06.11.2018
MÜNCHEN, W-JAX 2018

Die Best Practices der Data Scientists

#Big Data & Machine Learning

#Software Architecture

Markus Harrer
Software Development Analyst

INNOQ

I ♥ legacy code!

Markus Harrer

Software Development Analyst

INNOQ

Blog: feststelltaste.de

Twitter: @feststelltaste

Folien: speakerdeck.com/feststelltaste

Data Science

Was ist ein Data Scientist?

"... einer, der Statistik am Mac macht."

"...eine Statistikerin, die in San Francisco lebt."

"...ist besser in Statistik als ein Entwickler und besser in Softwareentwicklung als ein Statistiker."

übersetzt von https://twitter.com/cdixon/status/428914681911070720

Worum geht es im „data“?

“Without **data** you’re just another person with an **opinion**.”

W. Edwards Deming

=> Belastbare Erkenntnisse mittels Fakten liefern

Worum geht es im „science“?

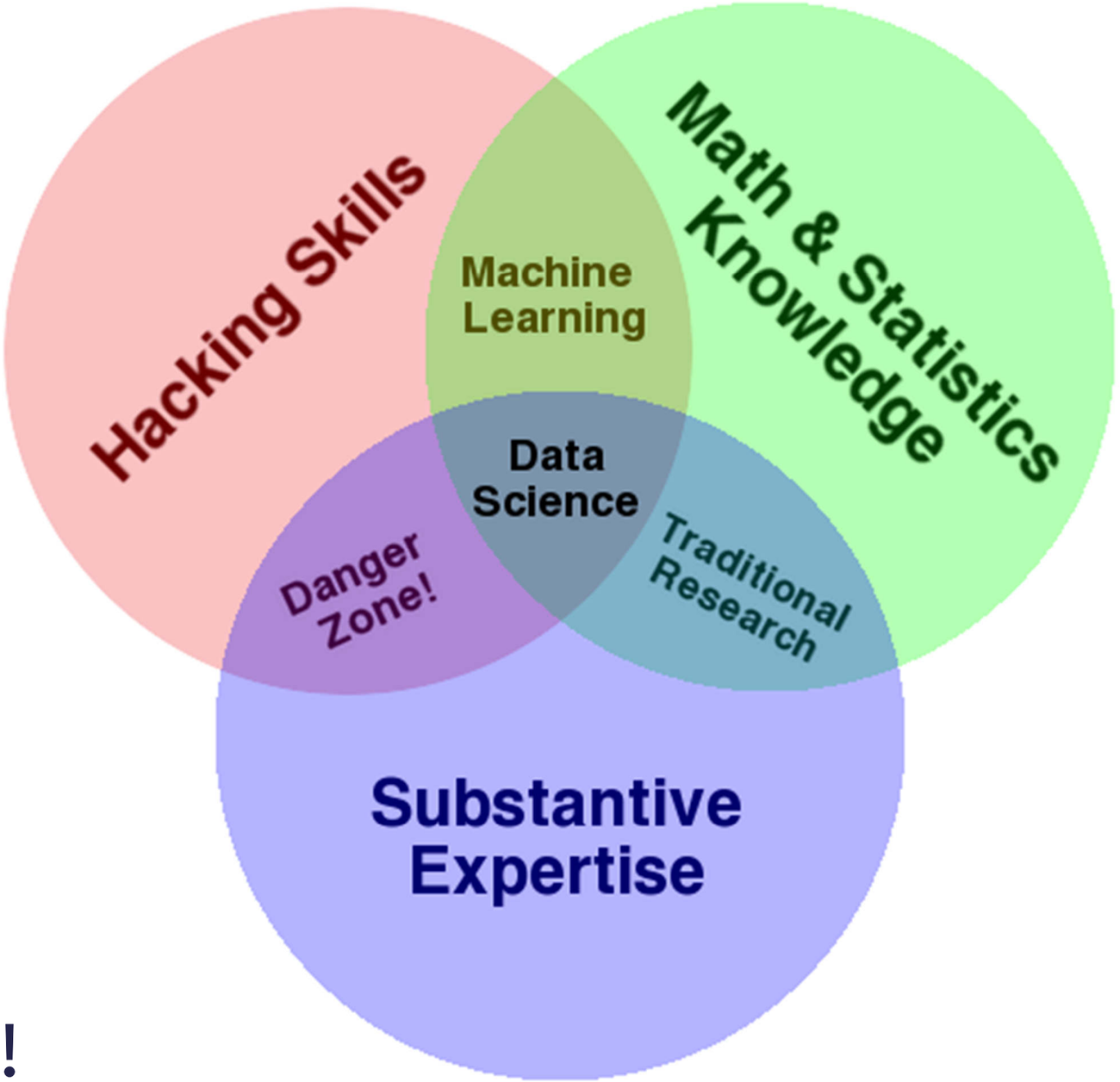
“The aim of science is to seek the simplest explanations of complex facts.”

Albert Einstein

=> Neue Erkenntnisse verständlich herausarbeiten

Data Science

Venn Diagramm von
Drew Conway



== Softwareentwicklung!

Einsatz von Data Science

in der Softwareentwicklung

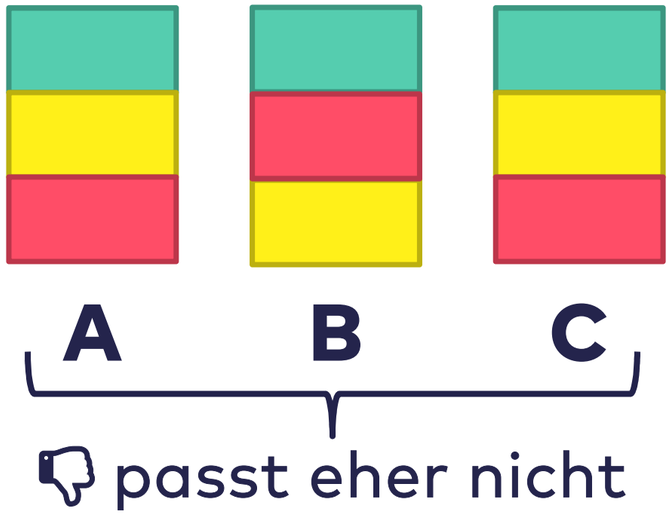
- Wo verletzen wir Architekturvorgaben?
- Weshalb schlagen unsere CI-Builds sporadisch fehl?
- Woher kommen die vielen DB-Calls?
- Welcher Entwickler kennt sich wo im Code aus?
- Wo gibt es sich gegenseitig überschreibenden Zustand?
- Ist unsere Software richtig geschnitten?
- ...

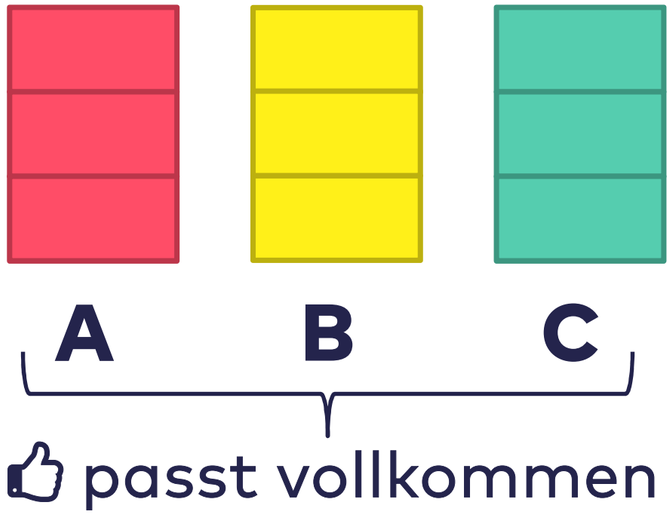
Ein Praxisbeispiel

https://www.feststelltaste.de/checking-the-modularization-of-software-systems-by-analyzing-co-changing-source-code-files/

Beispiel: Fragestellung

„Wie gut passt der fachliche Schnitt zur Entwicklungsaktivität?“





Legende

fachliches Modul

Entwicklungsaktivität

Beispiel: Idee

Heuristik

„Werden Änderungen innerhalb einer Komponente zusammengehörig vorgenommen?“

- Änderungen => Commits aus Versionsverwaltung
- Komponenten => Teil von Dateipfad

Beispiel: Daten

Commit und Dateipfad

```
git log --numstat --format=...
```

commit_id	filepath
#59a26	.../todo/Get.java
#59a26	.../todo/New.java
#34af9	.../site/Main.java
...	...

Beispiel: Analyse

Pivot-Tabelle mit Commits für jede Datei

	#59a26	#35e25	#34af9	…
…/todo/Get.java	1	1	0	…
…/todo/New.java	1	1	0	…
…/site/Main.java	0	0	1	…
…	…	…	…	…

=> pro Datei ein Vektor (=> reine Mathematik)

Beispiel: Modell

Ähnlichkeitsberechnung

=> Cosinus-Ähnlichkeit zwischen Vektoren / Dateien

	.../todo/Get.java	.../todo/New.java	.../site/Main.java	...
.../todo/Get.java	1	0.8	0.3	...
.../todo/New.java	0.8	1	0	...
.../site/Main.java	0.3	0	1	...
...	...	...	...	...

