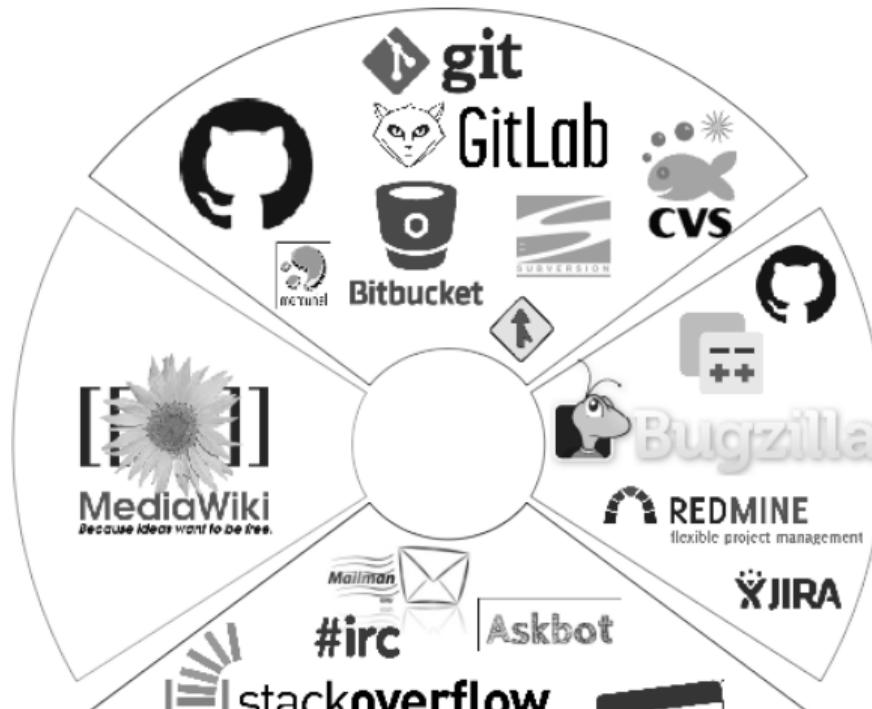
Performance

Demographics

Diversity

Final remarks

Repositories, repositories...



Source code management

- Client/server: CVS, Subversion
- Decentralized: git, Mercurial, Bazaar, etc.
- Most of them accessible through git...
(with some problems)
- Can be integrated with other tools:
Gerrit, GitHub, GitLab, etc.

Analytics with
GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks

Issue tracking

Many different systems:

- Bugzilla
- Jira
- GitHub issues
- GitLab Issues
- Phabricator
- RedMine...

Each with a different model, data, operations...

Analytics with
GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks

Code review

Usually: peer review pre-merge review

Different methods:

- Mailing lists (eg: Linux)
- Gerrit (eg: OpenStack)
- GitHub pull requests (eg: ElasticSearch)
- GitLab merge requests (eg: GNOME)
- or even Jira, Bugzilla...

Much of the control on the software lies here

Analytics with
GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context
Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity
Remaining code
Performance
Demographics
Diversity

Final remarks

Async communication

Mailing lists:

- Mailing lists systems (Mailman)
- Google Groups
- Mailing list archivers

Forums: too many to mention

Question/Answer sites: StackOverflow, Askbot

Information is always archived

Sync communication

Systems:

- Traditionally: IRC
- Nowadays: Slack & many others
- Not always text/based (eg: videoconferences)

Notes:

- In many cases, lack of archives
- Privacy concerns: considered informal

Analytics with
GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks



Analytics with
GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

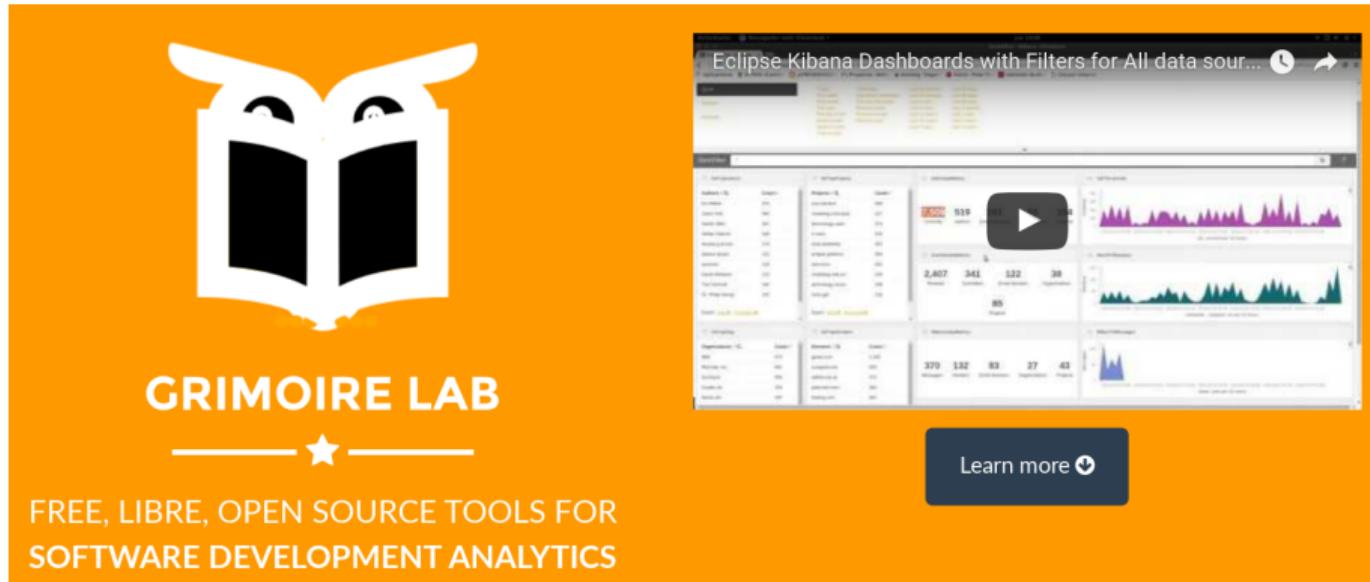
Remaining code

Performance

Demographics

Diversity

Final remarks



The image shows the GrimoireLab landing page. It features a large orange background with a white owl logo and the text "GRIMOIRE LAB". Below the logo is a star icon. The text "FREE, LIBRE, OPEN SOURCE TOOLS FOR SOFTWARE DEVELOPMENT ANALYTICS" is displayed. To the right is a screenshot of the Eclipse Kibana Dashboards interface, showing various data visualizations and filters. A "Learn more" button is visible.

<https://chaoss.github.io/grimoirelab>

Analytics with GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

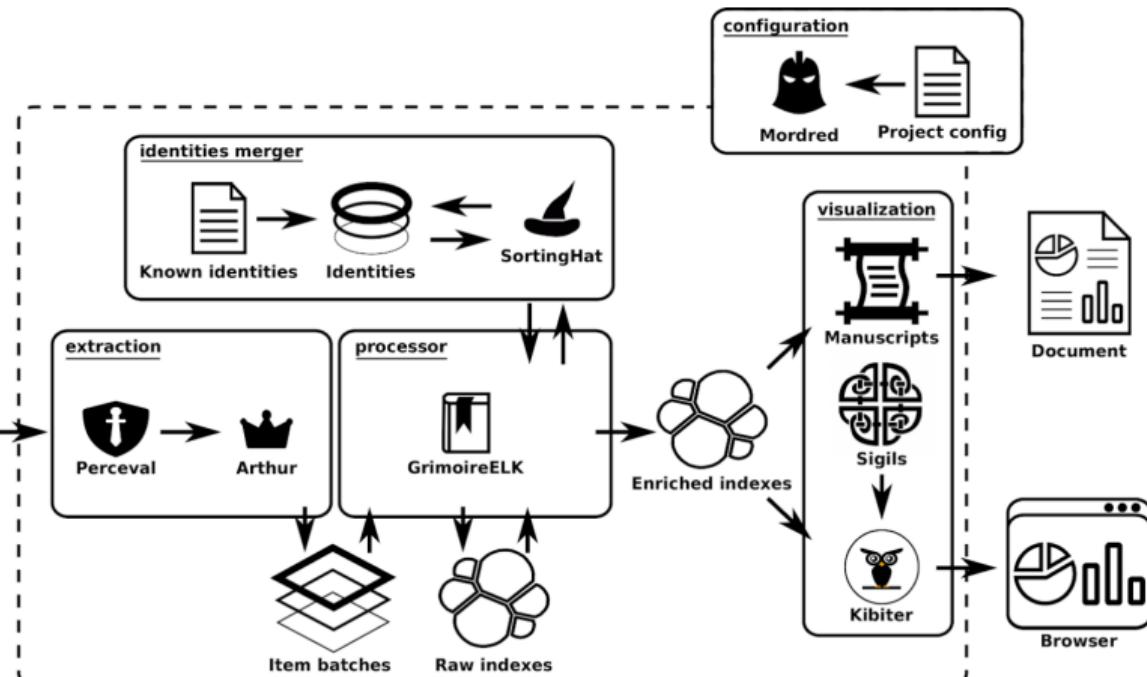
Remaining code

Performance

Demographics

Diversity

Final remarks



<https://chaoss.github.io/grimoirelab>

Analytics with
GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context
Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks

