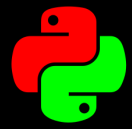
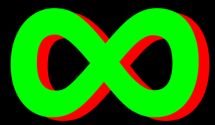
visualizing java code bases

01

Andrey Adamovich @codingandrey

$ whoami

- developer    ...
- devops guy 
- trainer   ...
- speaker Jfokus  DEVOXX™ ...
- co-organizer  

Let's start!

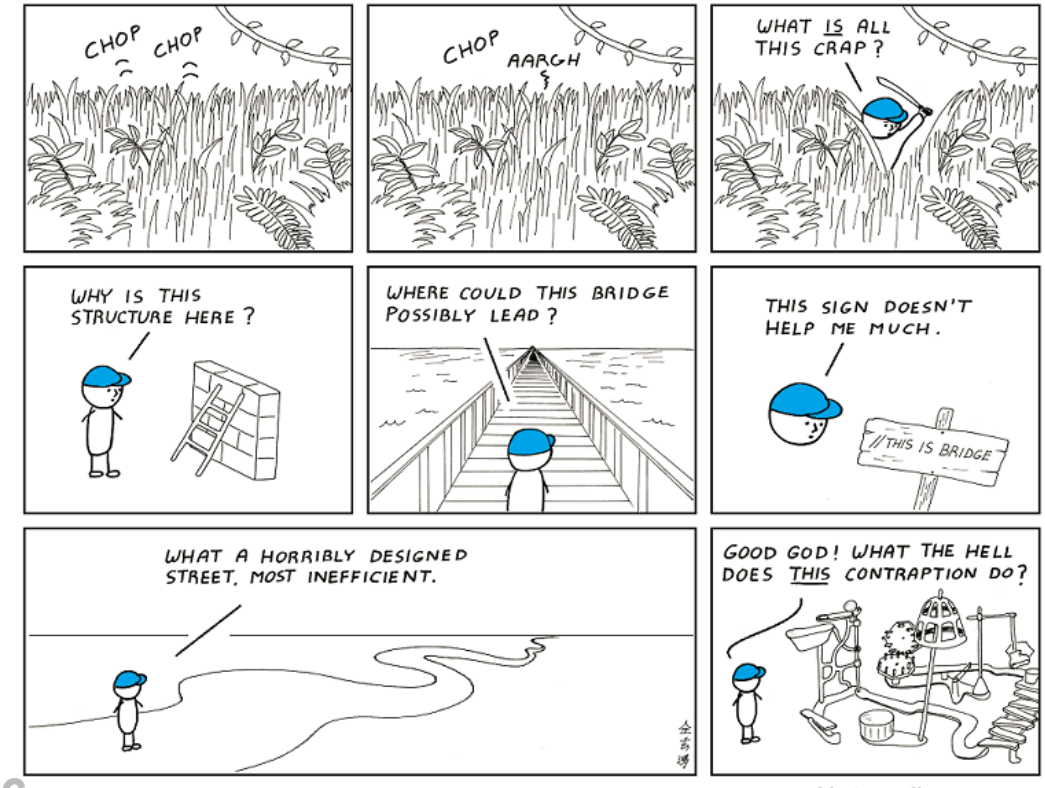
Background

- Find new clients
- Face big code bases
- Need quick analysis
- Need quick results

Code size

- Google: 2 billions LOC
- Facebook: 61 million LOC
- Me: from 20K to 2M LOC

I hate reading...



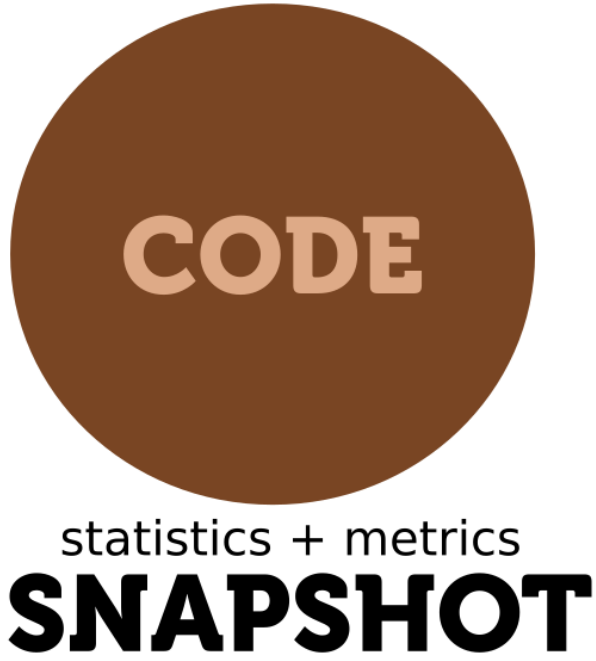
I hate reading
other people's code.

It's life, Jim, but not as we know it!

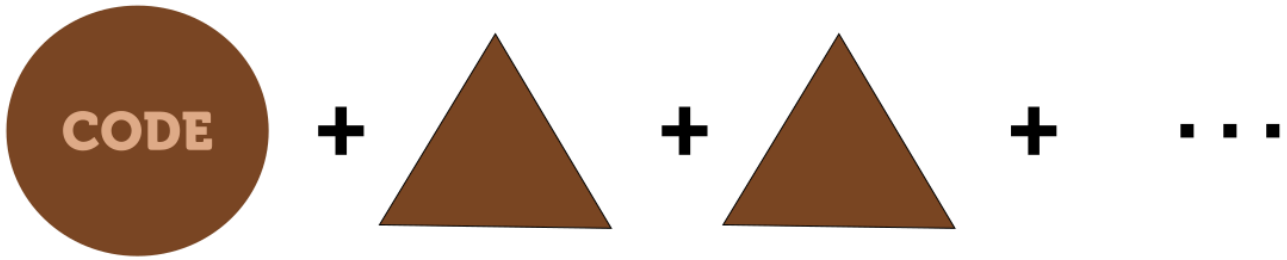
Code is data!


Well, big data

Snapshot



Temporal



HISTORY

Who uses SonarQube?

SonarQube

- Don't get me wrong: I love it!
- BUT... it's not that easy to get data in.
- It needs to be tuned for each team.
- It's not easy to make your team to use the data.
- Data is impersonal.

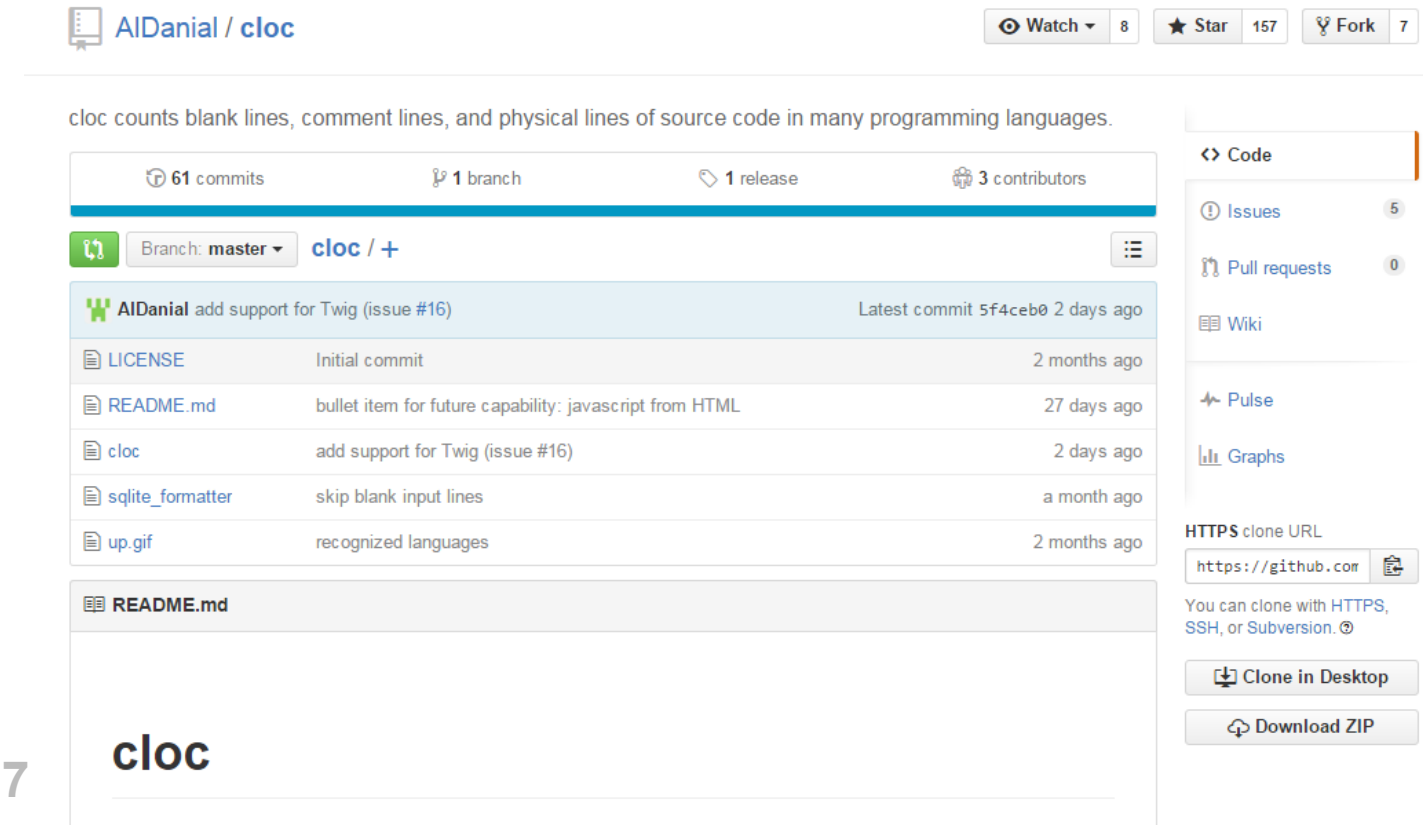
First steps

Count the lines!

