

# Software Analytics with Data Science on Software Data

**Markus Harrer**  
Software Evolutionist @ INNOQ

**INNOQ**



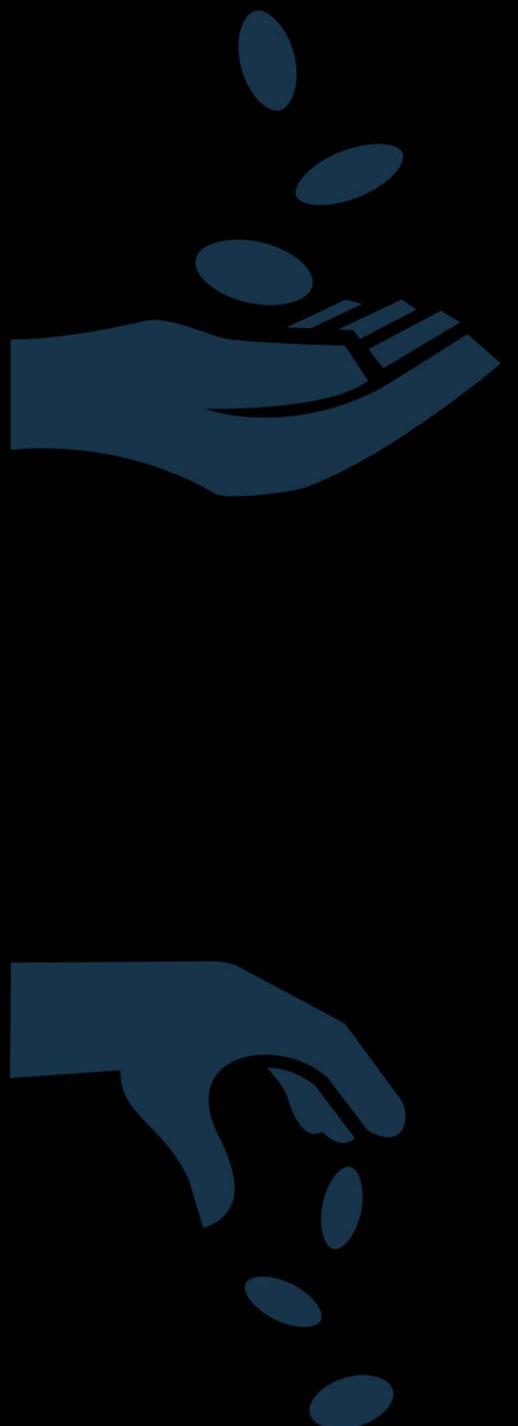
Social: [markusharrer.de](http://markusharrer.de)



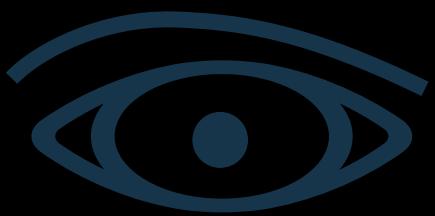
THE ORIGINAL HORROR SHOW

LEGACY SYSTEMS

# THE PROBLEM WITH THE PROBLEMS



FEATURES



ARCHITECTURE



ERRORS

STRANGE  
- THINGS -

MANAGEMENT

---

COMMUNICATION GAP

---

DEVELOPMENT

# SHIPPING APP

LCARS 40274

02-654598

2385	8578232	9	5789	3882	5893	9885	3489	3465	0846	9798	9629	29
2064	2064962	7	9776	626	1276	7612	126	97	6165	6626	876	74
34	279	89	6589	6547	6587	3465	867	2347	5762	4588	05	
4768	8967248	7	9798	8969	476	9047	8476	9749	0982	8969	0247	89
685	3478	8	867	346	34	48	49	8	89	897	38	
757	898990	8	200	285	923	9	387	238	578	875	87	9
484	947589	7	569	68	678	893	56	584	678	476	458	4