Main components

- Perceval: data retrieval
- Arthur: retrieval orchestration
- GelK: enrichment
- SortingHat: identity management
- ElasticSearch (*): database
- Kibiter: dashboard (light fork of Kibana)
- Sigils: visualizations for Kibana/Kibiter

(*) Not a part of GrimoireLab

Different scenarios

- JSON dump of a single repo
- Database dump of a collection of repos
- Database with unified identities
- Visualization in an interactive dashboard
- Generation of reports
- Dynamic reports (eg. Pandas notebooks)
- ...

Analytics with GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks

Perceval

```
$ python3 -m venv gl
$ source gl/bin/activate
(gl) $ pip install grimoirelab
(gl) $ perceval git \
      https://github.com/chaoss/grimoirelab-perceval
(gl) $ perceval github \
      chaoss grimoirelab-perceval
```

<https://chaoss.github.io/grimoirelab-tutorial/perceval>

Analytics with GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks

```
{"backend_name": "Git",
"backend_version": "0.11.1",
"category": "commit",
"classified_fields_filtered": null,
"data": {
    "Author": "Santiago Due\u00f1as <sduenas@bitergia.com>",
    "AuthorDate": "Tue Aug 18 18:08:27 2015 +0200",
    "Commit": "Santiago Due\u00f1as <sduenas@bitergia.com>",
    "CommitDate": "Tue Aug 18 18:08:27 2015 +0200",
    "commit": "dc78c254e464ff334892e0448a23e4cfbf637a3",
    "files": [
        "action": "A",
        "added": "10",
        "file": ".gitignore",
```

Analytics with GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks

```
{"backend_name": "GitHub",
"backend_version": "0.22.1",
"category": "issue",
"classified_fields_filtered": null,
"assignee_data": {},
"assignees": [],
"assignees_data": [],
"author_association": "CONTRIBUTOR",
"body": "Based on Sphynx, prepared...",
"closed_at": "2016-01-04T13:51:56Z",
"comments": 0,
"comments_data": [],
"comments_url": "https://api.github.com/...",
"created_at": "2016-01-03T23:46:04Z",
```

Perceval as a module

```
#! /usr/bin/env python3
from perceval.backends.core.git import Git

repo_url = 'http://github.com/chaos/grimoirelab-perceval'
repo_dir = '/tmp/perceval.git'

repo = Git(uri=repo_url, gitpath=repo_dir)
for commit in repo.fetch():
    print(commit['data']['commit'])
```

```
import argparse
from perceval.backends.core.git import Git

parser = argparse.ArgumentParser(description = "Count commits")
parser.add_argument("repo", help = "Repository url")
parser.add_argument("--print", action='store_true', help = "Print commits")
args = parser.parse_args()

repo = Git(uri=args.repo, gitpath='/tmp/perceval.git')
count = 0
for commit in repo.fetch():
    if args.print:
        print(commit['data']['commit'])
    count += 1
print("Number of commits: %d." % count)
```

SirMordred

Producing a dashboard:

- Elasticsearch installed
- Kibana / Kibiter installed
- MariaDB installed
- Config: mordred.cfg, projects.json, identities.yaml, menu.yaml

(gl) \$ mordred -c mordred.cfg

<https://chaoss.github.io/grimoirelab-tutorial/sirmordred>

Analytics with
GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks



GrimoireLab

Software development analytics with
free, open source software

(a CHAOSS project)

chaoss.github.io/grimoirelab
chaoss.github.io/grimoirelab-tutorial



Analytics with
GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

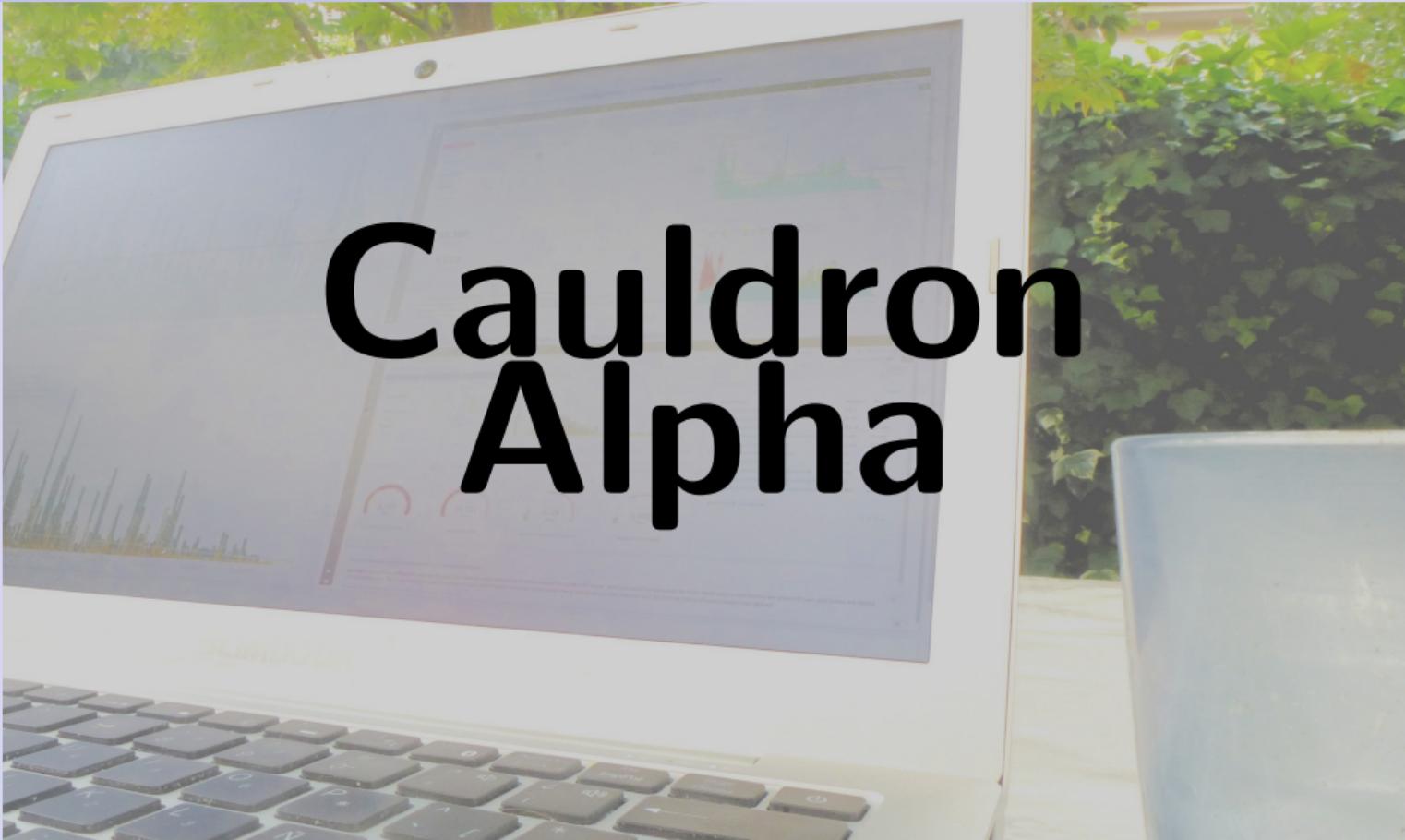
Remaining code

Performance

Demographics

Diversity

Final remarks



Analytics with GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks

The screenshot shows the 'Cauldron (a version)' interface. At the top, there's a navigation bar with a logo, the title 'Cauldron (a version)', a 'Send us feedback!' button, and a user icon. A yellow banner below the title contains the text: 'This is an alpha version of the Cauldron. All data can be removed without prior notice. Feedback is welcome!'. The main content area features a large heading 'Welcome to Cauldron (a version)!', a subtext 'Create an analytics environment for the software development projects that matter to you!', and a paragraph explaining the service: 'The Cauldron is a PoC service developed by [Bitergia](#) to analyze community and processes in software development projects.' A blue button labeled 'Analyze a project' is located on the right side of this section.