Size does matter

- Gives you an estimate (on how much reading is needed).
- Most of the code bases are **polyglot**. Ratio between languages can tell something.
- Ratio between test code, comments, blank lines is also interesting.

cloc


AIDanial / cloc
Watch 8
Star 157
Fork 7
cloc counts blank lines, comment lines, and physical lines of source code in many programming languages.
61 commits
1 branch
1 release
3 contributors
Branch: master
cloc / +
AIDanial add support for Twig (issue #16)
Latest commit 5f4ceb0 2 days ago
LICENSE Initial commit 2 months ago
README.md bullet item for future capability: javascript from HTML 27 days ago
cloc add support for Twig (issue #16) 2 days ago
sqlite_formatter skip blank input lines a month ago
up.gif recognized languages 2 months ago
README.md
cloc
Code
Issues 5
Pull requests 0
Wiki
Pulse
Graphs
HTTPS clone URL
https://github.com
You can clone with HTTPS, SSH, or Subversion.
Clone in Desktop
Download ZIP

Usage

```
01. cloc --help
02. cloc --write-lang-def=lang.defs
```

Usage

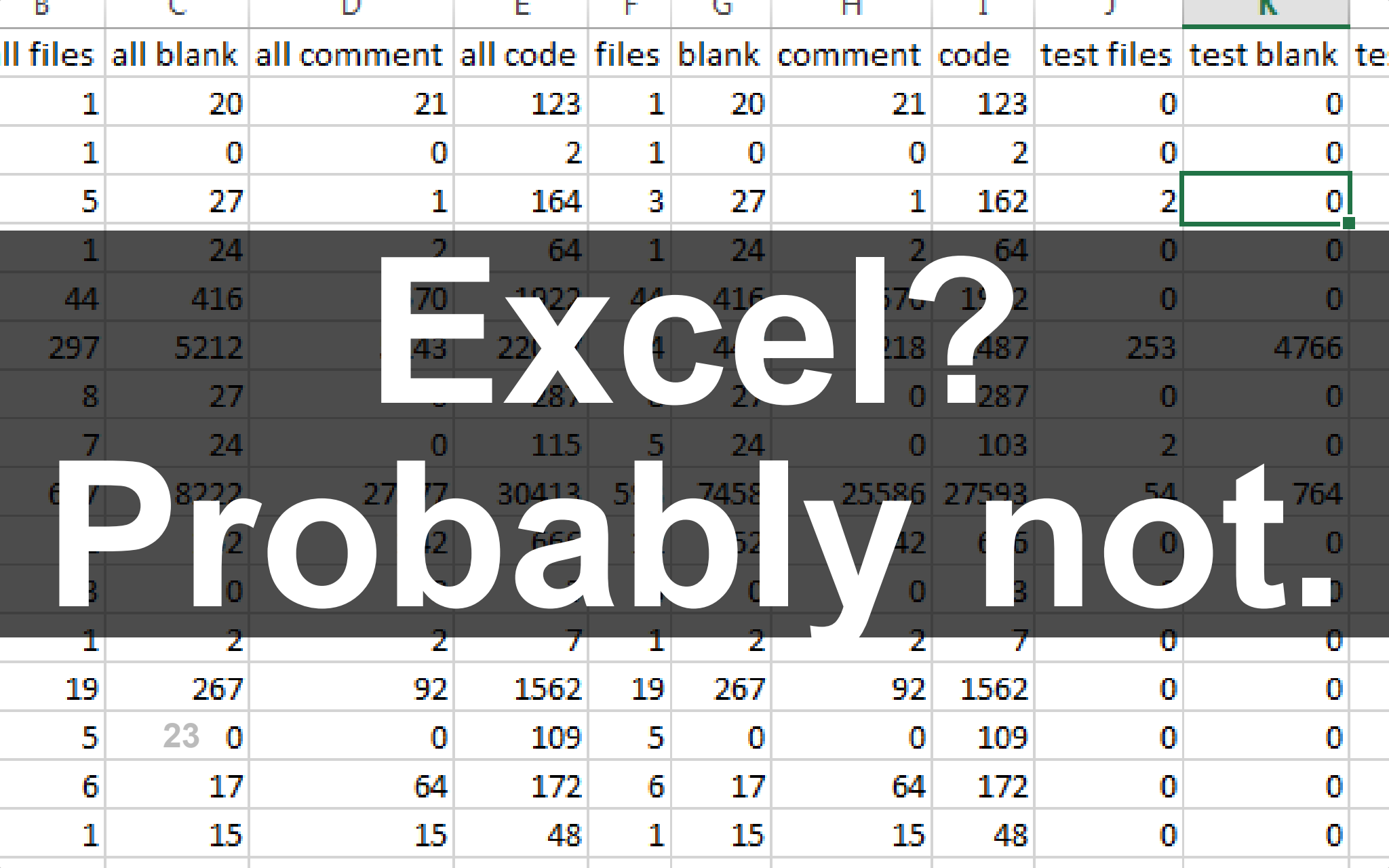
```
cloc --csv
     --quite
     --progress-rate=0
     --ignored=files.ignored
     --exclude-dir=test,build
     --read-lang-def=lang.defs
     --out=data.csv
     .
```

Language definitions

```
Gradle
    filter remove_matches ^\s*//
    filter remove_inline //.*$
    filter call_regexp_common C
    extension gradle
    3rd_gen_scale 4.10
```

Where are the pictures?

We have stats, let's plot them!

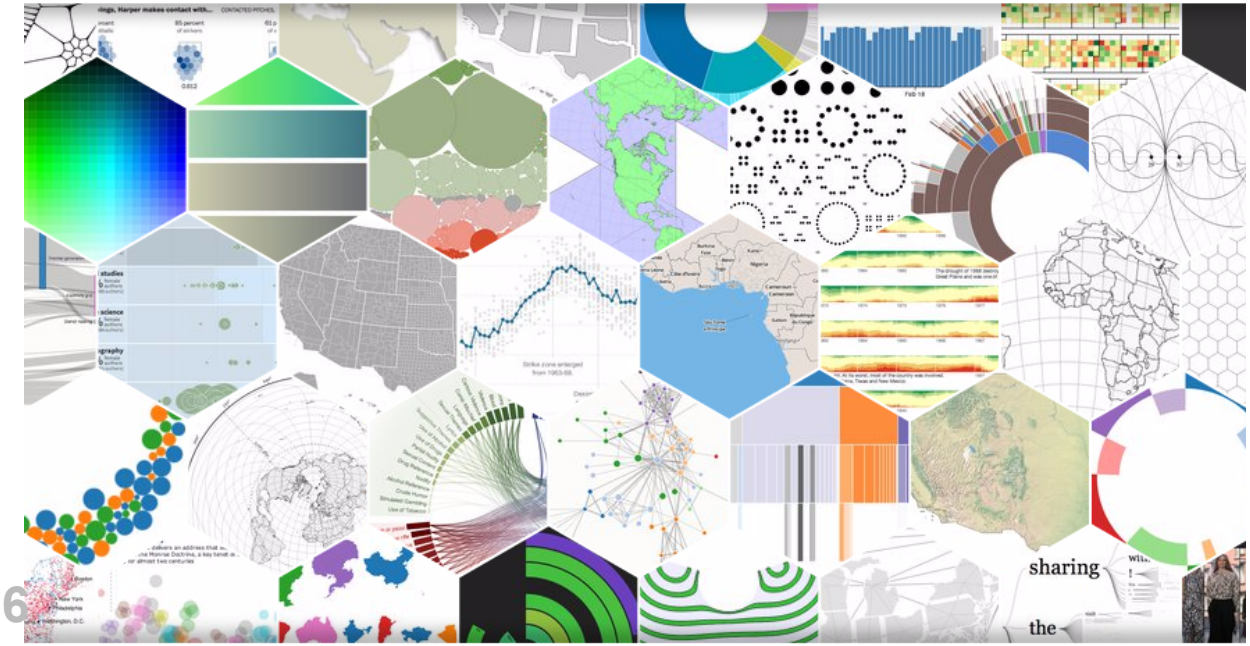

Excel?
Probably not.

Pie charts are boring!