Beispiel: Visualisierung (1/2)

Informationsreduzierung

- Multidimensionale Skalierung reduziert n Dimensionen auf zwei Dimensionen unter Beibehaltung der Abstände
- Aus Dateipfad lässt sich Komponente extrahieren
  - Beispiel: `.../todo/Get.java => todo`

filepath	x	y	komp
.../todo/Get.java	0.14	0.67	todo
.../todo/New.java	0.13	0.70	todo
.../site/Main.java	0.31	0.50	site
...	...	...	...

Beispiel: Visualisierung (2/2)

Erzeugung interaktiver Grafik

- Dateien des Softwaresystems

  => Punkte

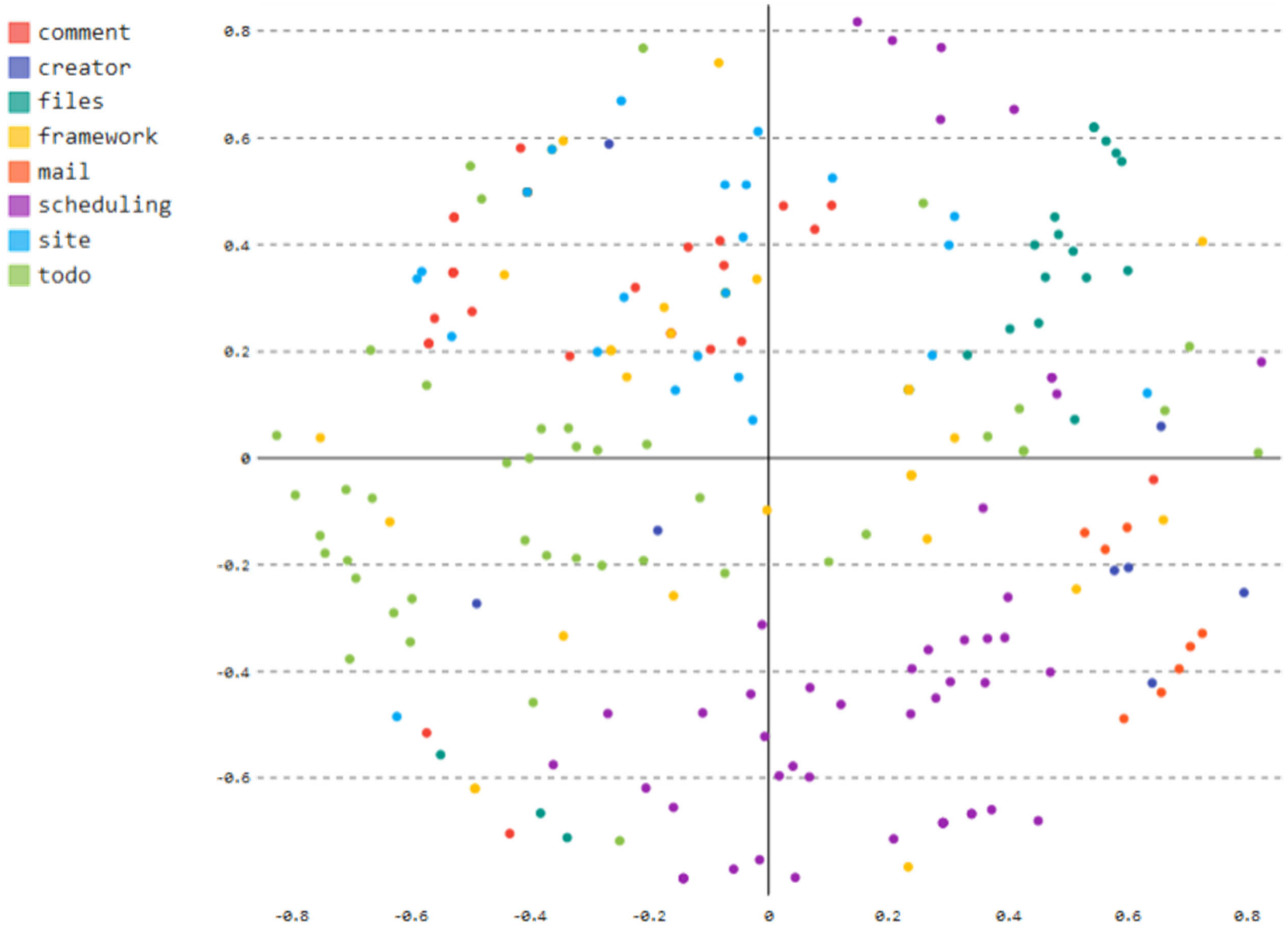
- Dateien, die gemeinsam geändert werden

  => Nähe der Punkte zueinander

- Komponenten des Softwaresystems

  => Farben der Punkte

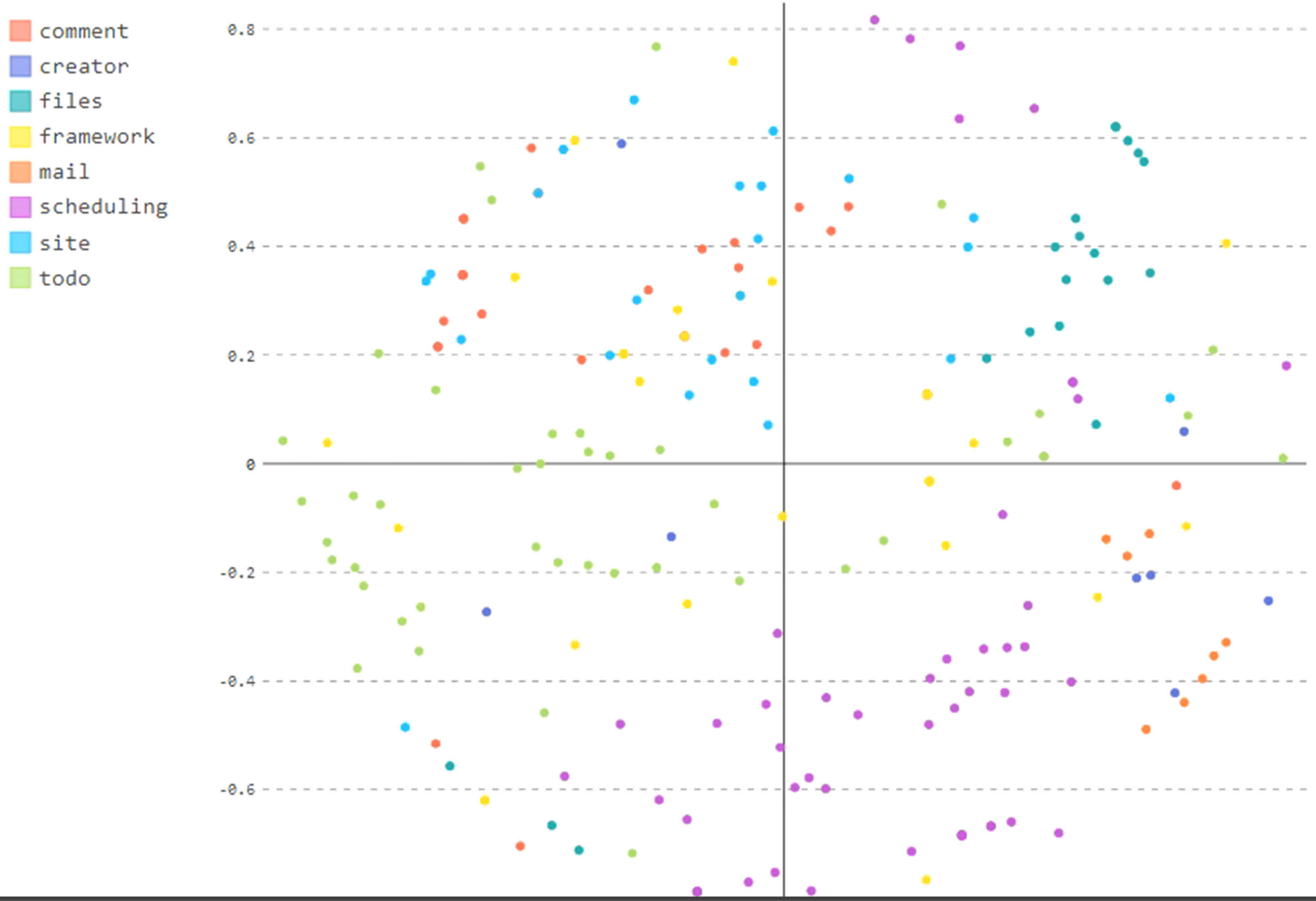
Beispiel: Erkenntnis



Demo

„Der Schnitt-Check"

https://feststelltaste.github.io/software-analytics/notebooks/vis/checking_modularization/dropover.html



https://www.youtube.com/watch?v=XAxseWdu5YA

„Sehr schön!“

„Aber ein paar Schritte zurück!"