<https://alpha.cauldron.io>

Analytics with
GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context
Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks

Internals

- Elasticsearch database (data dumps)
- MariaDB database (identities)
- Django App (frontend)
- Python App, on Mordred (workers)
- Kibana (visualization)
- OpenDistro: integration

<https://alpha.cauldron.io>

Analytics with GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks

The screenshot shows the Cauldron Alpha dashboard interface. At the top, there's a header with the title "Cauldron (a version)" and a "Send us feedback!" button. On the right, a user profile icon for "jgbarah" is visible.

The main area has a title "Dashboard 93" with an edit icon. To the right is a "View project data" button. Below the title, it says "0 data sources found" with dropdown filters for "backend: any" and "status: all". A "Refresh all" button is also present.

The dashboard features three data source configuration sections:

- GitHub:** Shows an icon, the name "GitHub", and a status indicator. Below is a text input field containing "django" and a blue "Add" button.
- Git:** Shows an icon, the name "Git", and a status indicator. Below is a text input field containing "git repository url" and a blue "Add" button.
- GitLab:** Shows an icon, the name "GitLab", and a status indicator. Below is a text input field containing "gitlab owner or own" and a blue "Add" button.

A prominent yellow callout box in the center-right area contains the text "Start adding data sources!" with a left arrow icon and a close "X" icon. It also includes the explanatory text: "Projects are defined by set of different data sources. Add them with the tools in the side bar."

Analytics with GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks

The screenshot shows the Cauldron Alpha interface for analyzing the Django project. At the top, there's a header with the Cauldron logo, a feedback button, and a user profile icon for 'igbarah'. Below the header, there's a section titled 'Add data sources' with five input fields: GitHub, Git, GitLab, Meetup, and a note about future data source additions. To the right, the main content area is titled 'Django' with a blue edit icon. It displays a table of 36 data sources found, with a 'Refresh all' button at the top right. The table has columns for Status, Data source, Last refresh, Duration, and actions (Logs, Delete, Refresh). All entries show a green checkmark and were last refreshed a month ago.

Status	Data source	Last refresh	Duration	
✓	https://github.com/django/ticketbot.git	a month ago	0 00:00:17	Logs Delete Refresh
✓	https://github.com/django/djangonippets.org.git	a month ago	0 00:00:25	Logs Delete Refresh
✓	https://github.com/django/djangoproject.com.git	a month ago	0 00:00:47	Logs Delete Refresh
✓	https://github.com/django/djangobench.git	a month ago	0 00:00:21	Logs Delete Refresh
✓	https://github.com/django/django-localflavor.git	a month ago	0 00:00:36	Logs Delete Refresh
✓	https://github.com/django/django-formtools.git	a month ago	0 00:00:24	Logs Delete Refresh
✓	https://github.com/django/django-docs-translations.git	a month ago	0 00:00:33	Logs Delete Refresh
✓	https://github.com/django/django-contrib-comments.git	a month ago	0 00:00:21	Logs Delete Refresh
✓	https://github.com/django/django-box.git	a month ago	0 00:00:17	Logs Delete Refresh
✓	https://github.com/django/django.git	a month ago	0 00:11:09	Logs Delete Refresh
✓	https://github.com/django/deps.git	a month ago	0 00:00:20	Logs Delete Refresh

It is planned for the future to add more data sources (like Discourse or Slack), but feel free to suggest any other data source option via the feedback button!

Analytics with GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

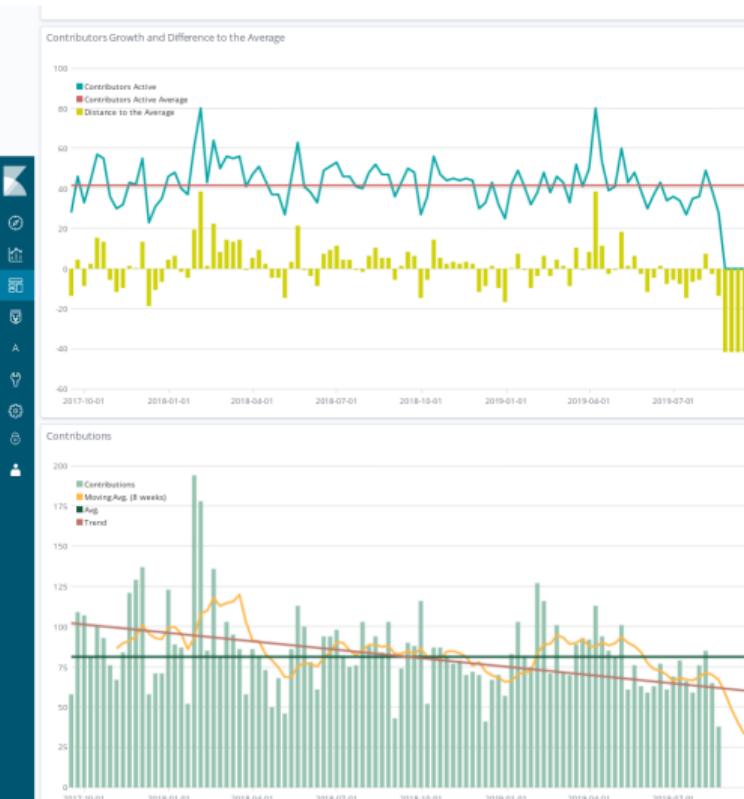
Remaining code

Performance

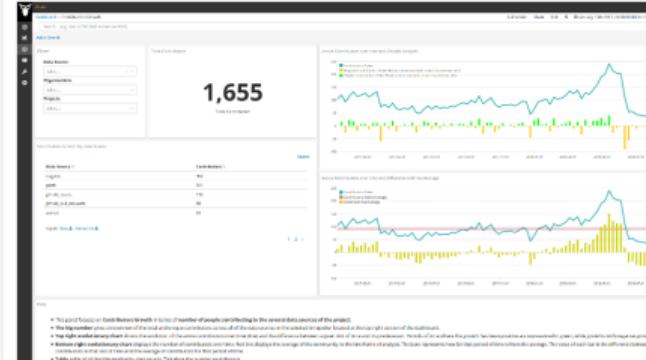
Demographics

Diversity

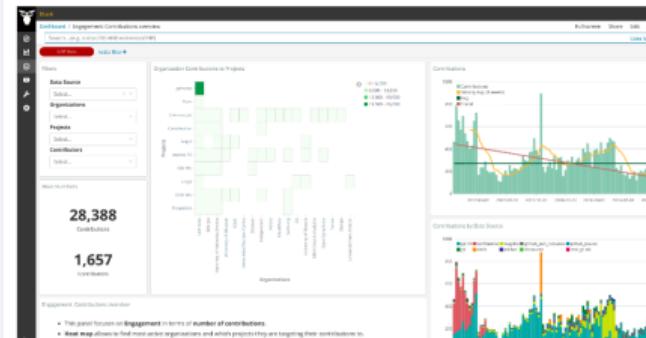
Final remarks



For more information on Contributors Growth visit [Contributors Growth Dashboard](#).



For more information on Engagement visit [Engagement: Contributions overview](#).



Dashboards

[Create new dashboard](#)

Search...