Infographics!

d3.js

Data-Driven Documents

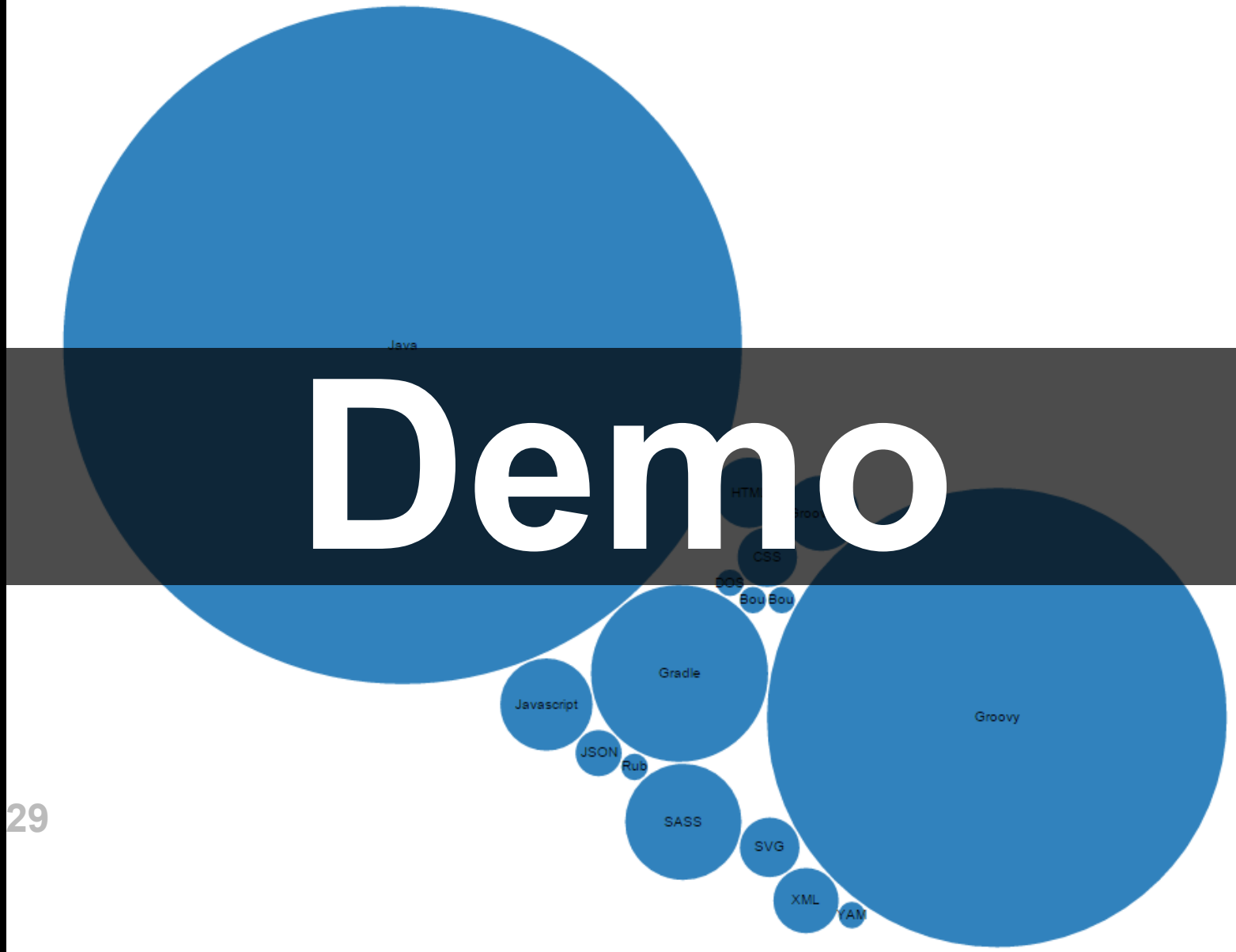


d3.js

- Javascript library for data visualizations
- Tons of examples
- Many libraries built on top of d3.js
- Several books

Useful charts

- Calendar View - https://bl.ocks.org/mbostock/4063318
- Buble Chart - https://bl.ocks.org/mbostock/4063269
- Hierarchical Edge Bundling - https://bl.ocks.org/mbostock/1044242