„Wie geht sowas?“

Markus Harrer
@feststelltaste

Antwort an @rotnroll666

Thanks for the kind words! But I'm just hacking my way through all the data with pure stubbornness 😇

Tweet übersetzen

10:25 - 20. Apr. 2017

1 „Gefällt mir"-Angabe

BEST PRACTICES

Data Science on Software Data

Problemfixierung

„Wenn ich eine Stunde habe,
um ein Problem zu lösen,
dann beschäftige ich mich
55 Minuten mit dem Problem
und 5 Minuten mit der Lösung."

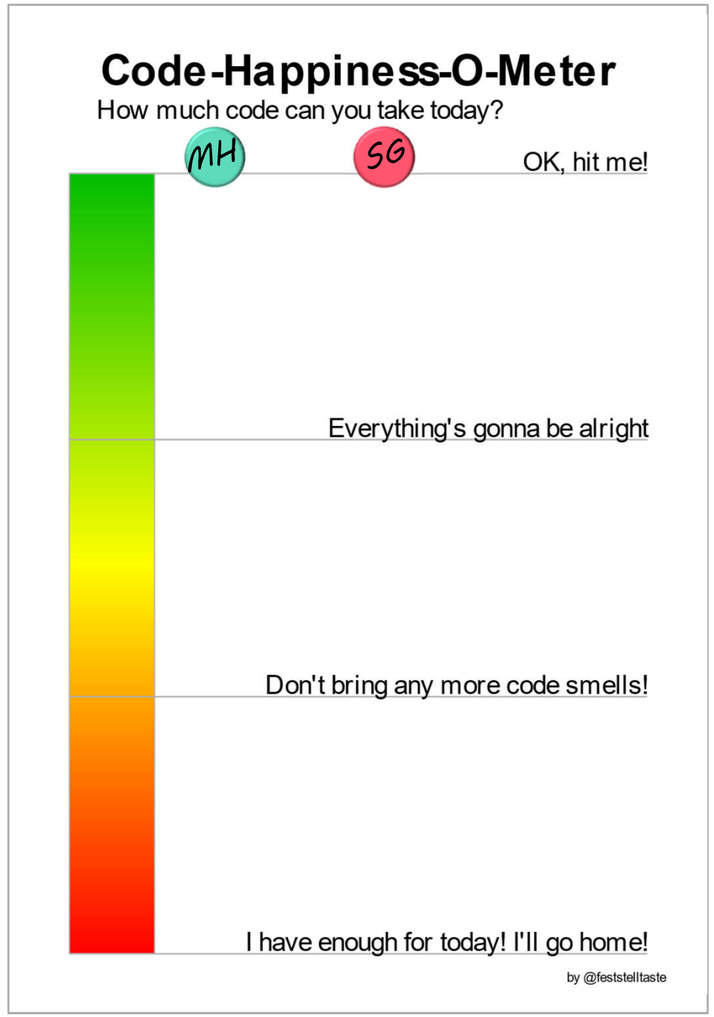
Albert Einstein

0. Fragestellung

Wie komme ich auf ein Problem?

Problemidentifikation

- Kneipe
- Retrospektiven / Pre-Mortems
- Code-Happiness-O-Meter



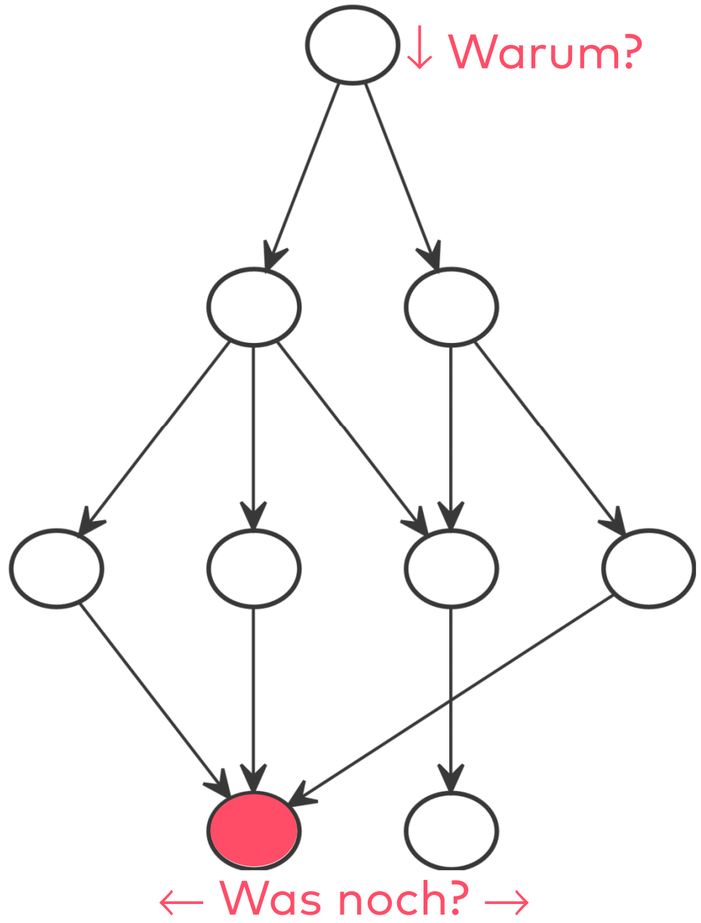
https://bit.ly/2PbJiPM

Wie komme ich auf die Ursache?

Ursachenanalyse

- Lösungsreflex unterdrücken
- Zuhören, nachfragen, ergründen
  - Tiefensuche mit „Warum?“
  - Breitensuche mit „Was noch?“
- Hotspots vornehmen

=> Auswirkungen quantifizieren



1. Idee

Datenzentriertes Denken

Frage stellen

- „Woran erkenne ich, dass sich etwas geändert hat?“

=> Vorhandenen Datenquellen bewusster wahrnehmen

2. Daten

Softwaredaten und Metadaten

statisch

Laufzeit

Community

chronologisch

Eigenheiten der Datenarten

zumindest der meisten...

Geschäftsdaten

- ✗ Lückenhaft
- ✗ Qualität durchwachsen
- ✗ Durcheinander
- ✓ Einfache Struktur
- ✓ Direkt
- ✓ Wenig detailliert

Softwaredaten

- ✓ Vollständig
- ✓ Qualität gut
- ✓ Formatiert
- ✗ Komplexe Strukturen
- ✗ Indirekt
- ✗ Sehr feingranular

Best Practices im Umgang mit Daten

Geschäftsdaten

- Daten aufwendig säubern
- Komplexere Modelle nutzen
- Unsicherheit offenlegen

Softwaredaten

- Datenherkunft angeben
- Heuristiken nutzen
- Annahmen offenlegen

Datenlieferanten intelligent nutzen

Ausgaben richtig formatieren

- `cloc ./ --quiet --csv`

Grundlegende Vorverarbeitung auf Werkzeug legen

- `git log --no-merges --no-renames --since=...`

Vorhandene Datenquellen nutzen

- Jenkins Remote Access API (Build-Logs, Test-Reports, ...)
- SonarQube Web API (Code Smells, Metriken, ...)

Best Practices für tabulare Daten

Single Responsibility Principles

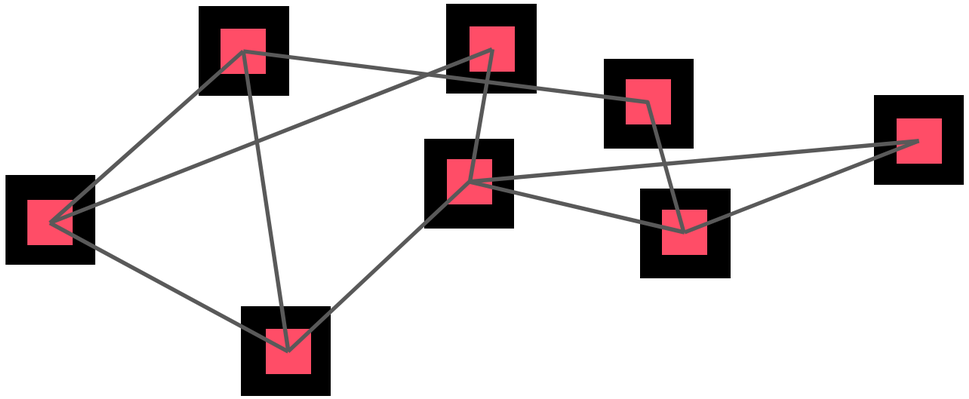
- Pro Variable eine Spalte
- Für jede Beobachtung eine Reihe
- Für alle zusammengehörigen Variablen eine Tabelle
- Für jede Tabelle einer Analyse eine verlinkende Spalte