<input type="checkbox"/>	Title ↑	Description	Actions
<input type="checkbox"/>	Contributors Growth	Contributors Growth Panel by Bitergia	Edit
<input type="checkbox"/>	Efficiency: Timing overview	Timing overview panel for tickets/issues by Bitergia	Edit
<input type="checkbox"/>	Engagement: Contributions overview	Overview panel for analyzing engagement in terms on contributions by Bitergia.	Edit
<input type="checkbox"/>	Meetup	Meetup Overview panel by Bitergia	Edit
<input type="checkbox"/>	Metrics Overview	Entry panel showing overall metrics.	Edit
<input type="checkbox"/>	Overview	Overview panel	Edit

Rows per page: 20 ▾

Analytics with
GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks



Analytics with
GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context
Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity
Remaining code
Performance
Demographics
Diversity

Final remarks

Tracking involved parties

Development is much more than developers
(this is explicit in FOSS & inner sourcing)

- Developers: all repositories
- Contributors: issue tracking, async communication
- Users: async communication, ...
- Ecosystem: difficult to track

Analytics with
GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks

Case studies

Activity

Activity / size

- committing patches:
source code management system
- reporting, commenting or fixing bugs:
issue tracking system
- submitting patches or reviewing them:
code review system
- sending messages:
async or sync communication systems

Most common cases

- Parameters reflecting activity for a period.
- People active for a certain period.
- Evolution of any of them.
- Trends for any of them.

Difficult to compare between projects
Interesting to compare in-project

Analytics with GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks

Many facets



Analytics with GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

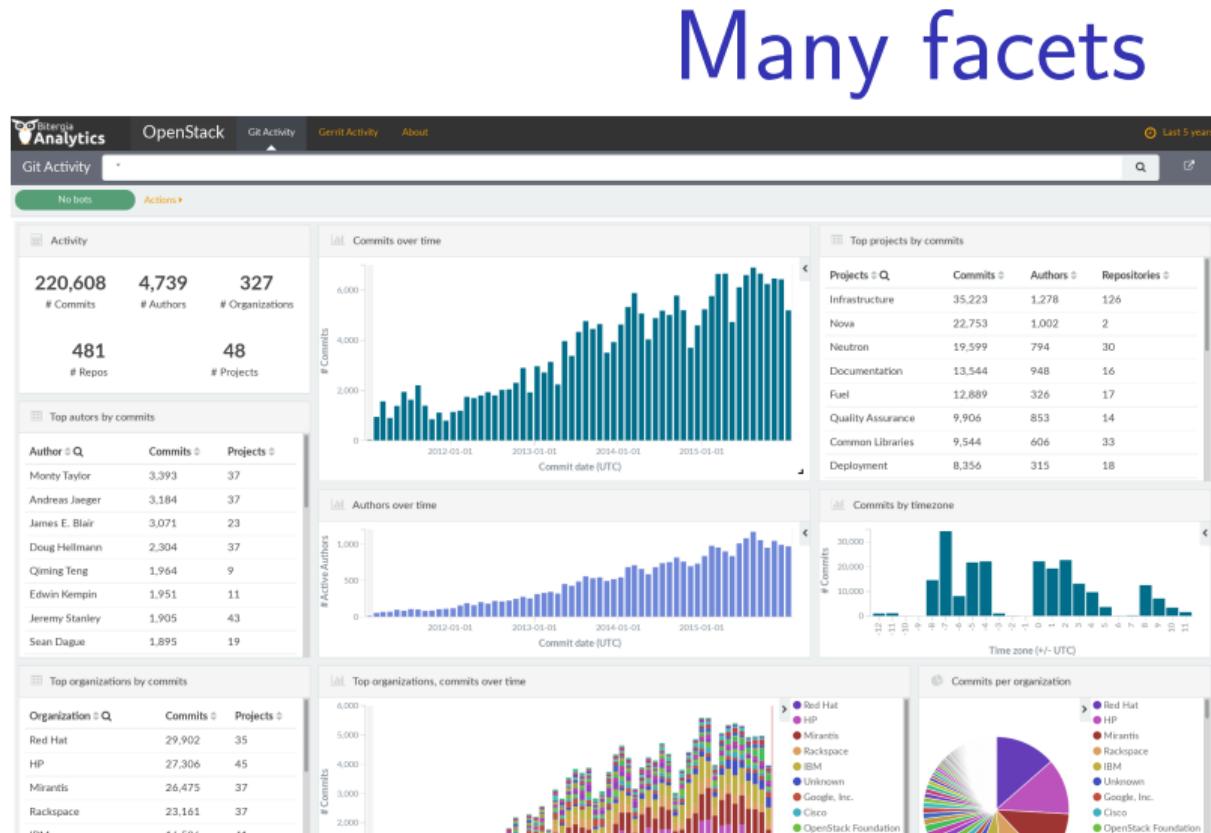
Remaining code

Performance

Demographics

Diversity

Final remarks



Analytics with GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks

Case studies

Remaining code

Analytics with GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

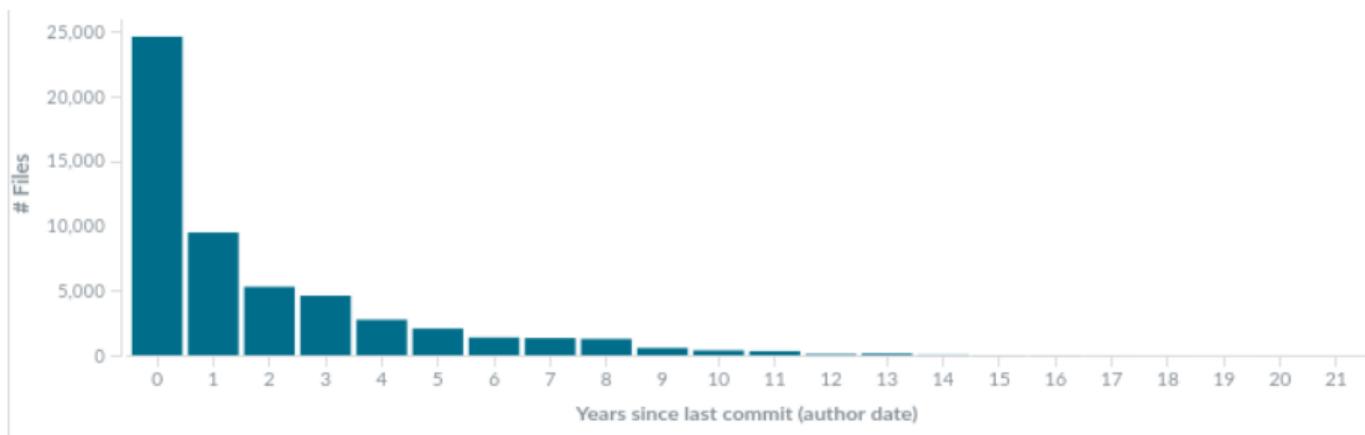
Performance

Demographics

Diversity

Final remarks

How old is code?



[Linux kernel, July 2016, C files by last commit]

Analytics with GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

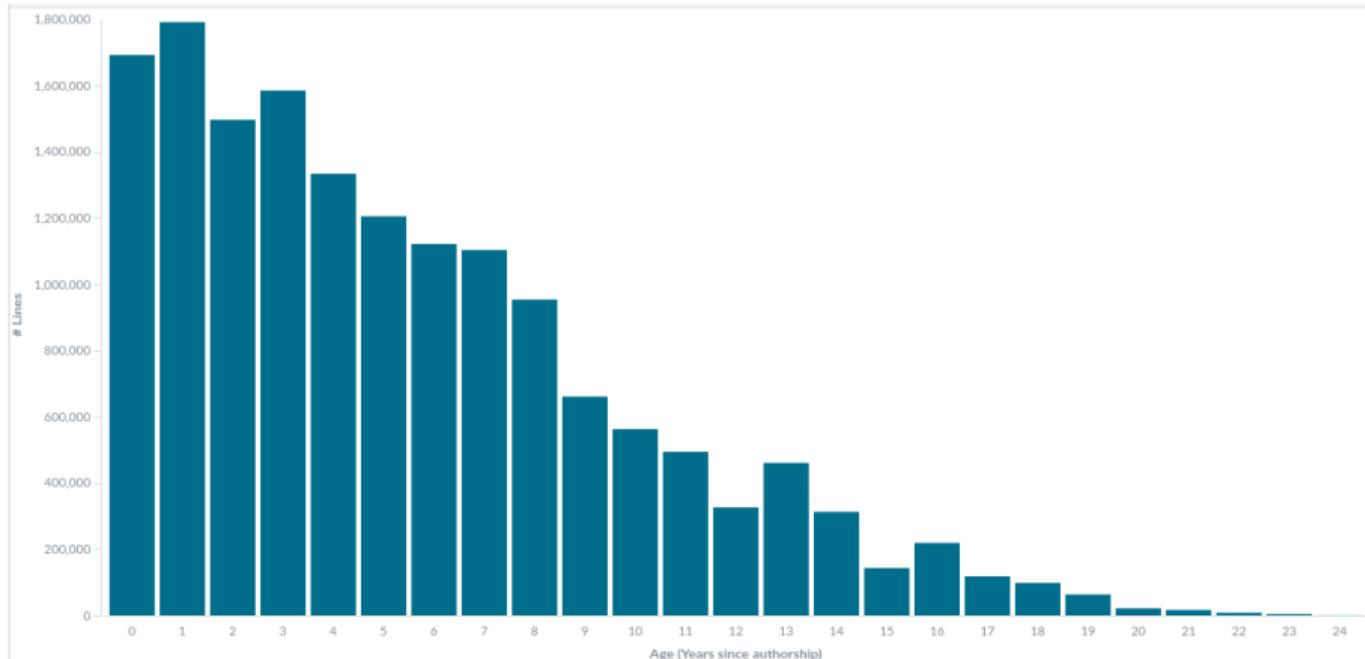
Remaining code

Performance

Demographics

Diversity

Final remarks



[Linux kernel, July 2016, lines in C files by age]