Java
HTM
CSS
Bou
Bou
Gradle
Javascript
JSON
Rub
Groovy
SASS
SVG
XML
YAM

Demo

Many alternatives

- vis.js
- raphael.js
- sigma.js
- many, many, many more

Tableau Public

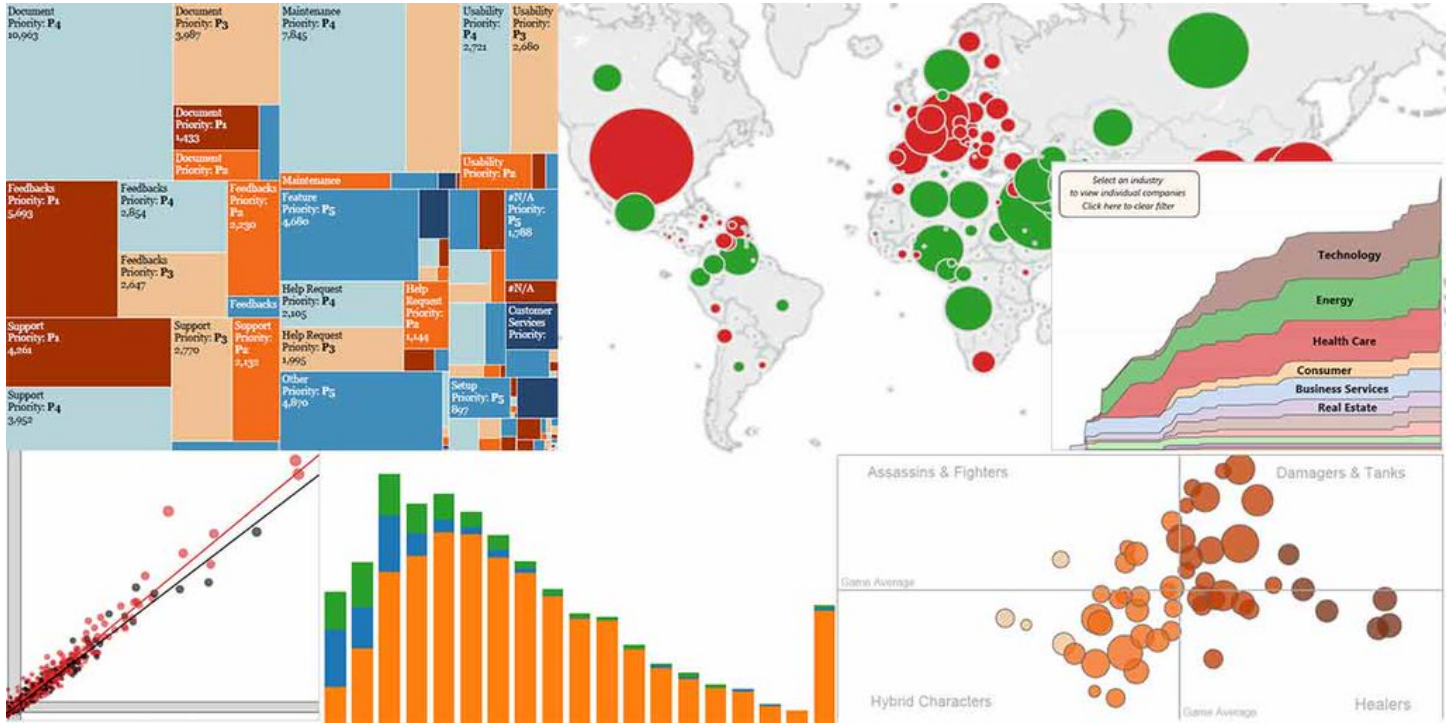


Tableau Public

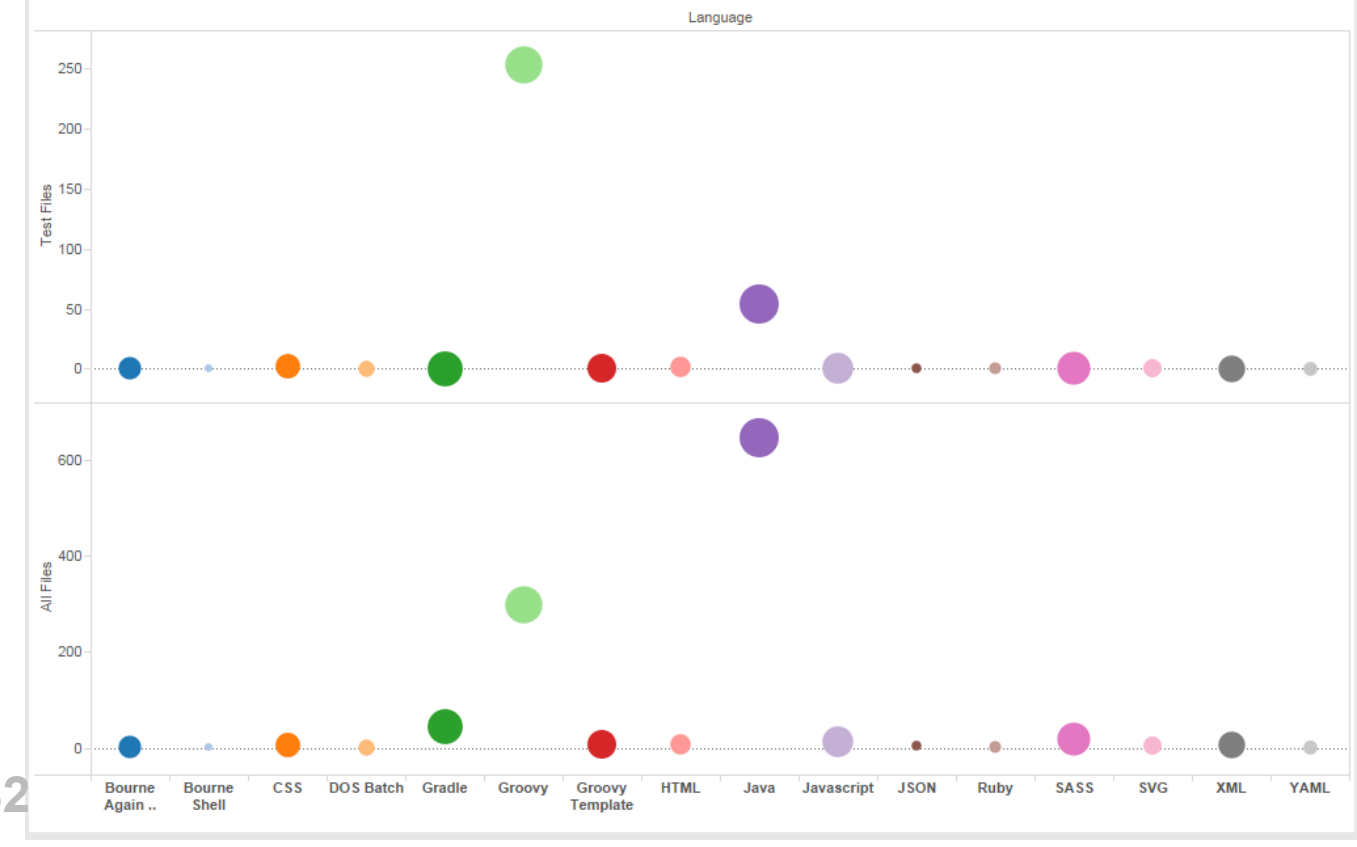
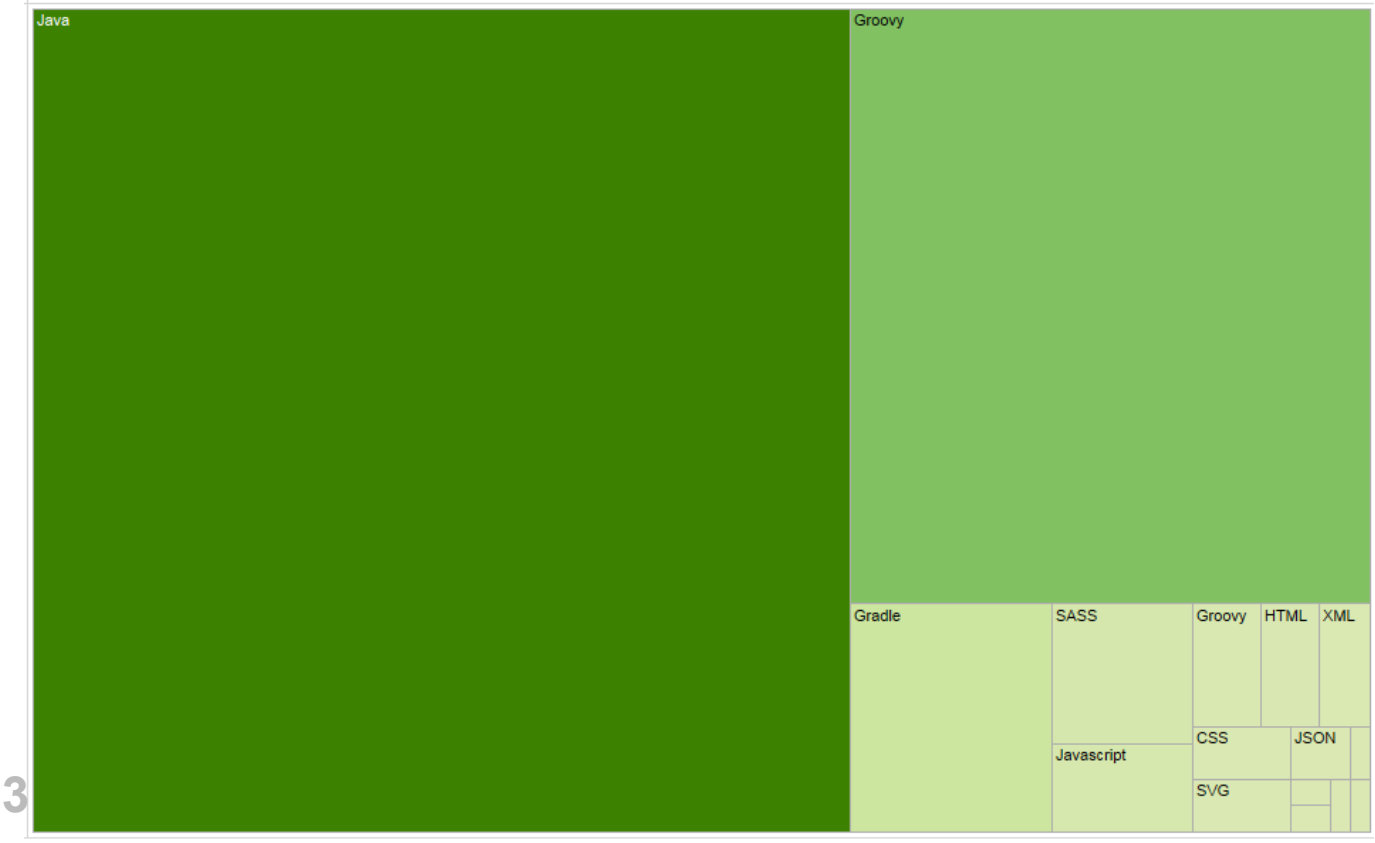


Tableau Public

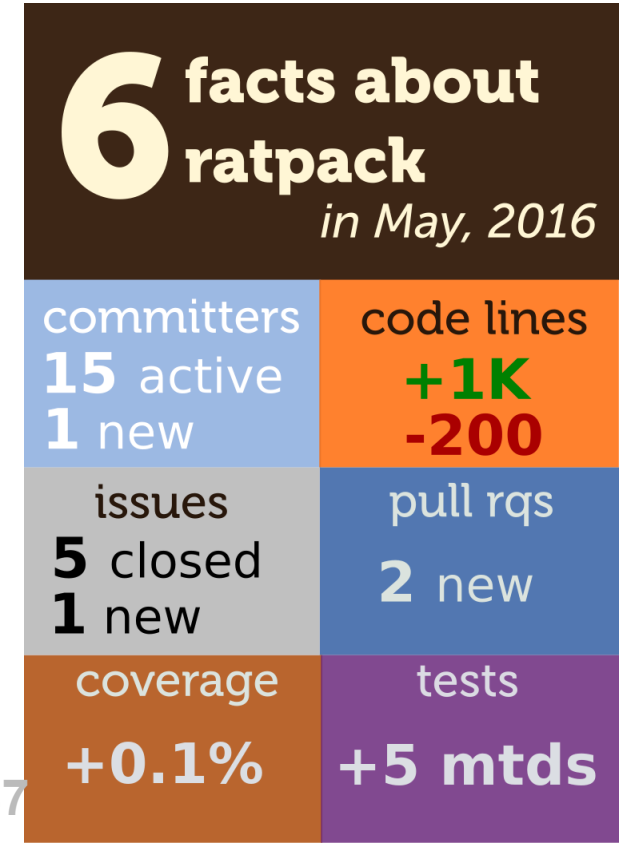


Demo

Design your own!

SVG + Inkscape

Home-made Inforgraphics



Code City

Code City

Richard Wettel

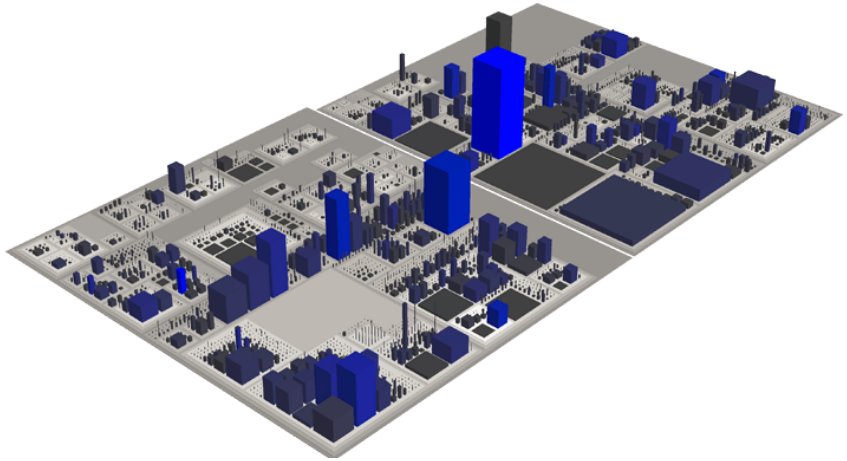
Home | Publications | Research | CodeCity | DuDe | Activities | Contact

Description | Experiment | Download | Tutorials | Movies | FAQ | Releases | Wall of Fame

Welcome to CodeCity!

CodeCity is an integrated environment for software analysis, in which software systems are visualized as interactive, navigable 3D cities. The classes are represented as buildings in the city, while the packages are depicted as the districts in which the buildings reside. The visible properties of the city artifacts depict a set of chosen software metrics, as in the polymetric views of CodeCrawler.