Jeff Leek: The Elements of Data Analytic Style

Best Practices für komplexe Daten

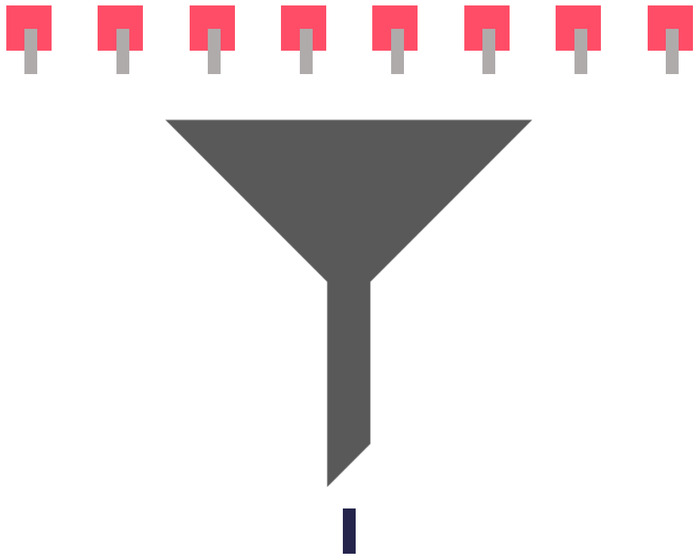
Vernetzte Strukturen

- Auf einen Aspekt festlegen
- Extrem denormalisieren



Vielfältige, feingranular Daten

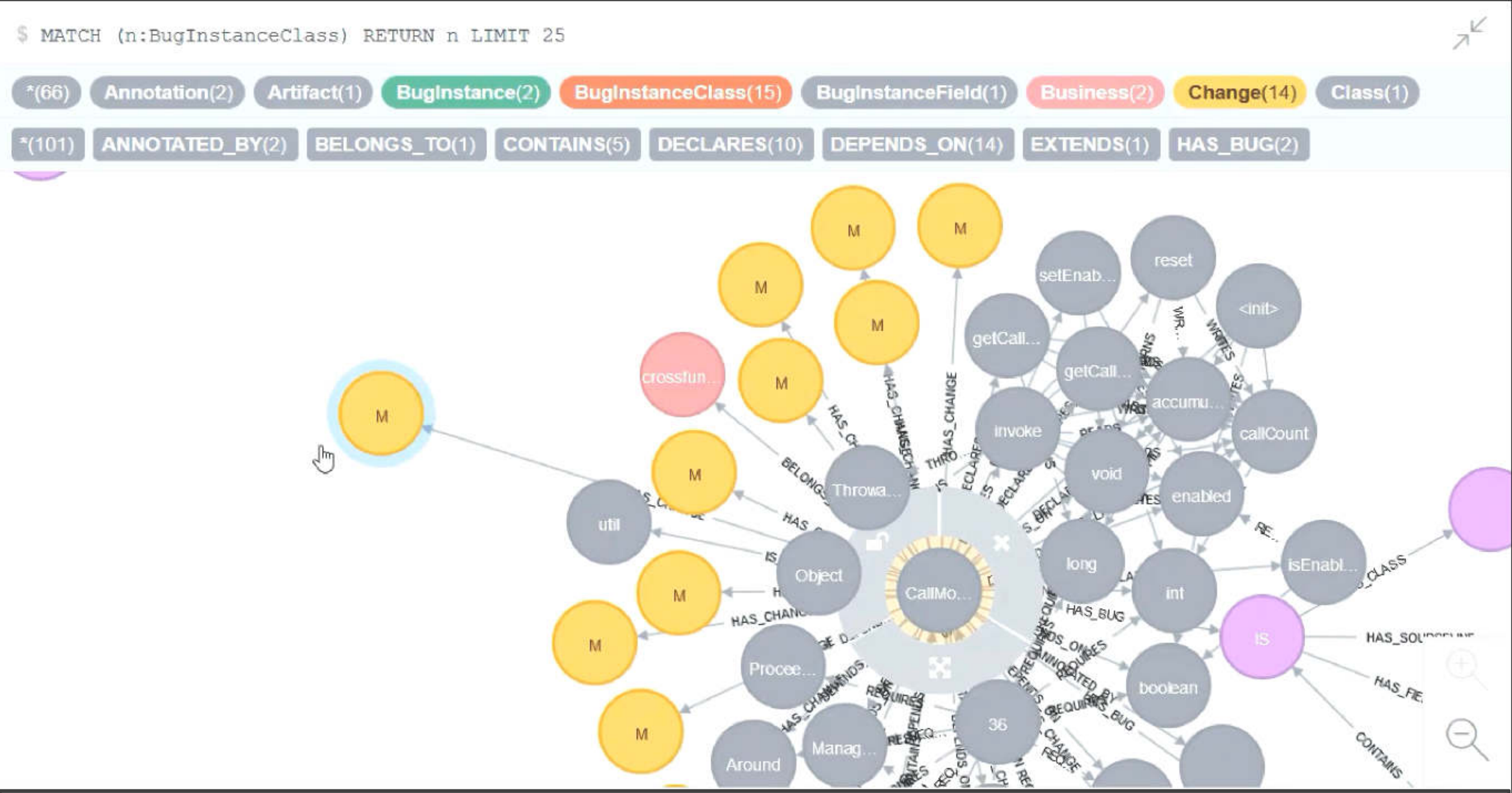
- Verschneiden (=joinen)
- Filtern (=reduzieren)
- Aggregieren (=verdichten)



Demo

„Code Smells Analyse mit jQAssistant/Neo4j"

Beispiel-Notebook: https://www.feststelltaste.de/race-condition-demo-notebook/



https://www.youtube.com/watch?v=SOREr0QPT4I

Weitere Details zu jQAssistant/Neo4j

https://easychair.org/publications/preprint/893N

Towards an Open Source Stack to Create a Unified Data Source for Software Analysis and Visualization

Richard Müller*, Dirk Mahler†, Michael Hunger‡, Jens Nerche§ and Markus Harrer¶
*Leipzig University, Germany
Email: rmueller@wifa.uni-leipzig.de
†buschmais GbR, Dresden, Germany
Email: dirk.mahler@buschmais.com
‡Developer Relations, Neo4j Inc., Malmö, Sweden
Email: michael.hunger@neo4j.com
§Application Development, Kontext E GmbH, Dresden, Germany
Email: j.nerche@kontext-e.de
¶Software Development Analyst, Freelancer, Roth, Germany
Email: contact@markusharrer.de

***Abstract*—The beginning of every software analysis and visualization process is data acquisition. However, there are various sources of data about a software system. The methods used**

Creating, storing, and querying the data captured by such graphs is very challenging. Diehl et al. summarize the most