Analytics with
GrimoireLabJesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

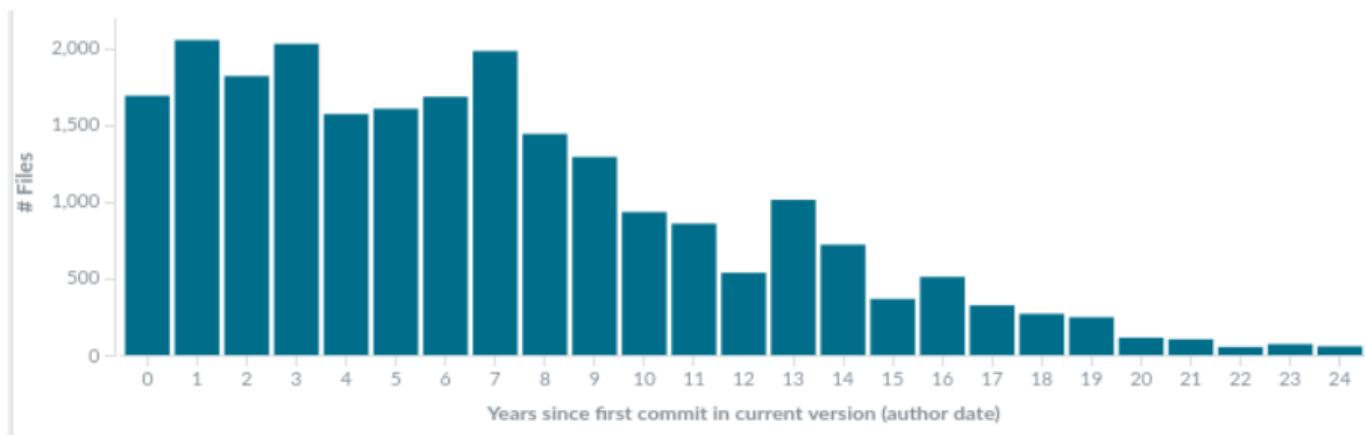
Performance

Demographics

Diversity

Final remarks

How old is code (3)



[Linux kernel, July 2016, C files by first remaining commit]

Analytics with GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

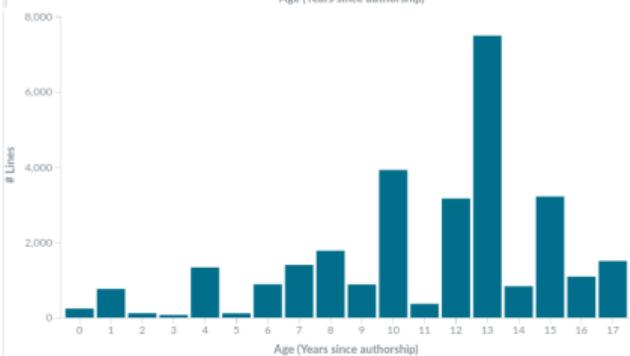
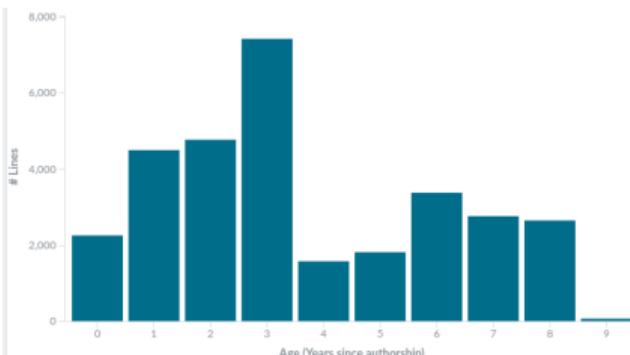
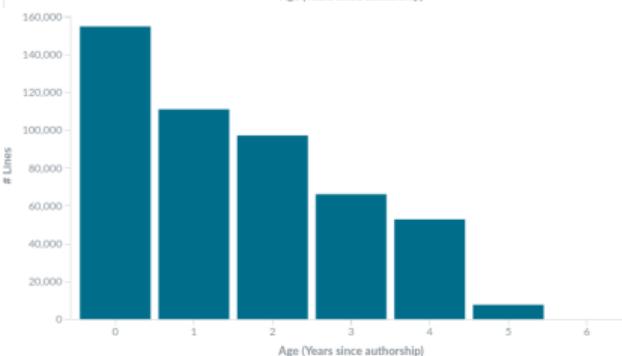
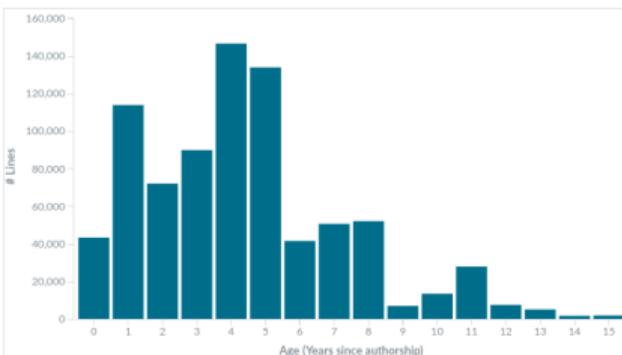
Remaining code

Performance

Demographics

Diversity

Final remarks



Age of lines (data of authorship, “.c” files in Linux)

From top left, clockwise: Wireless, USB, IRDA Ethernet

Analytics with GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks

Case studies Performance

Analytics with GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

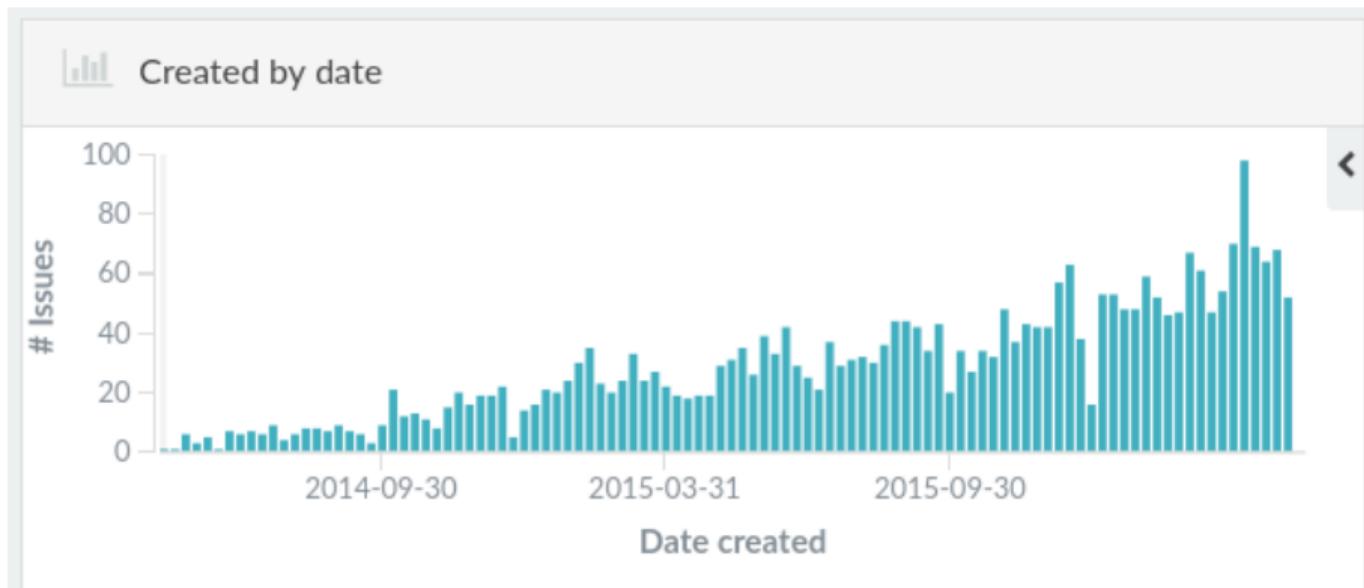
Performance

Demographics

Diversity

Final remarks

Performance (backlog)