You can read more about my approach in the ICPC 2007, VISSOFT 2007, Softvis 2008, WASDett 2008, WCRE 2008, FAMOOSr 2008 papers and the ICSE 2008 tool demo on my **Publications** page. If you would rather prefer a hands-on learning process, **download** CodeCity, run it, and tell me what you think. I appreciate every piece of feedback I get. CodeCity is programmed in VisualWorks Smalltalk on top of the **Moose** platform, uses OpenGL for rendering, and runs on every major platform. To see CodeCity in action visit the **Wall of Fame**. Here is a sneak preview with a CodeCity visualization of JDK (Java Development Kit) v1.5:



Implementations

- eXcentia plugin for SonarQube
- JSCity

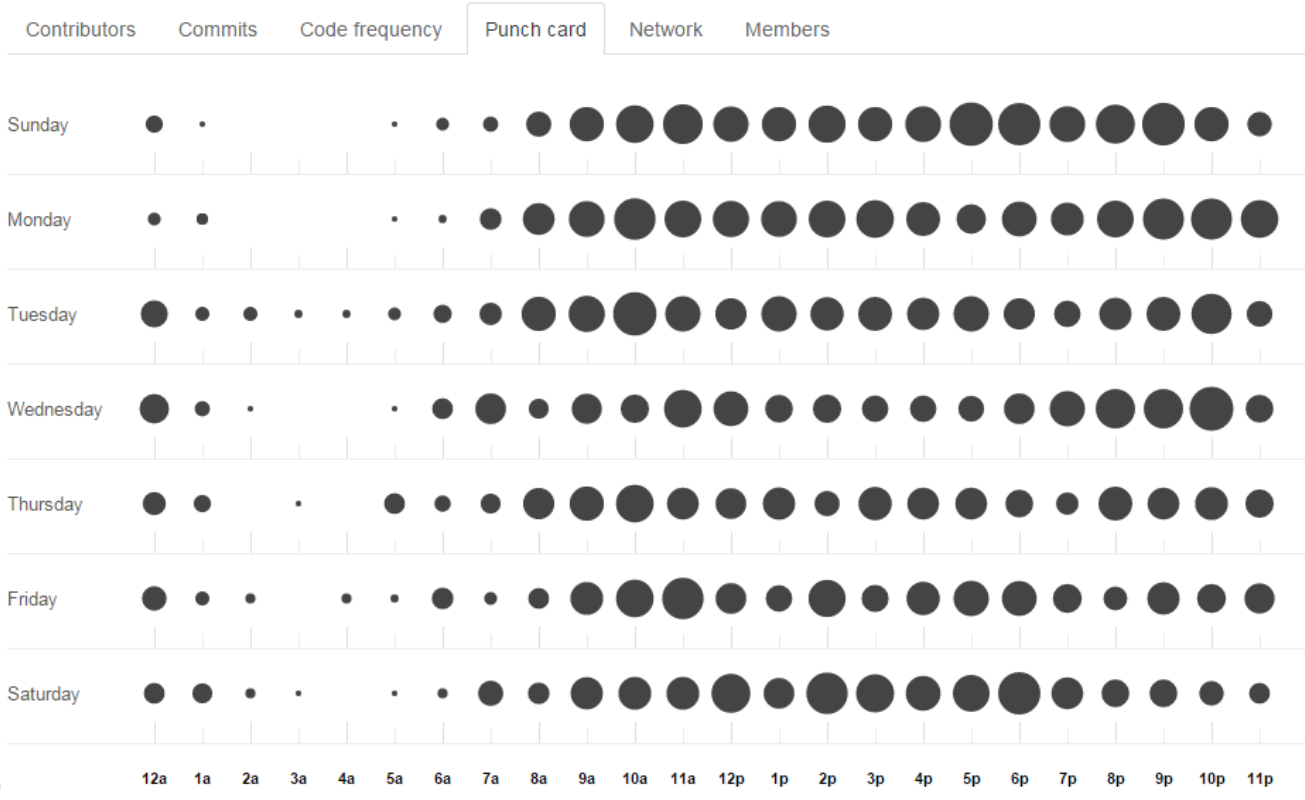
Demo

Temporal analysis

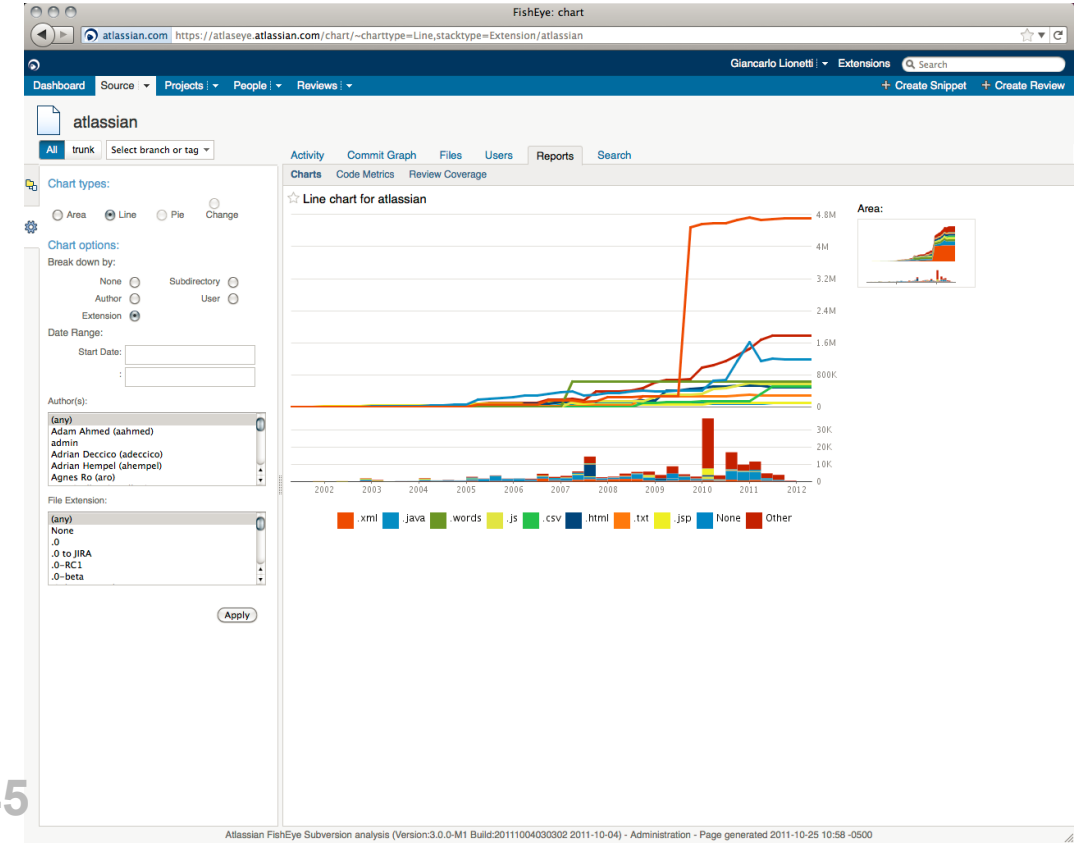
GitHub


Contributors
Commits
Code frequency
Punch card
Network
Members
Use ← and → to navigate
80
60
40
20
0
11/09
11/30
12/21
01/11
02/01
02/22
03/15
04/05
04/26
05/17
06/07
06/28
07/19
08/09
08/30
09/20
10/11
11/01

GitHub


Contributors
Commits
Code frequency
Punch card
Network
Members
Sunday
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday
12a 1a 2a 3a 4a 5a 6a 7a 8a 9a 10a 11a 12p 1p 2p 3p 4p 5p 6p 7p 8p 9p 10p 11p

FishEye


FishEye: chart
atlassian.com https://atlaseye.atlassian.com/chart/~charttype=Line,stacktype=Extension/atlassian
Giancarlo Lionetti
Extensions
Search
Dashboard
Source
Projects
People
Reviews
+ Create Snippet
+ Create Review
atlassian
All
trunk
Select branch or tag
Activity
Commit Graph
Files
Users
Reports
Search
Charts
Code Metrics
Review Coverage
Chart types:
Area
Line
Pie
Change
Chart options:
Break down by:
None
Subdirectory
Author
User
Extension
Date Range:
Start Date:
Author(s):
(any)
Adam Ahmed (aahmed)
admin
Adrian Deccico (adeccico)
Adrian Hempel (ahempel)
Agnes Ro (aro)
File Extension:
(any)
None
.0
.0 to JIRA
.0-RC1
.0-beta
Apply
Line chart for atlassian
Area:
4.8M
4M
3.2M
2.4M
1.6M
800K
0
30K
20K
10K
0
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
.xml
.java
.words
.js
.csv
.html
.txt
.jsp
None
Other
Atlassian FishEye Subversion analysis (Version:3.0.0-M1 Build:20111004030302 2011-10-04) - Administration - Page generated 2011-10-25 10:58 -0500

Gource

Software projects are displayed by Gource as an animated tree with the root directory of the project at its centre.

Directories appear as branches with files as leaves.

Developers can be seen working on the tree at the times they contributed to the project.

