

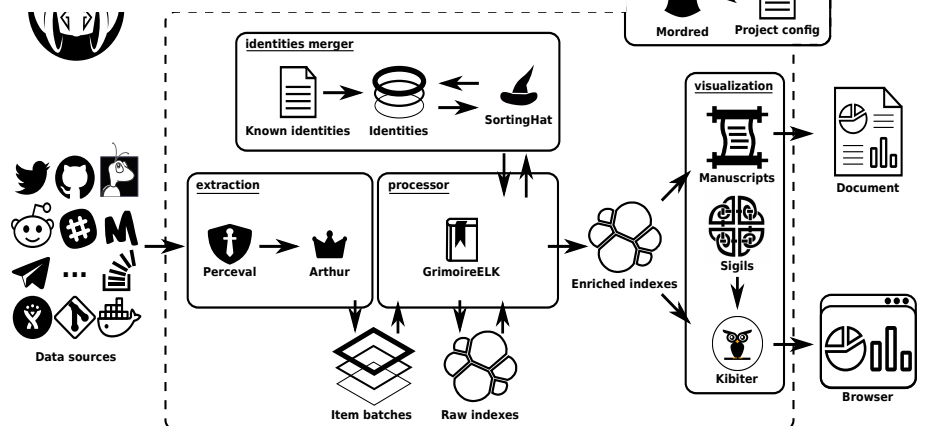
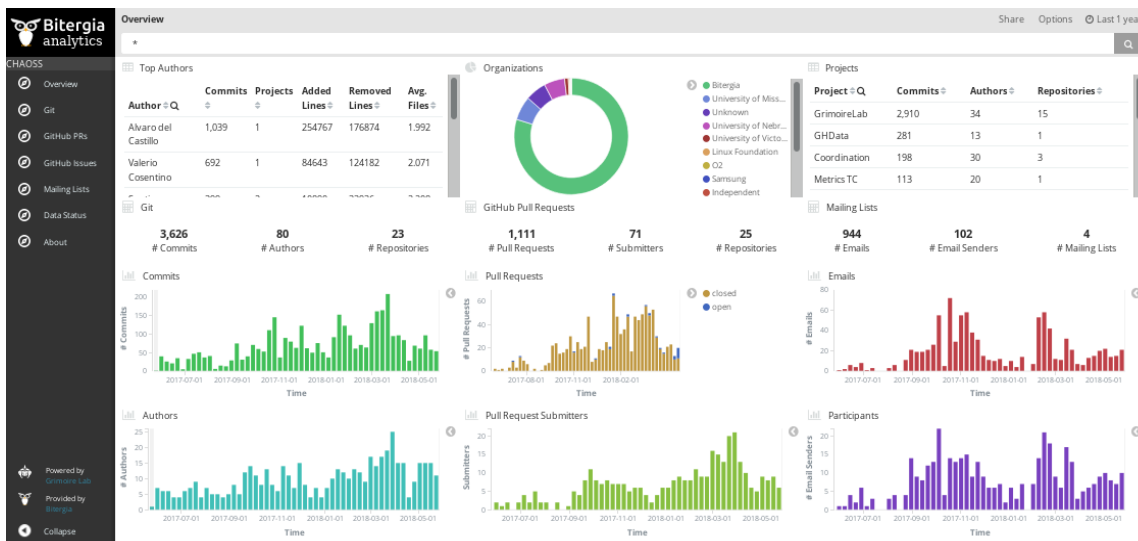


# GrimoireLab

Python toolset for  
software development

analytics

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



# CHA OSS

GrimoireLab is a CHAOSS project

<http://chaoss.community>



# Perceval

A GrimoireLab tool

Retrieve data  
from +20 kinds of  
software development repositories  
with a single Python API

askbot  
bugzilla  
bugzillares  
confluence  
discourse  
dockerhub  
gerrit  
git  
github  
gitlab  
hyperkitty  
jenkins  
jira  
launchpad  
mattermost  
mbox  
mediawiki  
meetup  
nntp  
phabricator  
pipermail  
redmine  
rss  
slack  
stackexchange  
supybot  
telegram

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

```
# url for the git repo to analyze
repo_url = 'http://github.com/grimoirelab/perceval.git'
# directory for letting Perceval clone the git repo
repo_dir = '/tmp/perceval.git'
```

```
# create a Git object, pointing to repo_url, using repo_dir for cloning
repo = Git(uri=repo_url, gitpath=repo_dir)
# fetch all commits as an iterator, and iterate it printing each hash
for commit in repo.fetch():
    print(commit['data']['commit'])
```

```
repo = GitHub(owner=owner, repository=repo, api_token=args.token)
# fetch all issues/pull requests as an iterator, and iterate it printing
# their number, and whether they are issues or pull requests
for item in repo.fetch():
    if 'pull_request' in item['data']:
        kind = 'Pull request'
    else:
        kind = 'Issue'
    print(item['data']['number'], ': ', kind)
```

```
# create a Gerrit object, pointing to hostname, using user for ssh access
repo = Gerrit(url=hostname, user=user)
```

```
# fetch all reviews as an iterator, and iterate it printing each review id
for review in repo.fetch(from_date=from_date):
    print(review['data']['number'])
```

[github.com/chaoss/grimoirelab-perceval](https://github.com/chaoss/grimoirelab-perceval)

**CHA OSS**  
GrimoireLab is a CHAOSS project  
<http://chaoss.community>