Analytics with GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

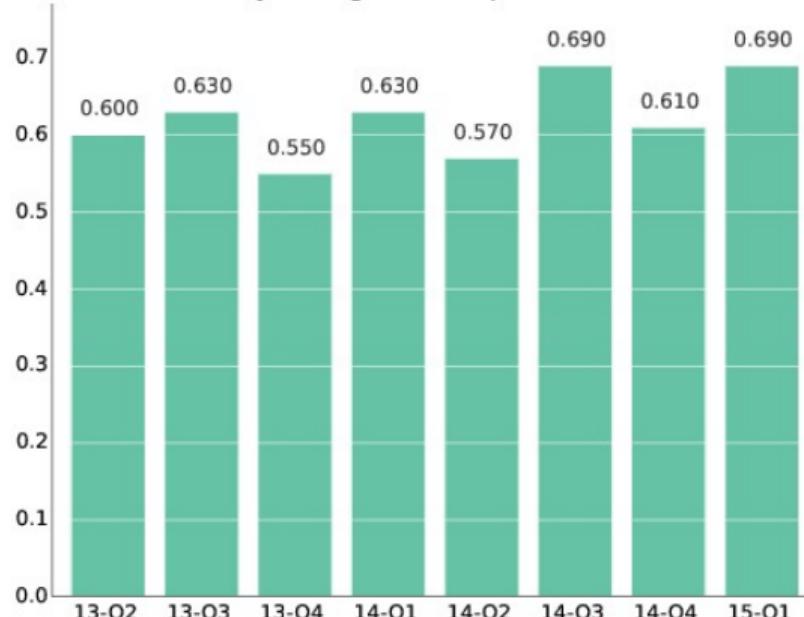
Performance

Demographics

Diversity

Final remarks

Efficiency closing tickets: OpenStack Software



Efficiency. Example: closed/opened tickets per quarter

Analytics with GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

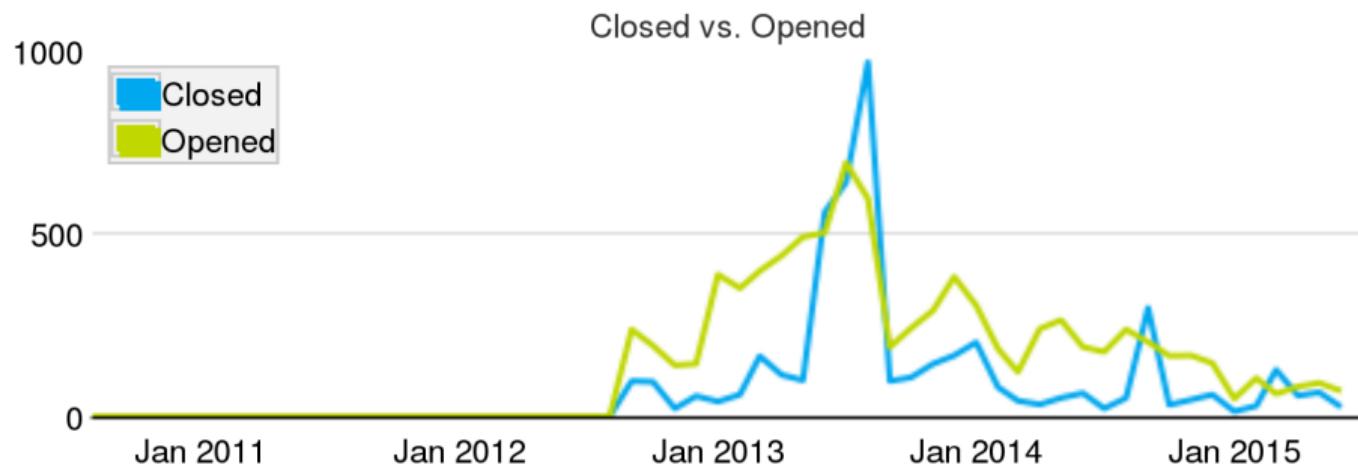
Performance

Demographics

Diversity

Final remarks

Tickets



Analytics with
GrimoireLabJesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

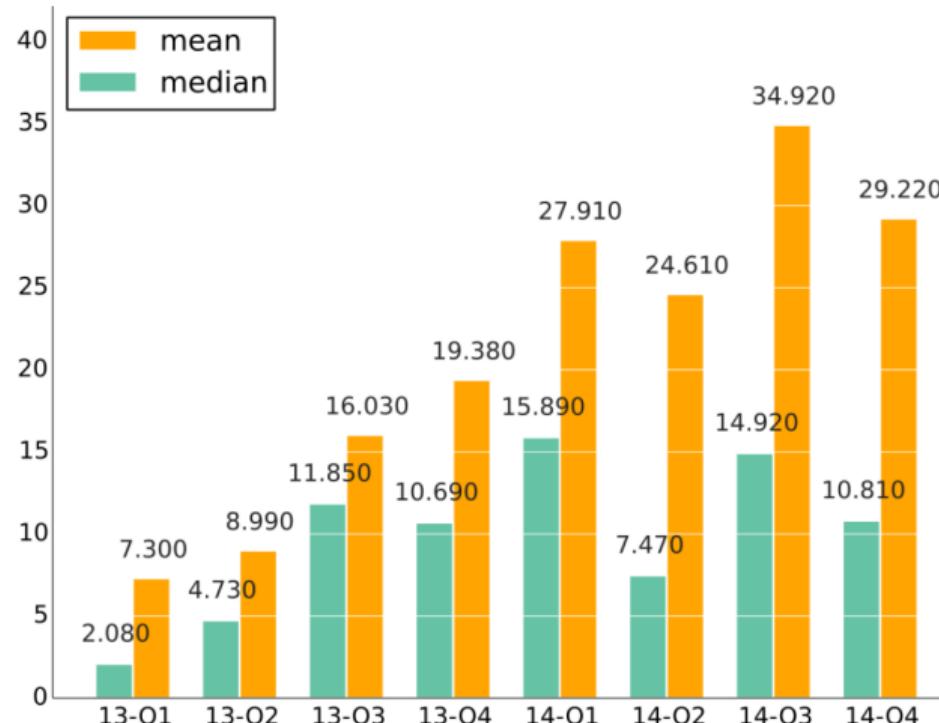
Performance

Demographics

Diversity

Final remarks

Review: time to merge



Analytics with
GrimoireLabJesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