Launch gource

```
01. gource -s 0.1 -1280x720
02. gource --output-custom-log log1.txt
03. gource -s 0.1 -1280x720 log1.txt
```

Log format

```
1444125624|Luke Daley|M|/ratpack-core/.../WiretapPublisher.java
1444125624|Luke Daley|M|/ratpack-core/.../YieldingPublisher.java
1444306114|Stian Lindhom|M|/ratpack-manual/.../13-http.md
1444312172|Andrey Antukh|M|/ratpack-core/.../WebSocketEngine.java
```

Demo

Code Maat

Code Maat is a command line tool used to mine and analyze data from version-control systems

Launch Maat

```
01. git log --pretty=format:'[%h] %aN %ad %s' \
02.     --date=short --numstat --after=YYYY-MM-DD
```

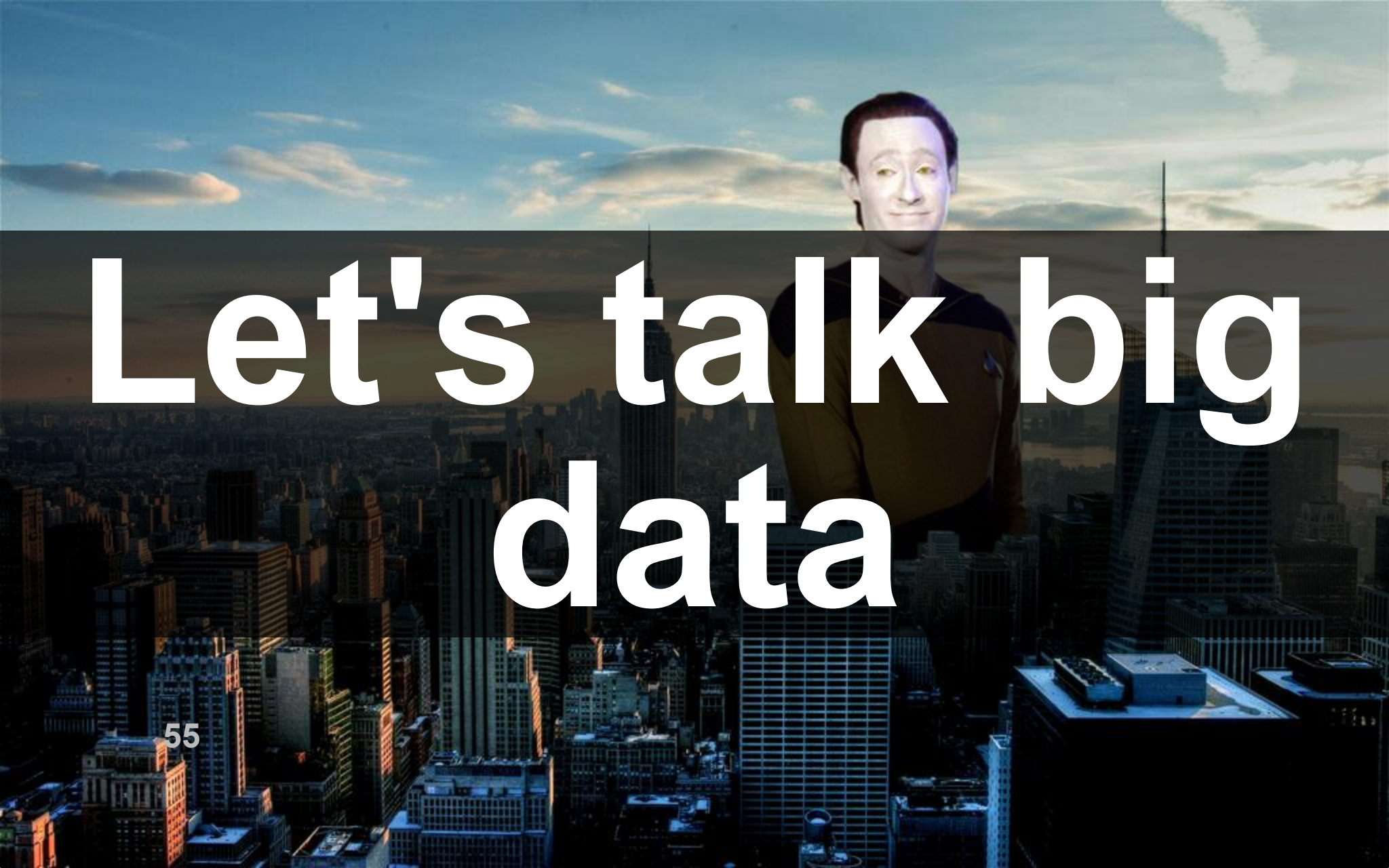
Launch Maat

```
01. maat -l git.log -c git -a summary
02. maat -l ratpack_evo.log -c git -a revisions > ratpack_freqs.csv
03. python scripts/merge_comp_freqs.py ratpack_freqs.csv ratpack_lines
```

Analysis types

abs-churn, age, author-churn, authors, communication, coupling, entity-churn, entity-effort, entity-ownership, fragmentation, identity, main-dev, main-dev-by-revs, messages, refactoring-main-dev, revisions, soc, summary

Demo


Let's talk big data

ElasticSearch

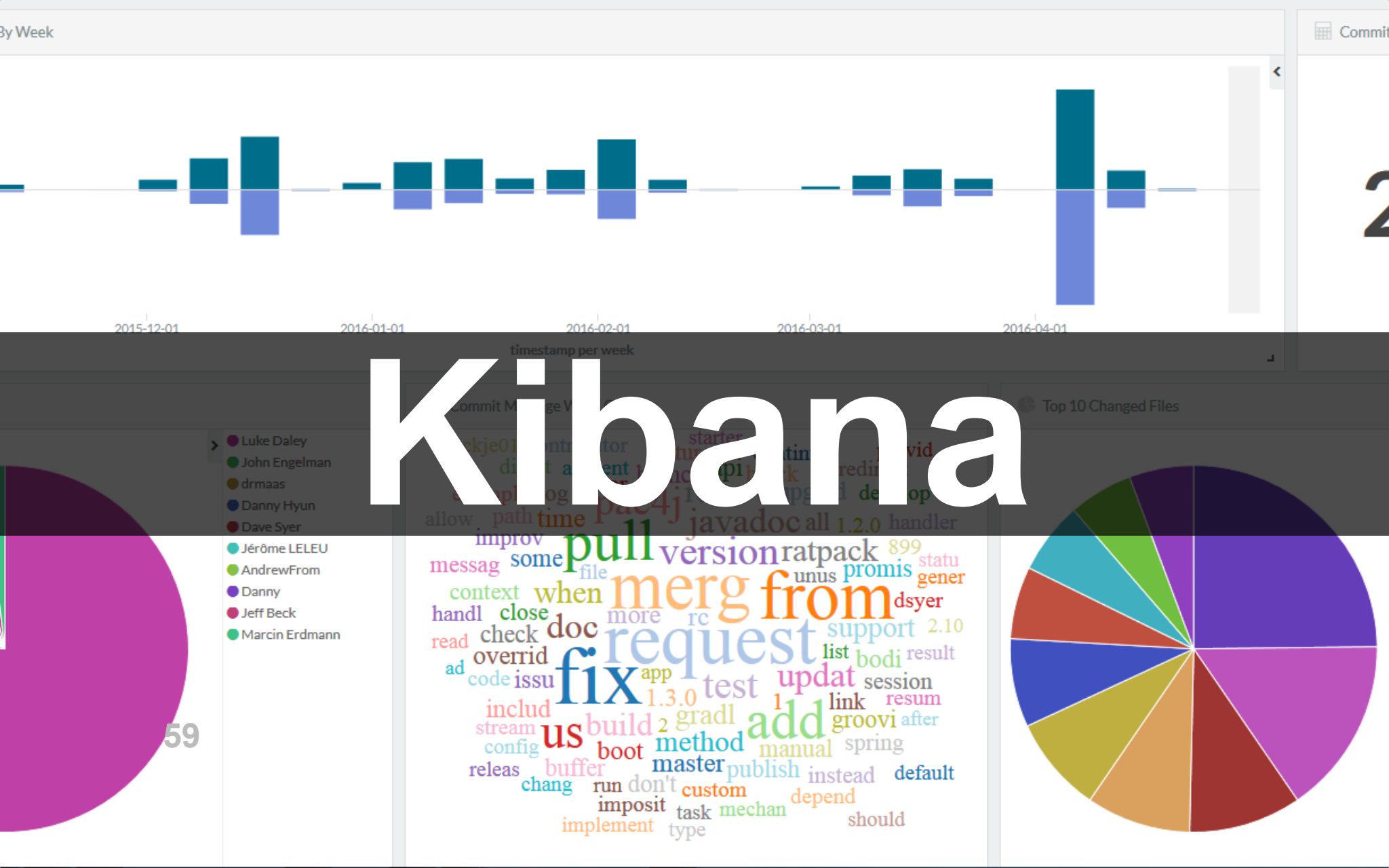
- Distributed, scalable, and highly available
- Real-time search and analytics capabilities
- Sophisticated RESTful API

Index data

```
01. $ curl -XPUT 'http://localhost:9200/gitlog/commit/123345' -d '{
02.   "commitId" : "123345",
03.   "timestamp" : "2009-11-15T14:12:12",
04.   "message" : "git into es"
05. }'
```

Kibana

- Flexible analytics and visualization platform
- Real-time summary and charting of streaming data
- Intuitive interface for a variety of users
- Instant sharing and embedding of dashboards

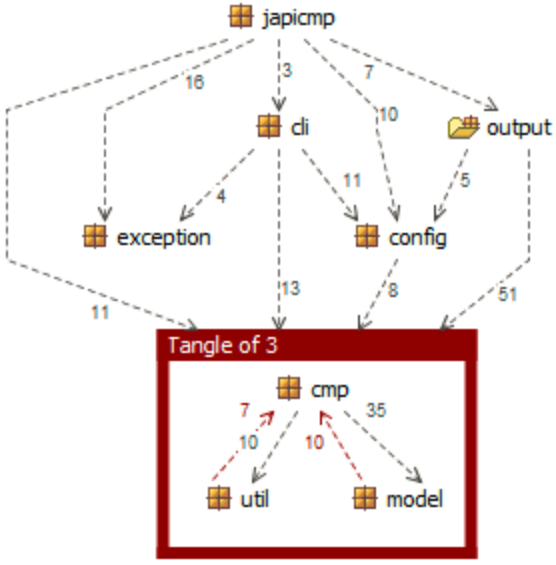

By Week
Commit
2015-12-01
2016-01-01
2016-02-01
2016-03-01
2016-04-01
timestamp per week
Top 10 Changed Files
Luke Daley
John Engelman
drmaas
Danny Hyun
Dave Syer
Jérôme LELEU
AndrewFrom
Danny
Jeff Beck
Marcin Erdmann
59
improv
pull
version
ratpack
javadoc
all
1.2.0
handler
messag
some
file
unus
promis
merg
from
dsyer
context
when
more
rc
handl
close
doc
request
support
2.10
read
check
overrid
fix
list
bodi
result
ad
code
issu
app
test
updat
session
1.3.0
link
resum
includ
gradl
add
groovi
after
stream
us
build
method
config
boot
manual
spring
releas
buffer
master
publish
instead
default
chang
run
don't
custom
depend
imposit
task
mechan
should
implement
type