9886-234

0128-069

1014-819

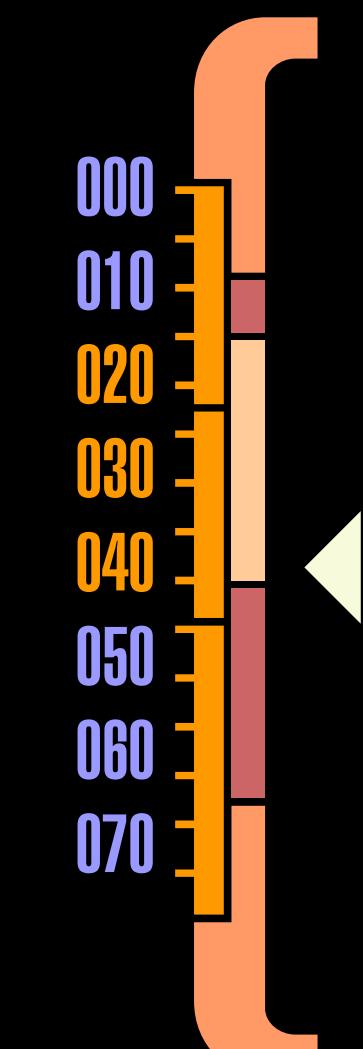
7232-838

03-975683

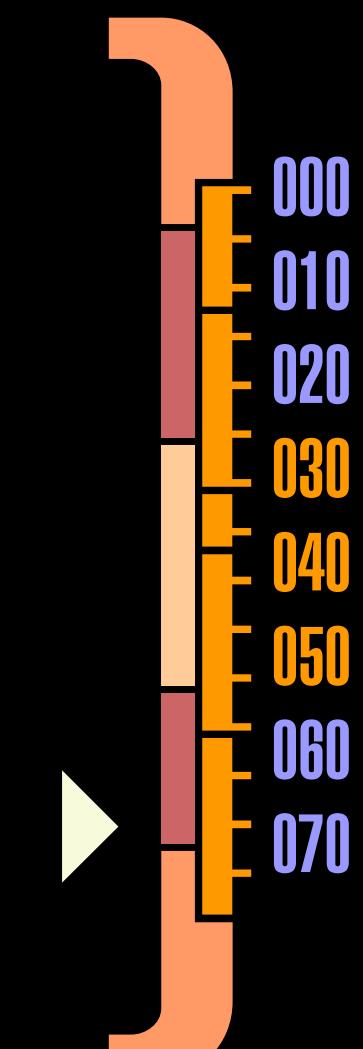
04-765466

05-224353

06-576565



DEVELOPER'S SANITY



PROJECT'S BUDGET

# THE ULTIMATE QUALITY DASHBOARD

# THE EMPIRIC STRIKES BACK

Not a long time ago, from brains  
not far, far away....