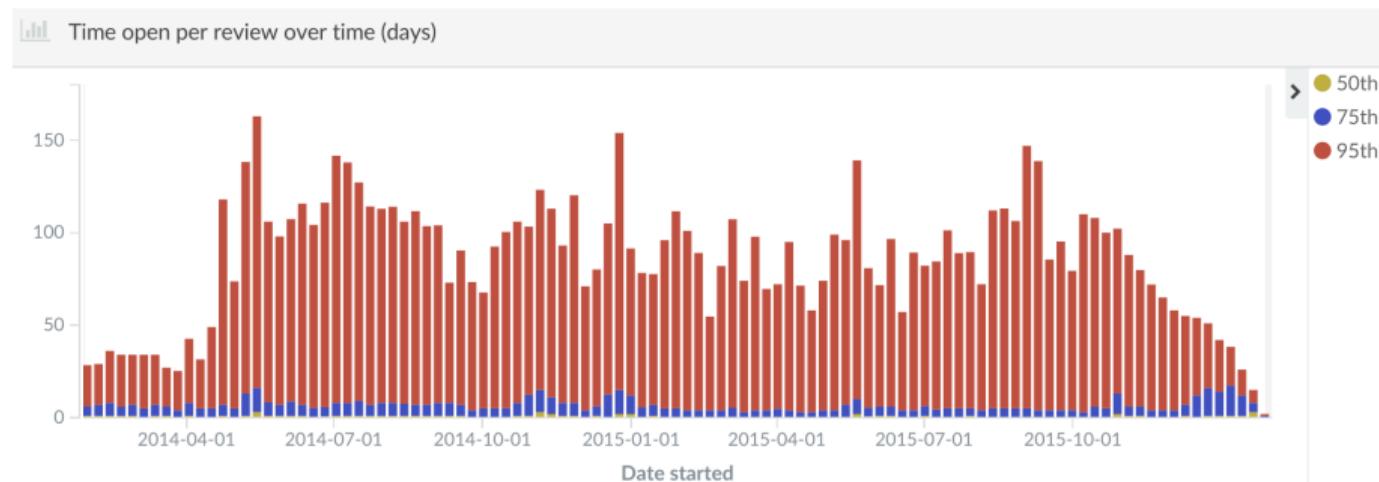
Performance

Demographics

Diversity

Final remarks

Review: time to merge



Analytics with GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

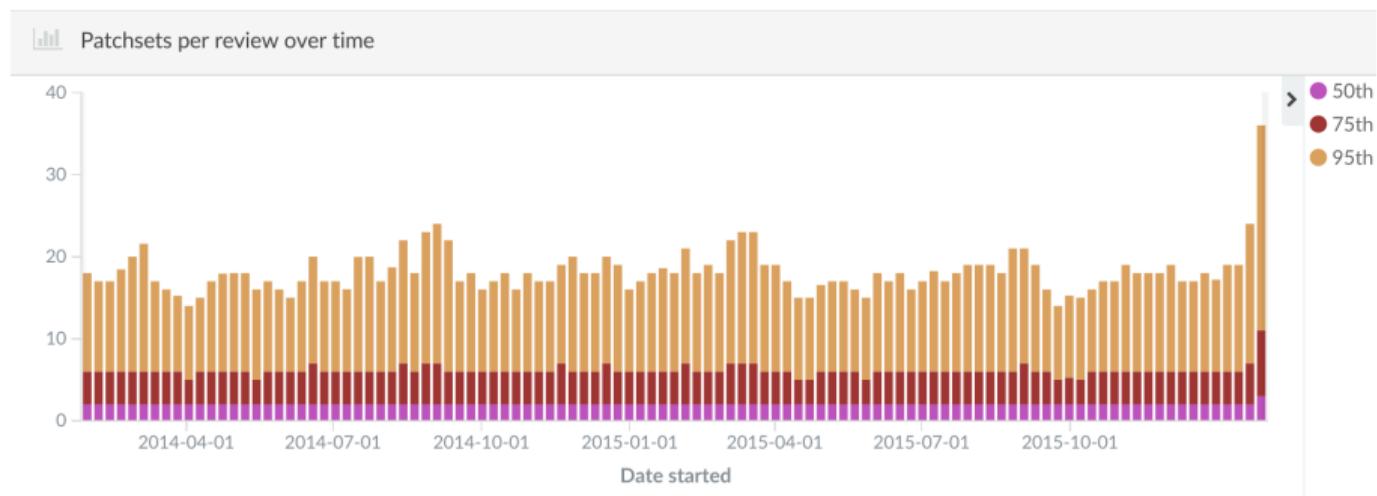
Performance

Demographics

Diversity

Final remarks

Versions per review



The coding process

From idea to implementation

- Story, design
- Ticket(s)
- Code review
- Automated testing
- Commit in code base

The OpenStack case

- Blueprint (if feature), Launchpad
- Ticket (bug, feature), Launchpad
- Code review, Gerrit
- Automated testing, Jenkins
- Commit in code base, Gerrit, Git

Analytics with
GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks

Case studies

Demographics

Analytics with
GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

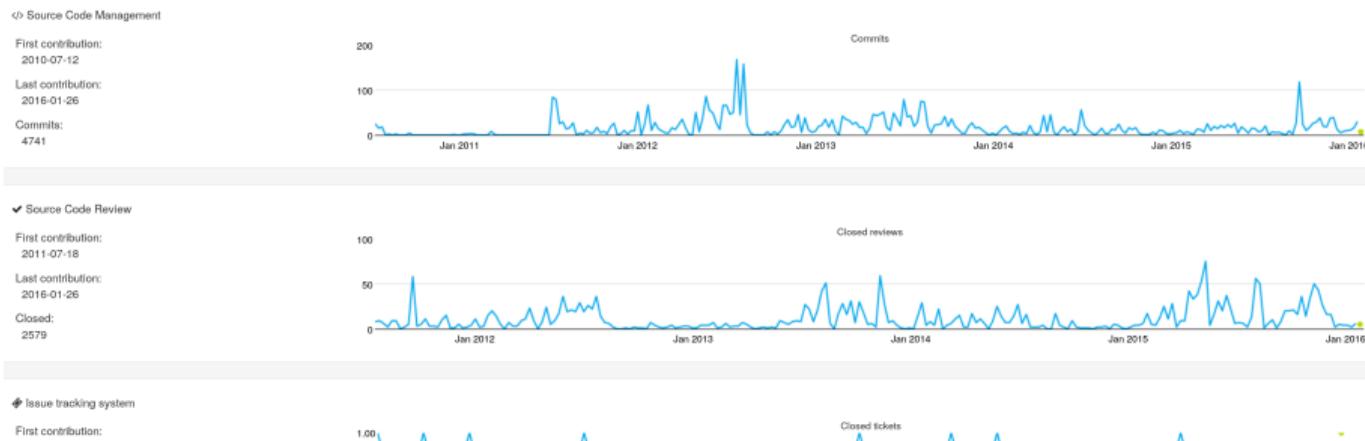
Performance

Demographics

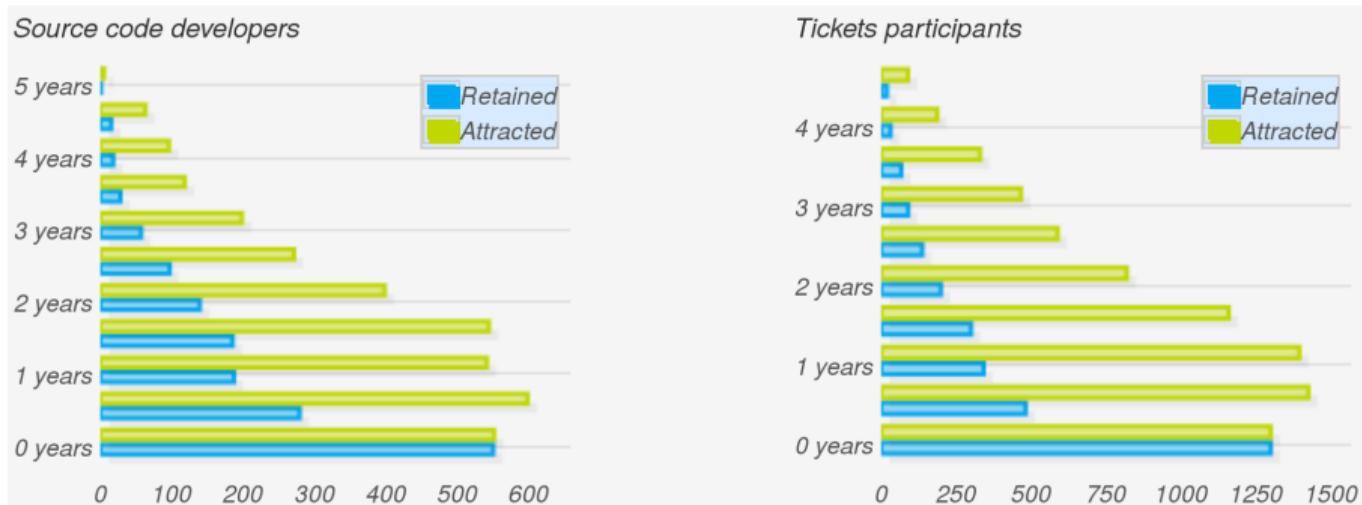
Diversity

Final remarks

- The repository level.
- The class of repository level.
- The project level.
- The global level.