Kibana

Demo

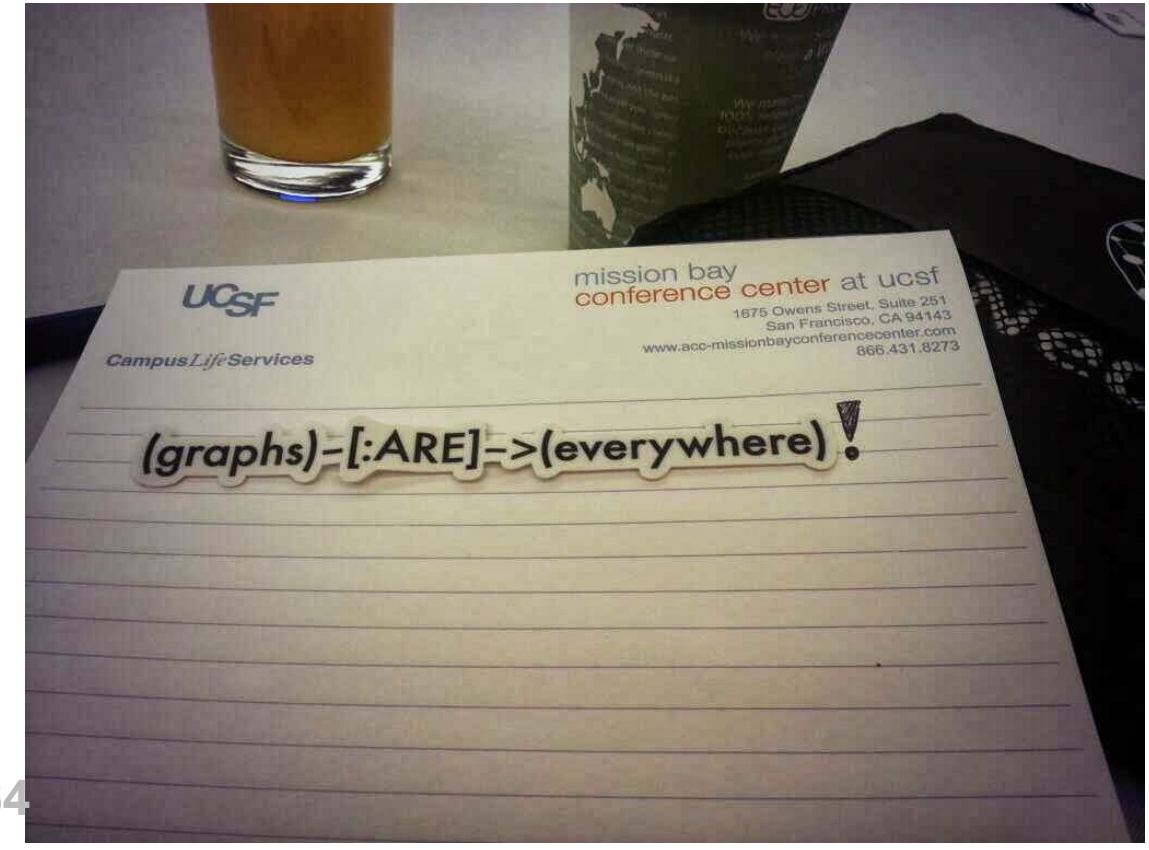
Structure

Structure101



Demo

Graphs are everywhere!



jQAssistant

jQAssistant is a QA tool which allows the definition and validation of project specific rules on a structural level.

It is built upon the graph database Neo4j and can easily be plugged into the build process to automate detection of constraint violations and generate reports about user defined concepts and metrics.

jQAssistant

```
01. jqassistant scan -f binaries
02. jqassistant server
```

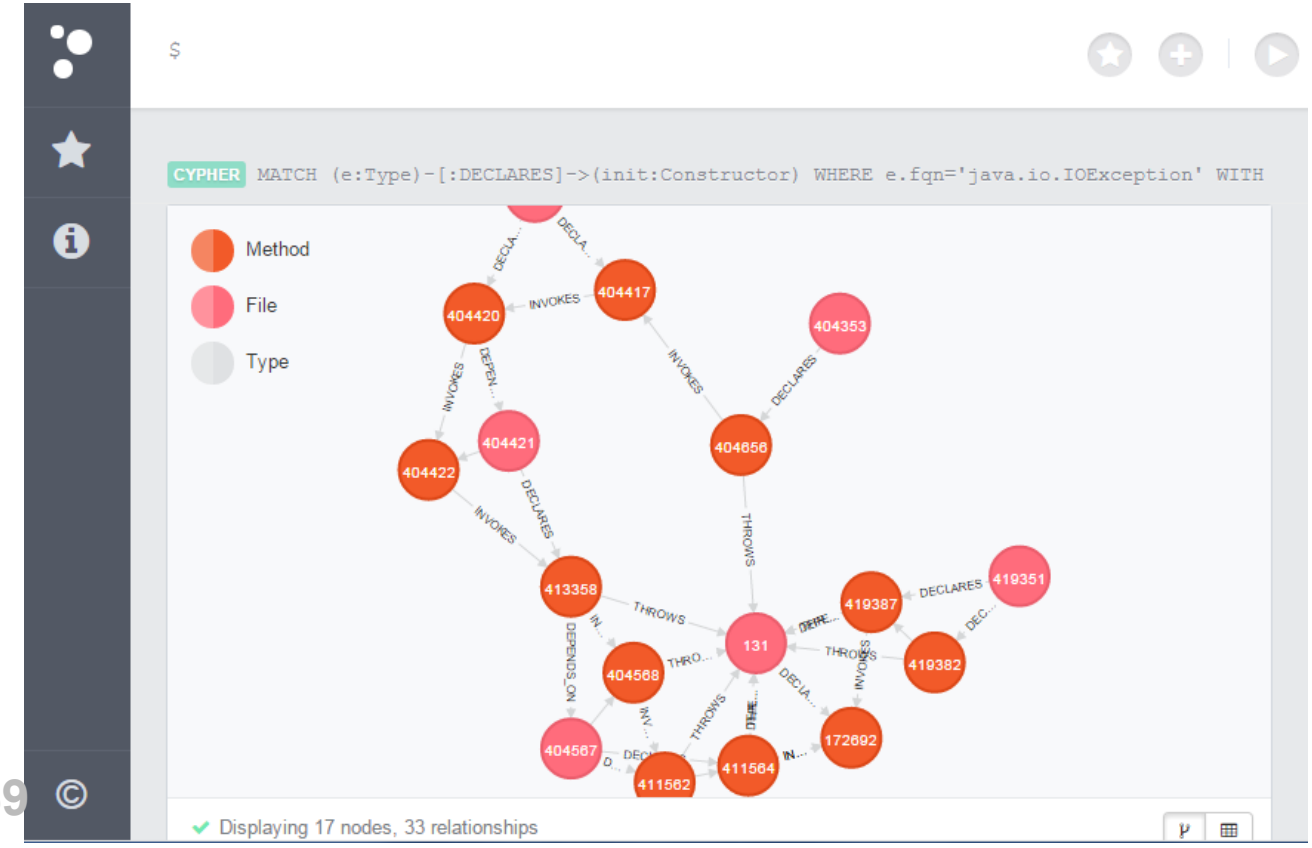
Query your code

```
01. MATCH
02.   (class:Class)-[:DECLARES]->(method:Method)
03. RETURN
04.   class.fqn, count(method) as Methods
05. ORDER BY
06.   Methods DESC
07. LIMIT 20
```

Query different versions

```
01. match
02.   (artifact:Artifact)-[:CONTAINS]->(type:Type)
03. return
04.   artifact.fileName as Artifact, collect(type.fqn) as Types
```