3. Analyse

Automatisierte,

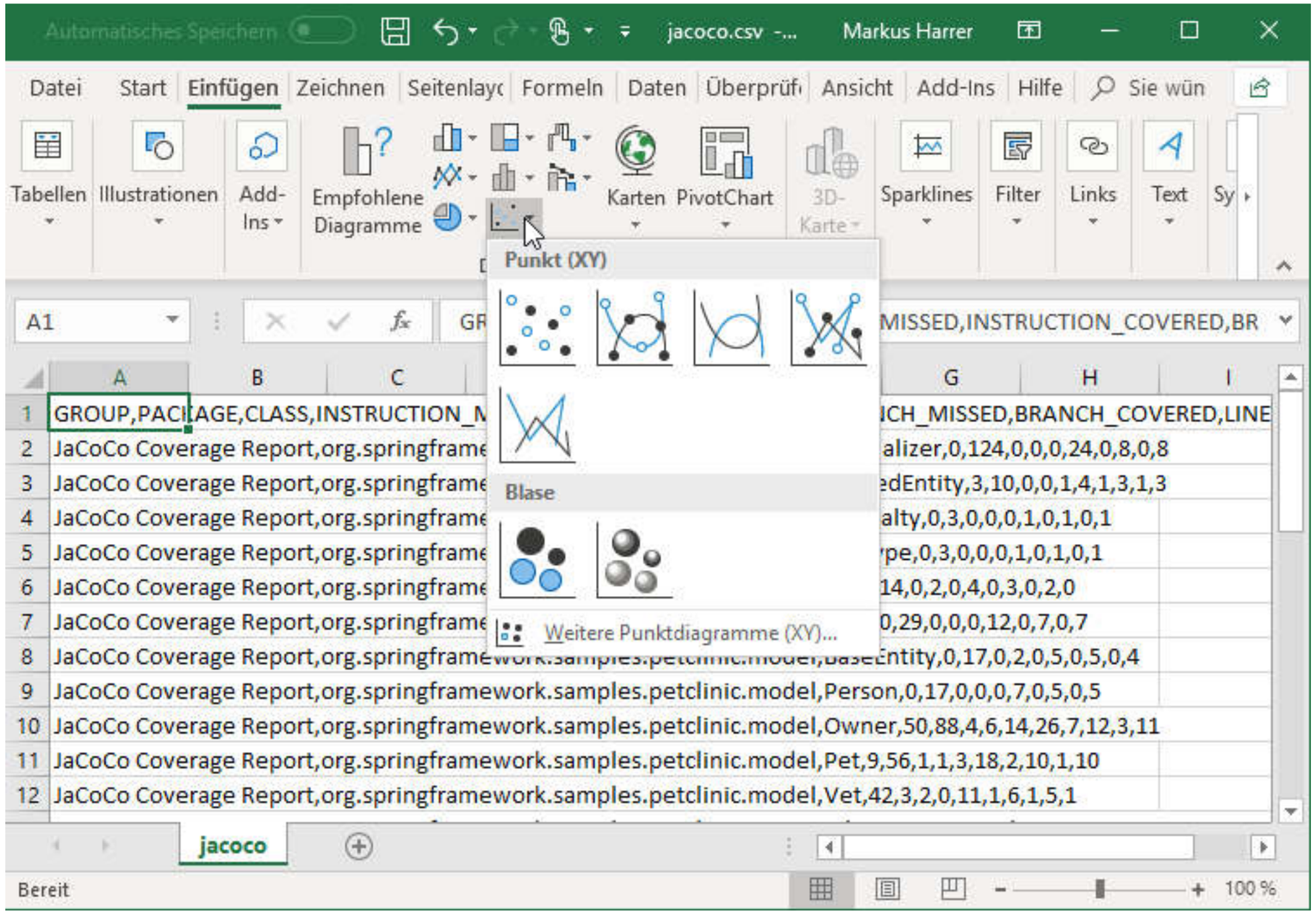
nachvollziehbare

Datenanalysen

Automatisierung

Manuelle Schritte vermeiden

- Automatisierung



Einfacher Ansatz

Automatisierung

Vorgehen zur Automatisierung

- 1 x per Hand Analyse durchführen
- Reflektieren
- Nachcodieren (= automatisieren)

Existierende Möglichkeiten nutzen

- Automatisierung

Kommandozeilen-Werkzeuge voll ausschöpfen

- `mvn dependency:analyze-duplicate`
- `git shortlog -ns -- *ViewModel.java`
- `cloc ./ --by-file --quiet --csv`

Web-APIs direkt anbinden

- Jenkins, SonarQube, Jira, BitBucket, …

Demo
„Software Archäologie mit Git"

```
/mnt/c/dev/repos/dropover
$
```

https://www.youtube.com/watch?v=ZaKYrvvLj0A

Nachvollziehbarkeit

Nutzung konventioneller Tools

- Nachvollziehbarkeit

Daten Analyse Ergebnis

Elemente offener Analysen

Nachvollziehbarkeit

- Rohdaten und Datentypen sind beschrieben
- Analyse-Code verfügbar
- Endergebnisse verständlich

Automatisierung

- Keine manuellen Eingriffe möglich / nötig

Einheitlicher Analyse-Aufbau

Nachvollziehbarkeit

Inhaltsverzeichnis

- Titel
- Motivation / Kontext
- Idee und verfügbare Daten
- Analyse
- Schlussfolgerung
  - Inkl. nächste Schritte

nach Jeff Leek: The Elements of Data Analytic Style

Glaubwürdigkeit herstellen

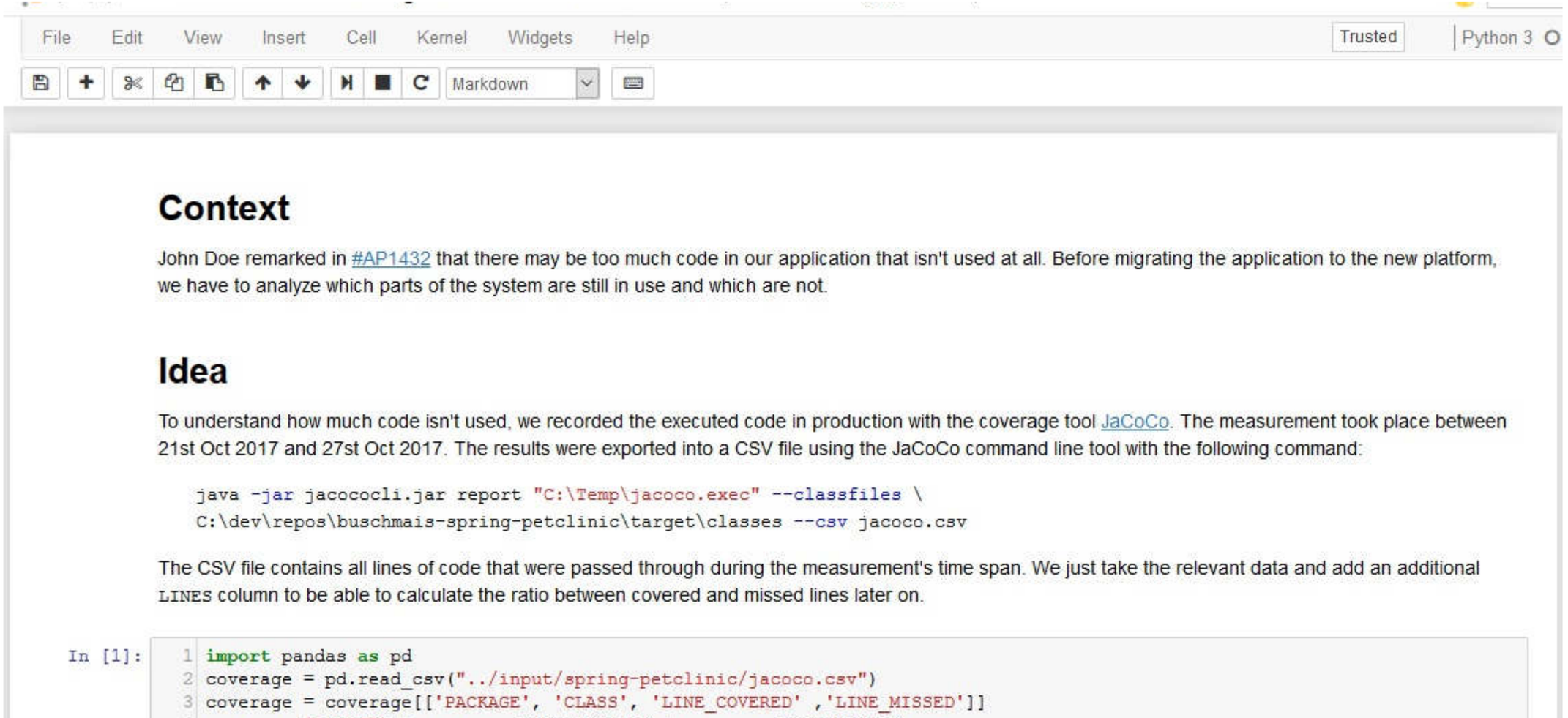
Nachvollziehbarkeit

Offenheit

- Filtern nur mit Begründung
- Zusammenfassen mit Erklärung
- Beschreibung, Code & Ergebnis pro gedanklichen Schritt
- Unterstützung mittels Visualisierungen

Notizbuch-Ansatz

- Nachvollziehbarkeit



4. Modell

Einfache Modelle verwenden

- Heuristiken statt 100%-Lösung
- Neue Perspektiven auf kleinteilige Daten schaffen
  - Verschiedenen Datenquellen verschneiden
  - Tabellen drehen / pivotieren
  - Gruppierungen vornehmen
    - Fachliche Bereiche, technische Komponenten, ...
    - Teams, Zeiträume, ...