The aging chart



Analytics with
GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks

Case studies

Diversity

Analytics with GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

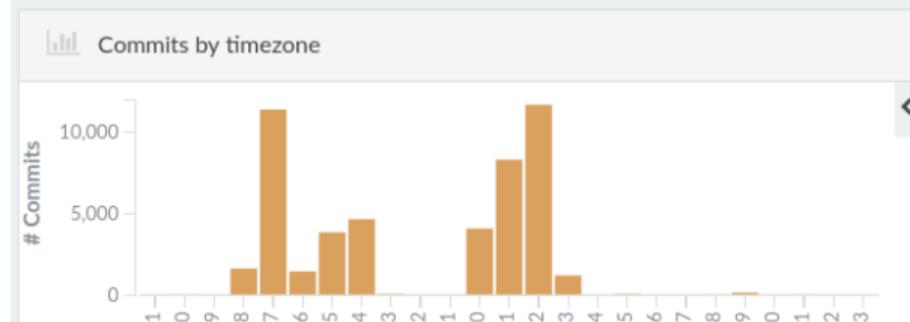
Performance

Demographics

Diversity

Final remarks

Time zones



Analytics with
GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks

GitHub profiles



Analytics with GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

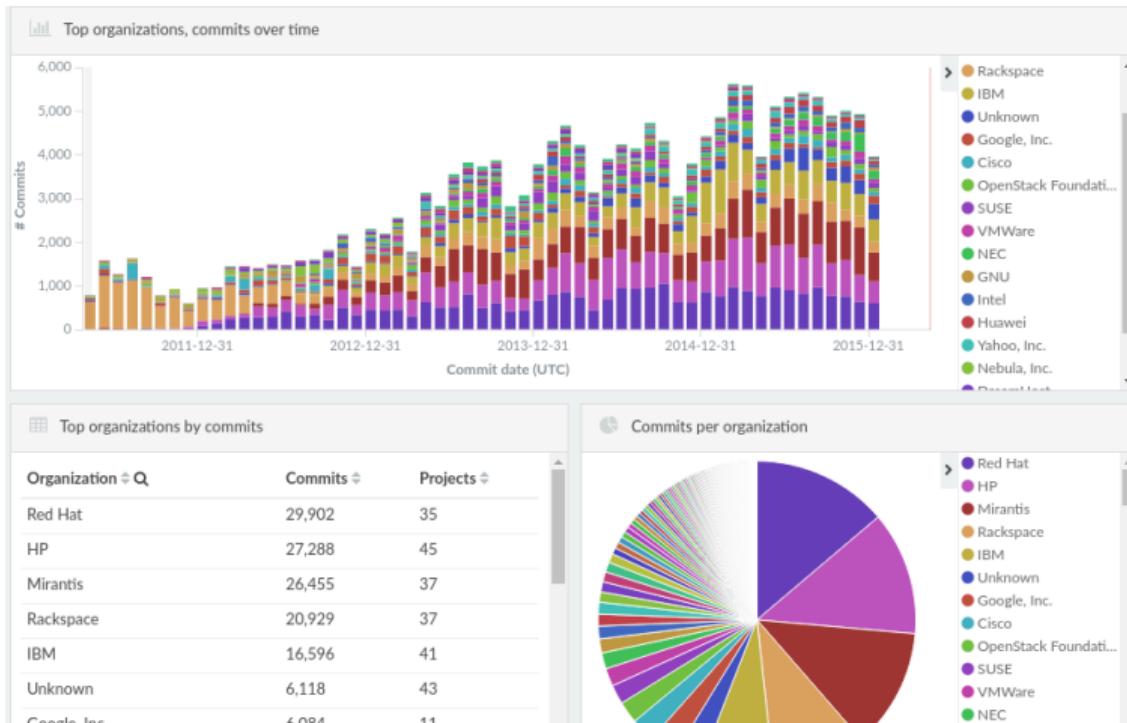
Performance

Demographics

Diversity

Final remarks

Affiliation



Analytics with
GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context
Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks

Apache Pony Factor

Pony Factor (PF) shows the diversity of a project in terms of the division of labor among committers in a project.

*Pony Factor is determined as: “**The lowest number of committers whose total contribution constitutes the majority of the codebase**”*

ke4qqq.wordpress.com/2015/02/08/pony-factor-math/

Analytics with
GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks

Bitergia Elephant Factor



Analytics with
GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context
Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity
Remaining code
Performance
Demographics
Diversity

Final remarks

Bitergia Elephant Factor

The elephant factor shows the diversity of a project in terms of the division of labor among companies (by mean of developers affiliated with them).

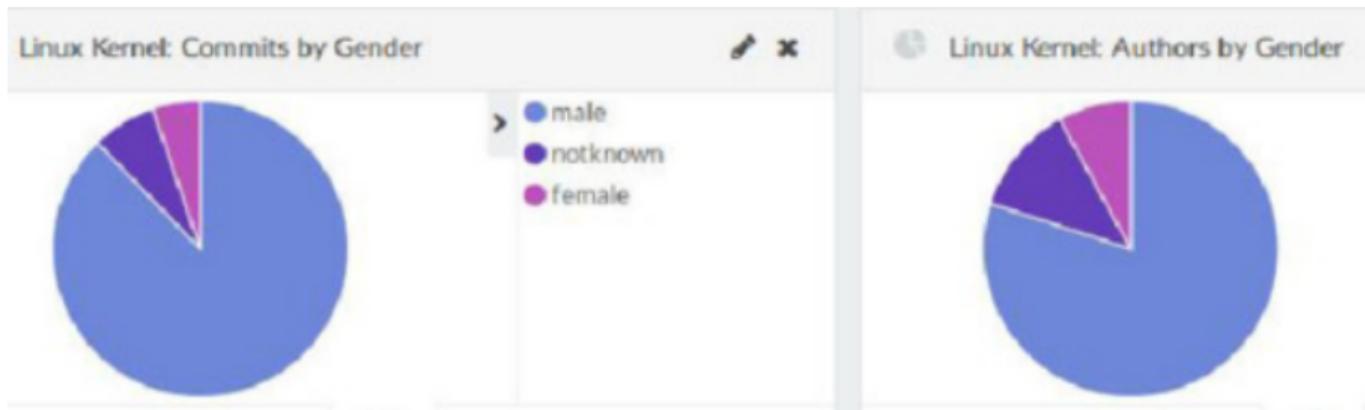
Elephant factor is determined as:

“The lowest number of companies whose total contribution (in commits by their employees) constitutes the majority of the commits”

Some projects (2016)

	Pony Factor	Elephant Factor	Commits (excl bots)
OpenNebula	4	1	12K
Eucalyptus	5	1	25K
CloudStack	14	1	42K
OpenStack	>100	6	126K
CloudFoundry	41	1	60K
OpenShift	10	1	15K
Docker	15	1	18K
Kubernetes	12	1	7K

Diversity: Gender gap



Commits by women: 6.8 % (4 Kcommits)

Women: 9.9 % (330 developers)

Linux kernel, Nov 2015 – Oct 2016

Analytics with
GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks



Room for improvement

- Many other aspects... explore your own
- Refine what is important
- Explore new ways of making data useful
- Tell interesting stories based on data
- Visualization is very important
- Higher-order metrics
- Simplify results, make them meaningful

Analytics with GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks

Summary

If you don't have data
you're just another person
with an opinion

Fortunately, you can have a lot of data...
Unfortunately, having the right data is not easy

<http://chaoss.github.io/grimoirelab>

Credits (1)

- “Man With Two Hats”

Statue by Henk Visch, located in Ottawa, Canada

Picture by Lezumbalaberjenja in Wikimedia Commons

License: Public domain

https://commons.wikimedia.org/wiki/File:Man_With_Two_Hats_Ottawa_Statue_by_lezumbalaberjenja.jpg

- “Crowd at FOSDEM 2008”

by Jesús Corrius

License: CC Attribution 2.0

<http://www.flickr.com/photos/jcorrius/2302302707/>

Analytics with GrimoireLab

Jesus M.
Gonzalez-Barahona

A bit of context

Dealing with dynamic complexity

Data sources

GrimoireLab

Cauldron Alpha

Case studies

Activity

Remaining code

Performance

Demographics

Diversity

Final remarks



©2016-2019 Jesus M. Gonzalez-Barahona.

Some rights reserved. This document is distributed under the terms of the Creative Commons License “Attribution-ShareAlike 4.0”, available in

<http://creativecommons.org/licenses/by-sa/4.0/>

This document (including source) is available from
<https://github.com/jgbarah/presentations>