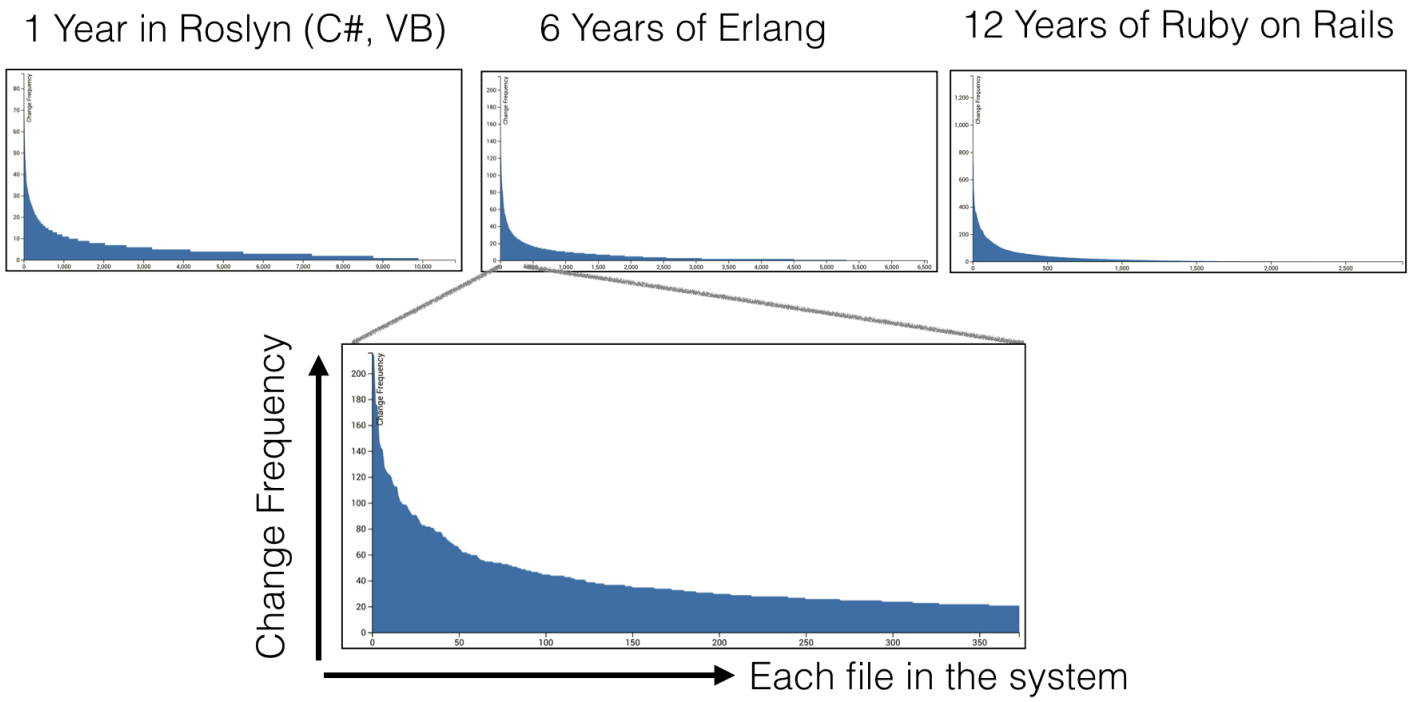
neo4j


$
CYPHER MATCH (e:Type)-[:DECLARES]->(init:Constructor) WHERE e.fqn='java.io.IOException' WITH
Method
File
Type
404417
404420
404353
404421
404422
404656
413358
131
404568
419387
419351
419382
404587
411564
411562
172692
INVOKES
DECLARES
THROWS
DEPENDS_ON
Displaying 17 nodes, 33 relationships

Demo

Conclusion

Adam Tornhill



Hotspot Analysis

- use hotspots to identify maintenance problems.
- use hotspots for risk management.
- hotspots point to code review candidates.
- hotspots are input to exploratory tests.

Conclusion

- Extract data from your code!
- Visualize it and search for hot spots!
- Search for new facts and knowledge!
- Become data scientist or data journalist!

Next time...

when you...

push your code

REMEMBER


IT'S
ALIVE!

Reading material

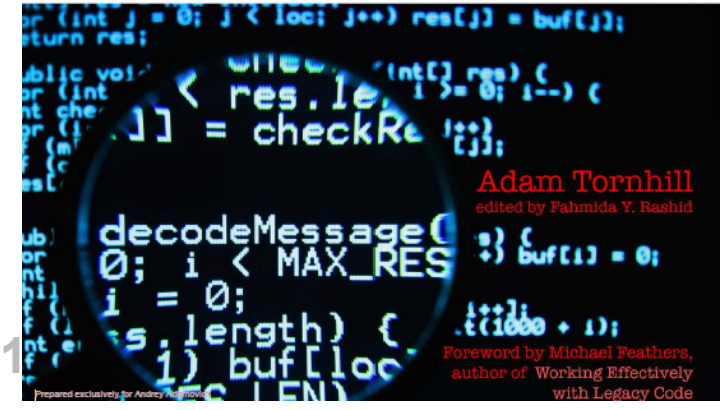
Your Code as a Crime Scene

The Pragmatic Programmers

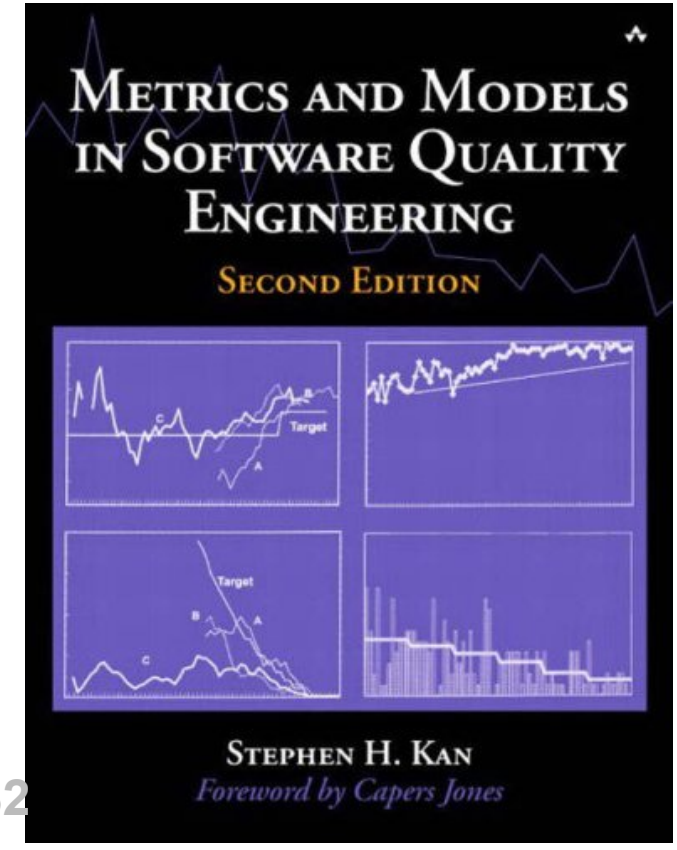
Investigator:
Date:
Case #:
Location:

Your Code As a Crime Scene

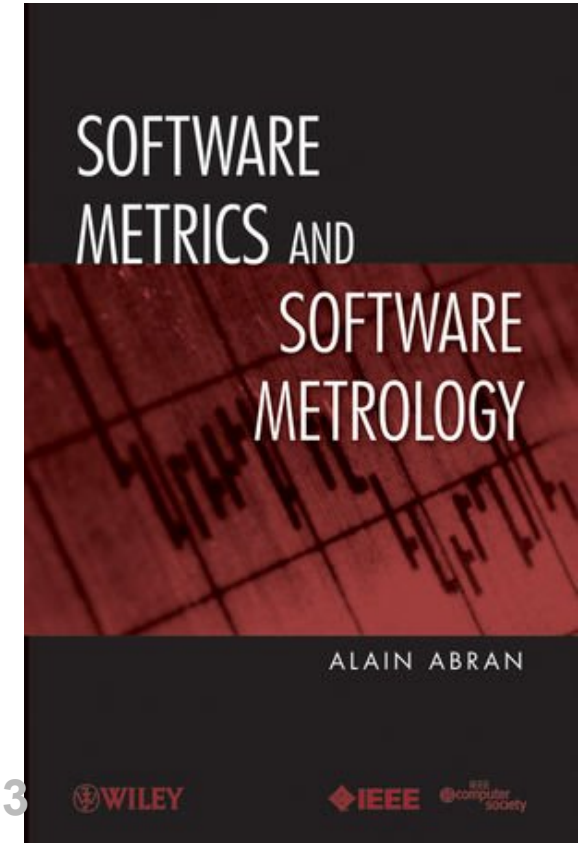
Use Forensic Techniques to Arrest Defects, Bottlenecks, and Bad Design in Your Programs



Metrics and Models in SQE



Software Metrics and Metrology



Links: images

- http://abstrusegoose.com/432
- http://camarenaphoto.tumblr.com/post/112238079516/its-life-jim-but-not-as-we-know-it-spock
- http://technology.ie/big-data-looks-like/
- http://www.informationisbeautiful.net/visualizations/million-lines-of-code/
- http://githut.info/
- http://emmanueloga.com/2013/10/07/Graphs-are-Everywhere-An-overview-of-GraphConnect-San-Francisco-2013.html

Links: tools

- https://github.com/AlDanial/cloc
- http://gource.io/
- http://wettel.github.io/codecity-wof.html
- https://github.com/adamtornhill/code-maat
- http://d3js.org/
- http://visjs.org/

Links: tools

- http://www.sonarqube.org/
- https://www.atlassian.com/pt/software/fisheye/overview
- https://www.elastic.co/products/elasticsearch
- https://www.elastic.co/products/kibana
- http://jqassistant.org/
- http://neo4j.com/

Links: tools

- https://github.com/ThoughtWorksStudios/saikuro_treemap
- https://www.youtube.com/watch?v=iiIytERhV9o
- https://codescene.io
- http://www.adamtornhill.com/articles/software-revolution/part1/index.html

That's all!

Thank you!

$ ping me


@codingandrey


github.com/aadamovich


lv.linkedin.com/in/andreyadamovich


lanyrd.com/profile/andrey-adamovich

andrey@aestasit.com