**SOFTWARE  
ANALYTICS**

# SOFTWARE ANALYTICS

A definition of

**MENZIES & ZIMMERMANN**

**Software Analytics**

*is analytics on software data for  
**managers** and **software engineers***

*with the aim of empowering  
**software development individuals**  
and teams*

*to gain and share insight from  
**their data** to make better  
decisions.*

**SOFTWARE  
ANALYTICS**

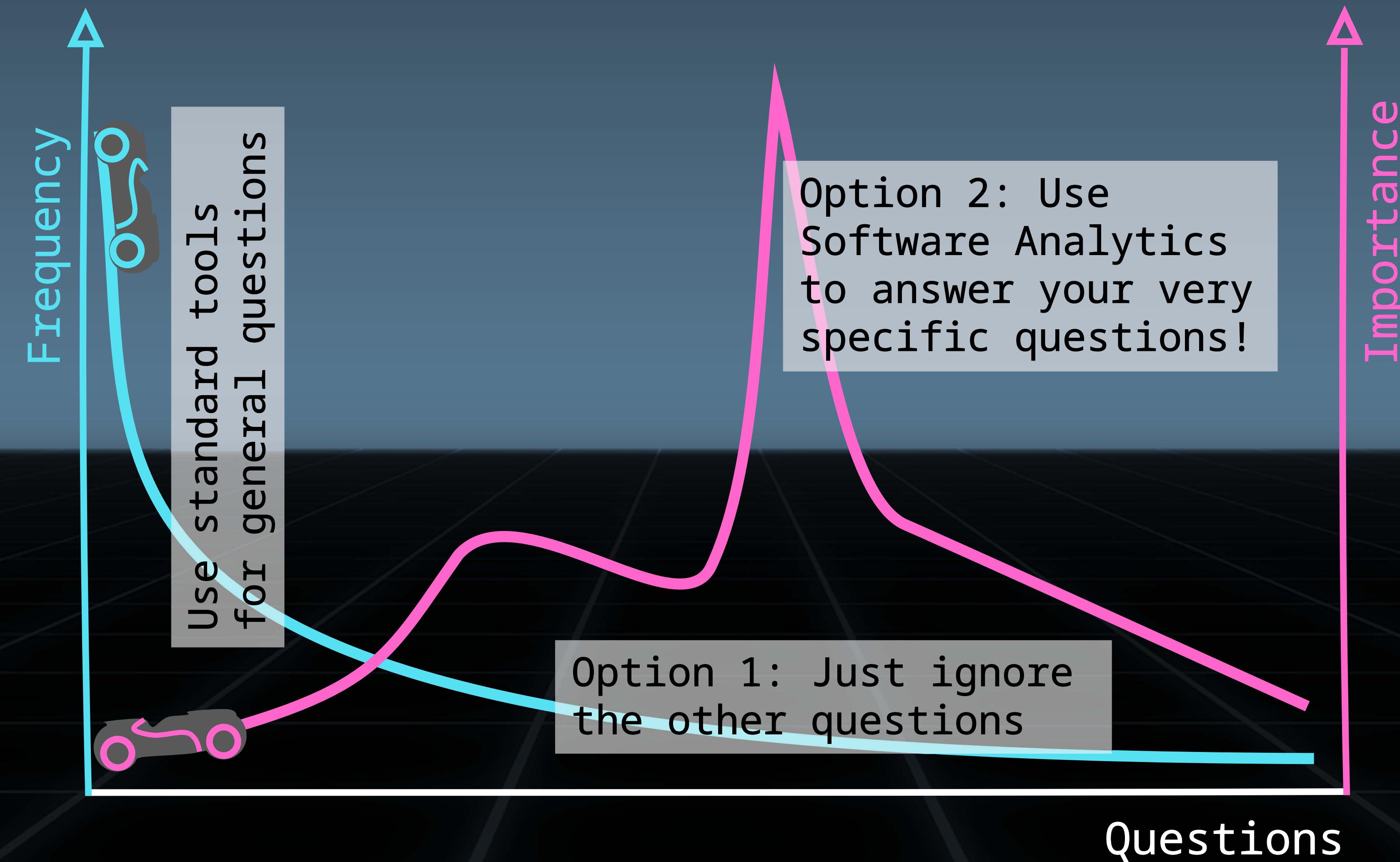
---

**EPISODE II**

---

**A NEW HOPE**

# ANSWERING YOUR SPECIFIC QUESTIONS



# TYPICAL ISSUES TO TERMINATE

- Spotting parts in source code no one knows of anymore
- Finding root causes of performance bottlenecks
- Identifying alternative modularization options
- Showing the progress of long-living restructurings
- Measuring the community activity around open source software
- <your very specific analysis in your very specific situation>