=> Kommunikation mit Nicht-Technikern ermöglichen

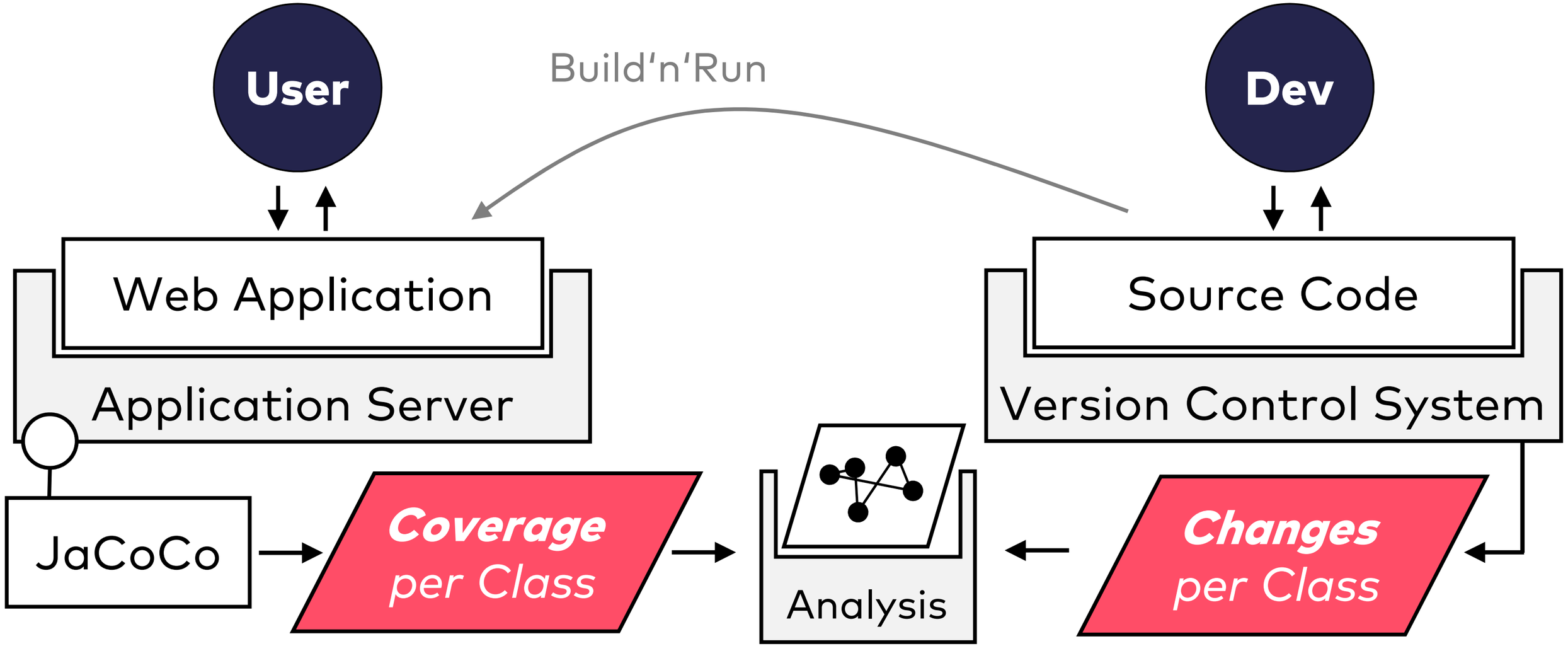
Beispiel
„Strategic Redesign"

Teil 1: Modell

https://www.feststelltaste.de/swot-analysis-for-spotting-worthless-code/

Strategic Redesign

Verbesserung von Quellcode, der auch wirklich genutzt wird



Strategic Redesign

Verbesserung von Quellcode, der auch wirklich genutzt wird

Subdomain	Invest	Usage	Size
Vet	75	17%	313
Visit	90	37%	472
Pet	169	49%	746
Owner	96	51%	531
crossfunctional	57	57%	268
Clinic	26	89%	110
Person	5	100%	53
Specialty	5	100%	28

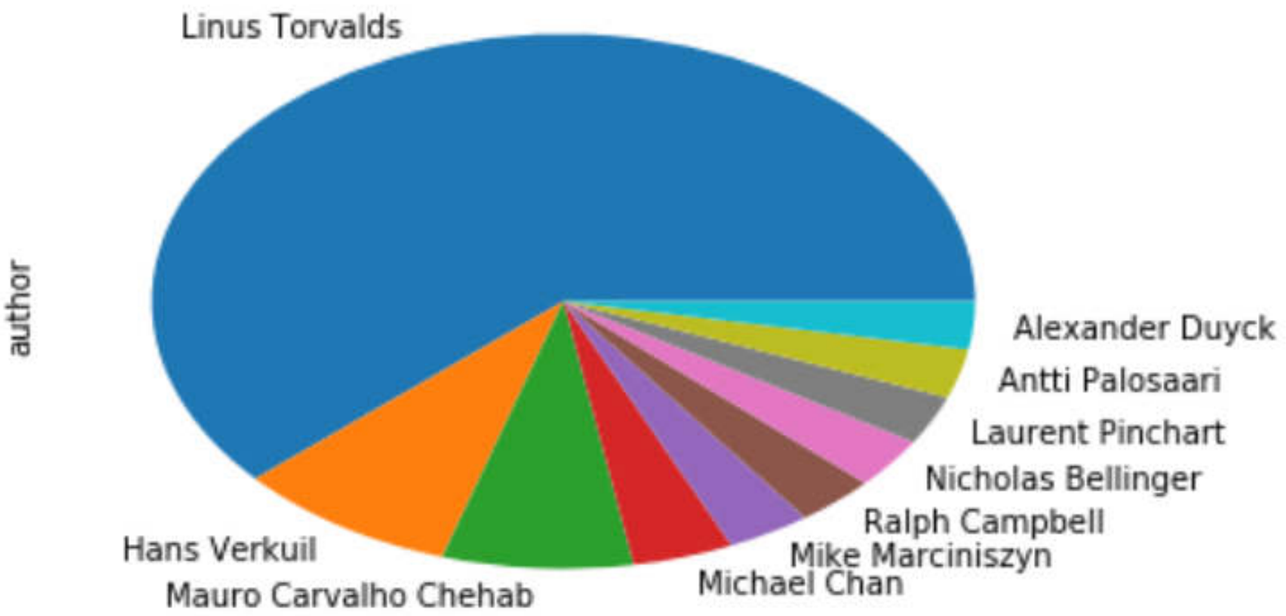
=> Fachliche Sicht auf technische Messwerte

5. Visualisierung

Effektive Visualisierungen

- Wenige, verständliche Informationen darstellen
- Zwischenergebnisse visualisieren
- Grafiken programmatisch aus Ergebnissen generieren

```
top10_authors.plot.pie()  =>
```



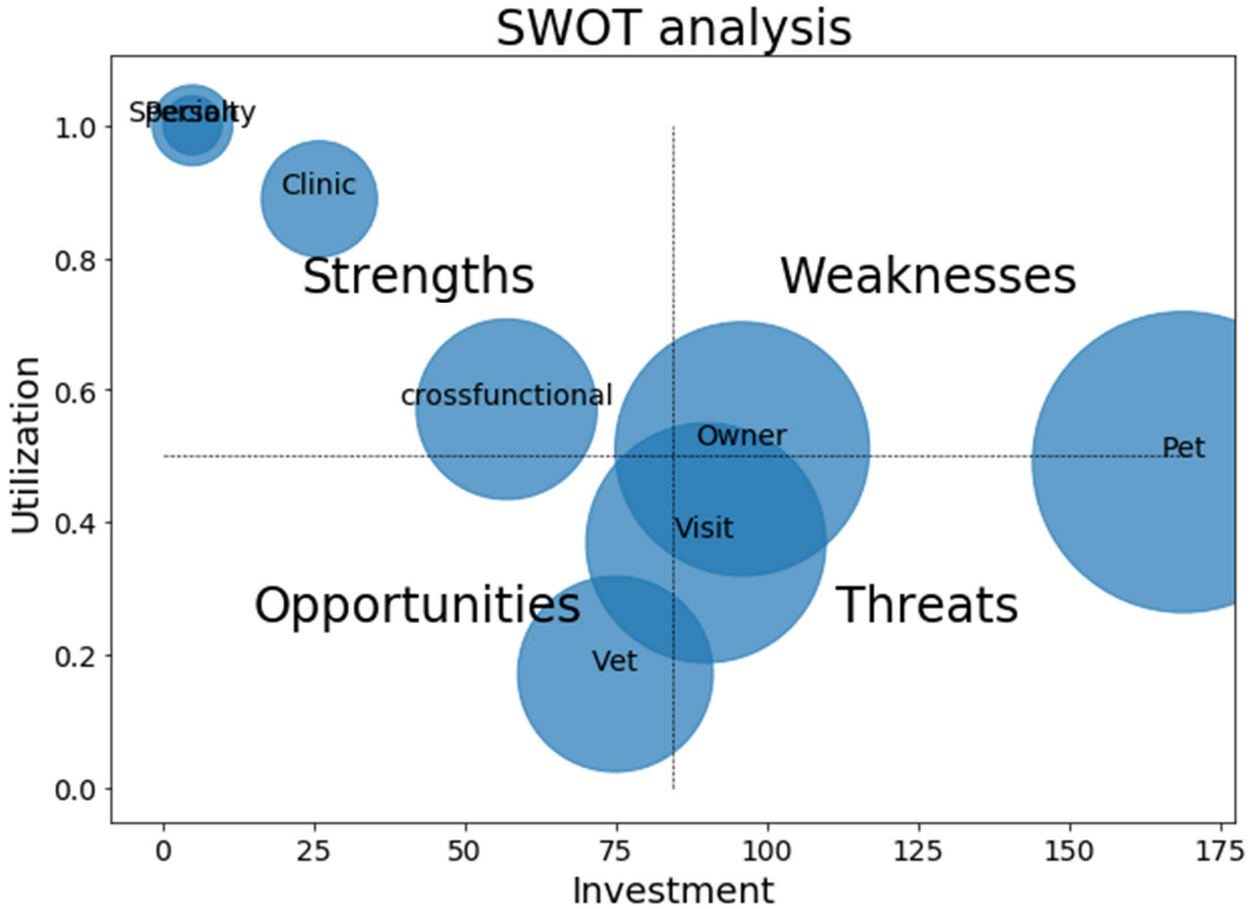
Beispiel
„Strategic Redesign"

Teil 2: Visualisierung

https://www.feststelltaste.de/swot-analysis-for-spotting-worthless-code/

Strategic Redesign

Verbesserung von Quellcode, der auch wirklich genutzt wird



=> Management-Sicht auf technische Messwerte

6. Erkenntnis

Kommunikation der Resultate

- Analyse-Notebooks und Daten bereitstellen

- Vorstellung mit Visualisierung und Kernaussagen starten
  - Bei Bedarf Rückweg bis zu den Rohdaten schildern

- Was immer dabei sein muss:
  - Zusammenfassung mit Schlussfolgerung
  - Nächste Schritte (=> Aktionen lostreten)

Demo
„No-Go-Areas"

https://www.feststelltaste.de/identifying-lost-knowledge-in-the-linux-kernel-source-code/

software-analytics/notebooks/ | No-Go-Areas

localhost:8888/notebooks/software-analytics/notebooks/No-Go-Areas.ipynb 150%

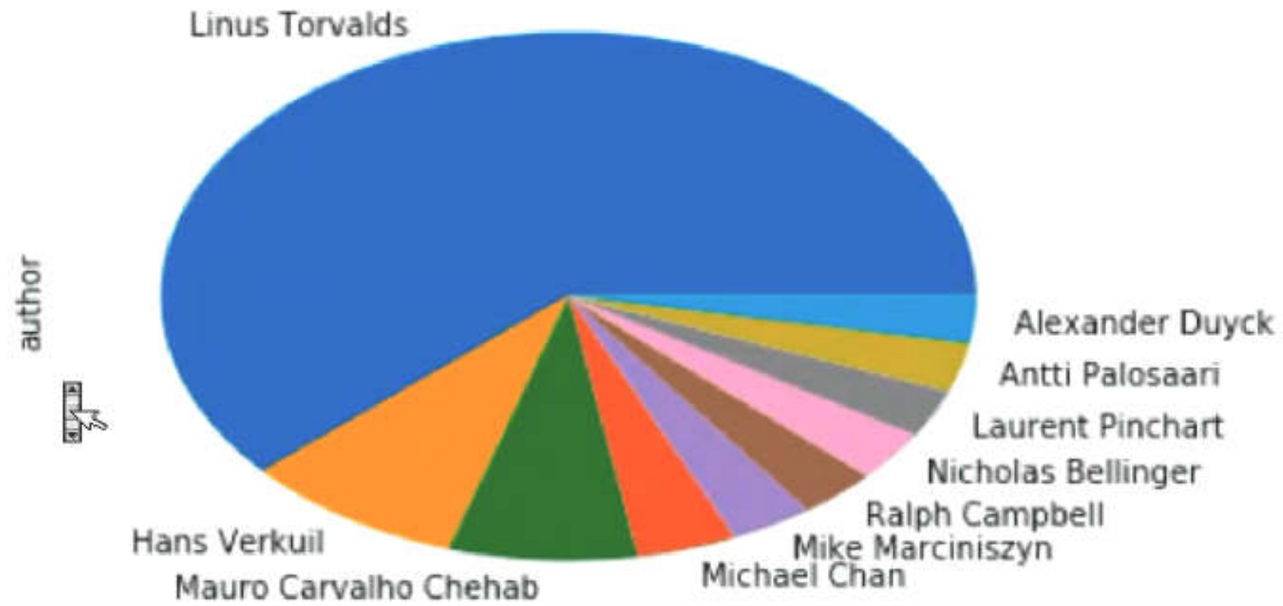
jupyter No-Go-Areas (unsaved changes) Logout

File Edit View Insert Cell Kernel Help Trusted Python 3

Run Code

```
2 top10.plot.pie()
```

Out[6]: <matplotlib.axes._subplots.AxesSubplot at 0x25948362278>



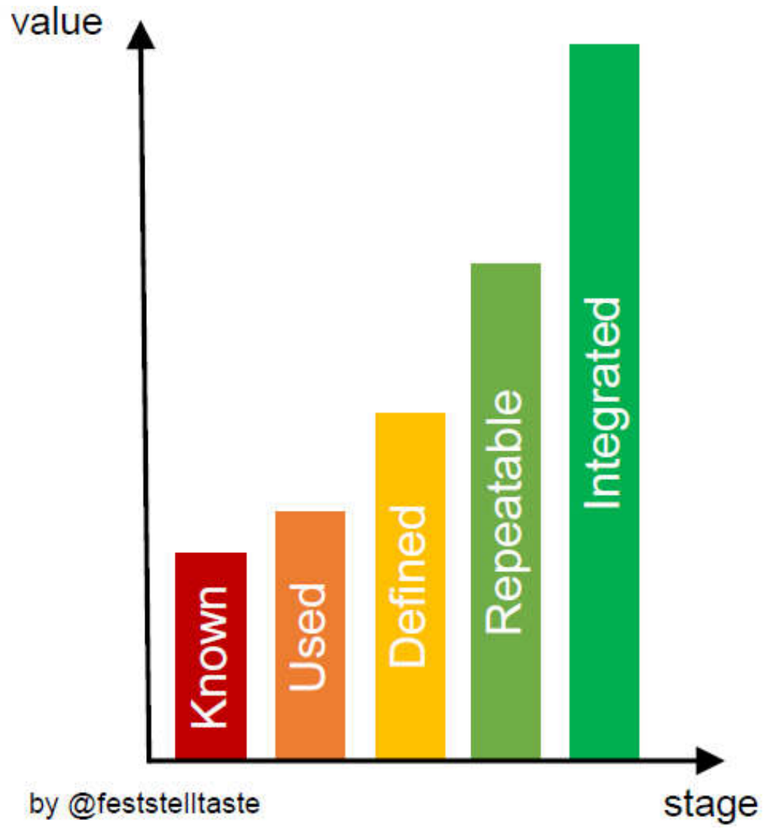
https://www.youtube.com/watch?v=fUlq2y3SI04

Getting Started

Eigenen Stand bzgl. Analysen klären

Analyse-Reifegrad

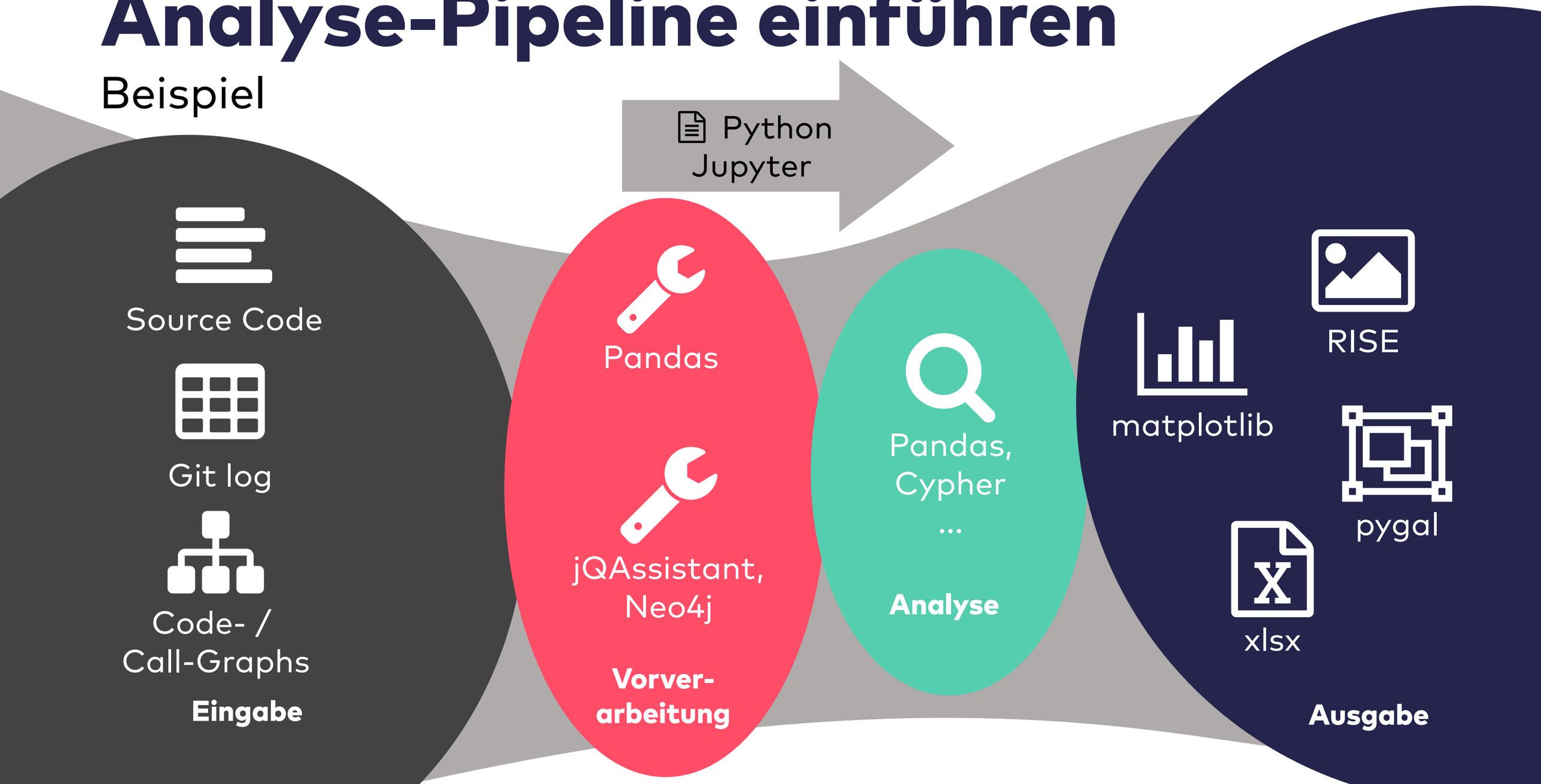
1. Bekannt
2. Genutzt
3. Definiert
4. Wiederholt
5. Integriert



https://www.feststelltaste.de/software-analytics-maturity-model/

Analyse-Pipeline einführen

Beispiel



Loslegen!

Blog: feststelltaste.de

Literatur

- Adam Tornhill: Software Design X-Ray
- Wes McKinney: Python For Data Analysis
- Leek, Jeff: The Elements of Data Analytic Style
- Christian Bird, Tim Menzies, Thomas Zimmermann:
  The Art and Science of Analyzing Software Data
- Tim Menzies, Laurie Williams, Thomas Zimmermann:
  Perspectives on Data Science for Software Engineering

Software

- Python Data Science Distribution: anaconda.com
- jQAssistant: github.com/JavaOnAutobahn/spring-petclinic
- GitHub-Repo: github.com/feststelltaste/software-analytics
- Mini-Tutorial: feststelltaste.de/mini-tutorial-git-log-analyse-mit-python-und-pandas

Zusammenfassung

Zusammenfassung

1. Methoden und Werkzeuge sind da
2. Kommunikation kniffliger Probleme möglich
3. Best Practices helfen beim Einstieg

Danke!

INNOQ
www.innoq.com

Markus Harrer
markus.harrer@innoq.com

+49 175 5753640
@feststelltaste
https://feststelltaste.de

Heute und morgen am INNOQ-Stand!

innoQ Deutschland GmbH

Krischerstr. 100
40789 Monheim am Rhein
Germany
+49 2173 3366-0

Ohlauer Str. 43
10999 Berlin
Germany

Ludwigstr. 180E
63067 Offenbach
Germany

Kreuzstr. 16
80331 München
Germany

innoQ Schweiz GmbH

Gewerbestr. 11
CH-6330 Cham
Switzerland
+41 41 743 01 11

Albulastr. 55
8048 Zürich
Switzerland

QUESTIONS

ASK ´ EM ALL

Auch gerne heute und morgen am INNOQ-Stand!

INNOQ

"Tools only find, people have to find out!"

Markus Harrer

Software Development Analyst

bei innoQ Deutschland GmbH

- Datenanalysen in der Softwareentwicklung
- Architektur-, Design- und Code-Reviews
- Reverse- und Re-Engineering von Legacy-Code

Anhang: Werkzeuge

Interaktives Notebooksystem

- Dokumentenzentrierte Analysen
- Ausführbare Codeblöcke
- Direkt sichtbare Visualisierungen

https://www.feststelltaste.de/top5-jupyter

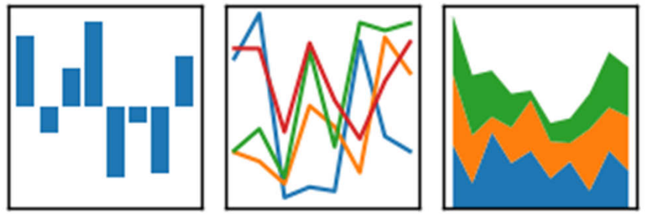
Programmiersprache für Data Science

- ✔ Einfach
- ✔ Effizient
- ✔ Schnell

https://www.feststelltaste.de/top5-python

pandas

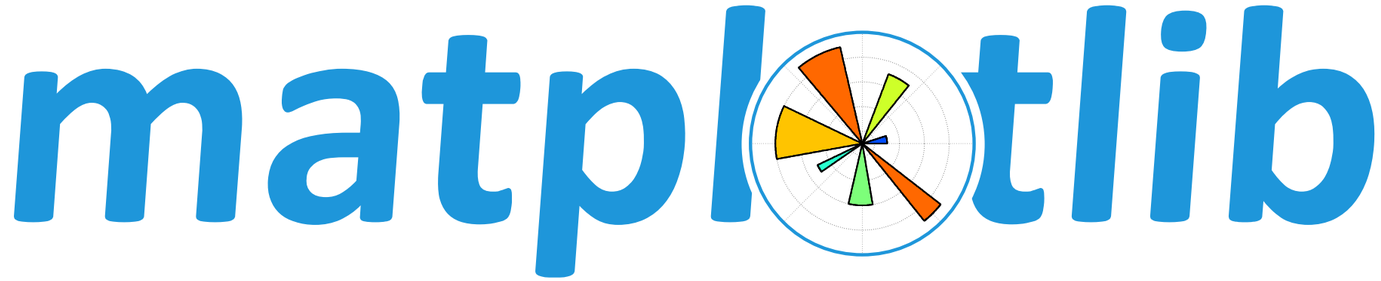
$$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$$



Pragmatisches Datenanalysewerkzeug

- ✔ Das programmierbare Excel-Arbeitsblatt
  - Richtig schnell
  - Flexibel
  - Ausdrucksstark
- ➔ Sehr gute Integration mit anderen Bibliotheken

https://www.feststelltaste.de/top5-pandas



Visualisierungsbibliothek

Ermöglicht die programmatische Erstellung von Grafiken

- Erstellung von Balken-, Linien-Diagrammen und mehr
- Gute Integration mit pandas & Co.

➔ Direkte Ausgabe in Jupyter Notebooks

https://www.feststelltaste.de/top5-matplotlib

Python Ökosystem

Datenanalysen

- NumPy
- scikit-learn
- TensorFlow
- Dask
- Py2neo
- Pygments

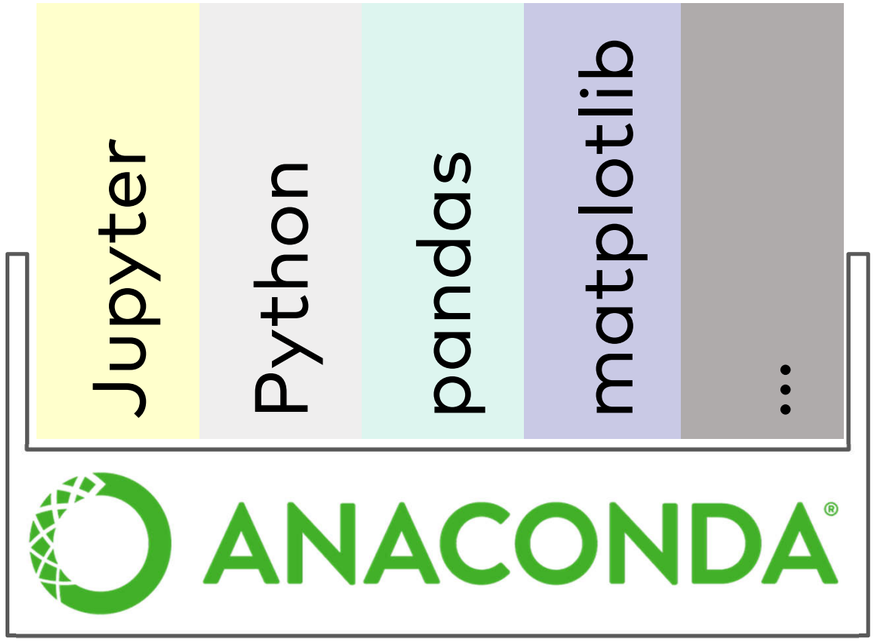
Visualisierung / Präsentation

- pygal
- Bokeh
- python-pptx
- RISE

Other

- Scrapy, Selenium, Flask

in Demos verwendet



Data Science Python Distribution

✔ All-Inclusive Paket (kostenlos!)

- Bringt alles mit, was für den Start gebraucht wird
- Mitgelieferte Pakete sind untereinander abgestimmt und für das jeweilige Betriebssystem optimiert

➔ Downloaden, installieren, loslegen!

Die Spezialisten für vernetzte Daten

ℹ Framework zur statischen Architektur- und Code-Analyse auf Basis von Softwaredaten

[:SPEICHERT_IN]

ℹ Graph-Datenbank zur Ablage und Analyse stark vernetzter Daten

https://www.feststelltaste.de/top5-jqassistant

www.innoq.com

SERVICES

Strategy & technology consulting
Digital business models
Software architecture & development
Digital platforms & infrastructures
Knowledge transfer, coaching & trainings

FACTS

~125 employees
Privately owned
Vendor-independent

OFFICES

Monheim
Berlin
Offenbach
Munich
Zurich

CLIENTS

Finance
Telecommunications
Logistics
E-Commerce
Fortune 500
SMBs
Startups