# THE SOFTWARE DATA OF A TEAM OF DEVELOPERS

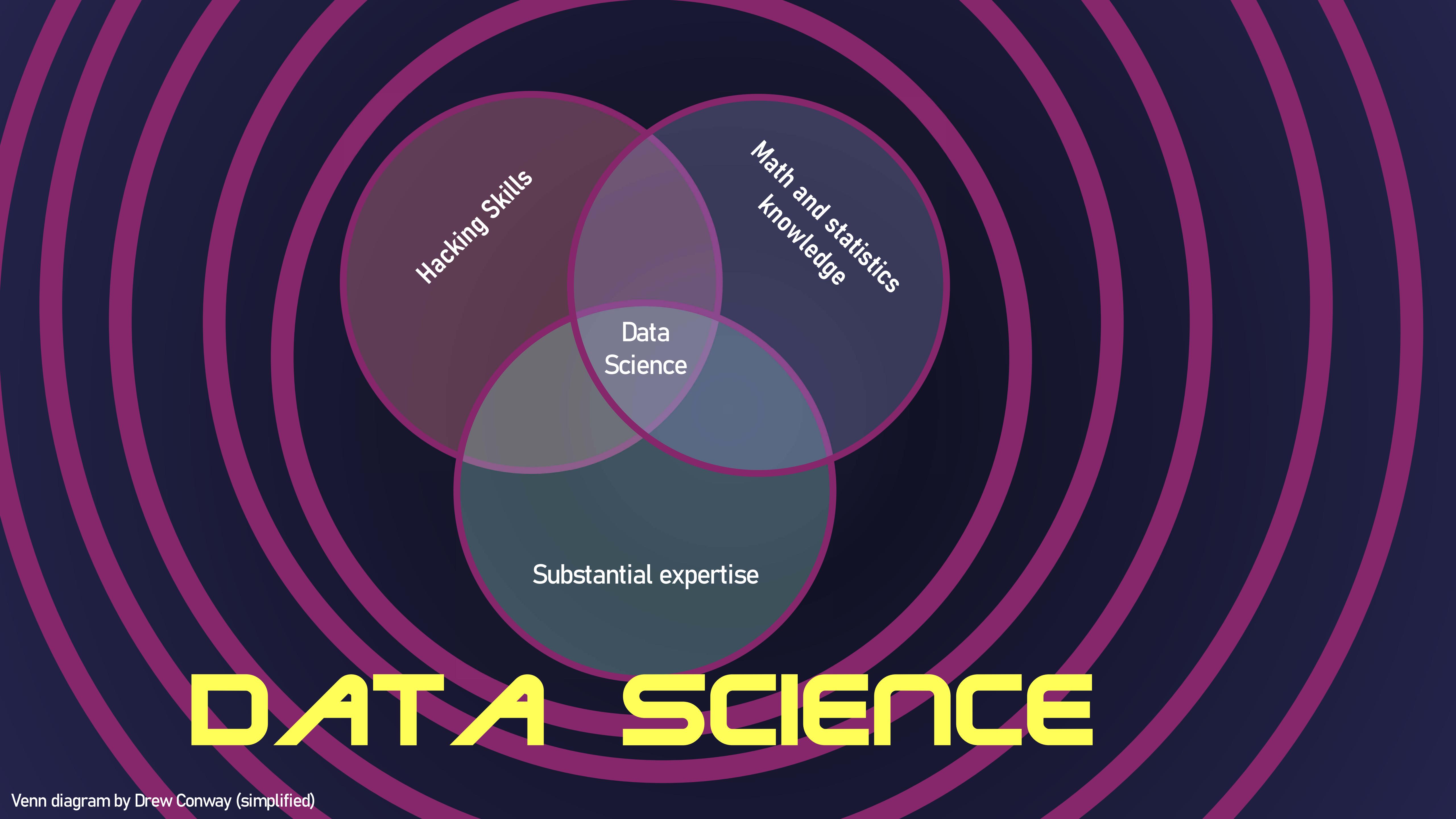
Chronolo-  
gical data

Static  
data

Community  
data

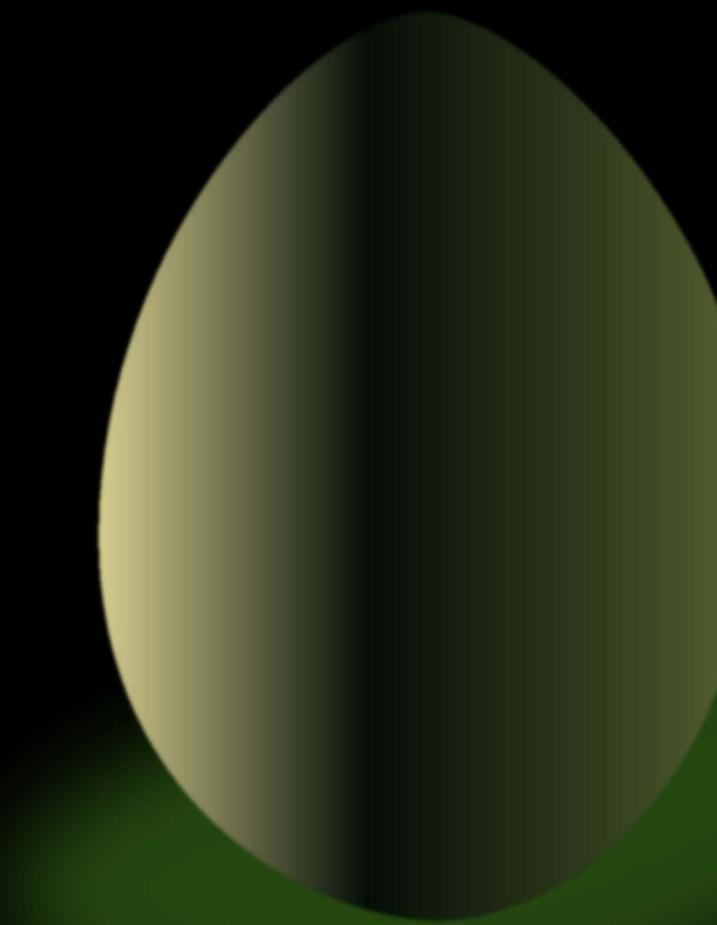
Runtime  
data

# DATA SCIENCE



# REPRODUCIBLE DATA SCIENCE

open



automated

systematic

comprehensible

= A WAY TO IMPLEMENT SOLID  
**SOFTWARE ANALYTICS**

TOOL TIME

# Python 3



# PANDAS



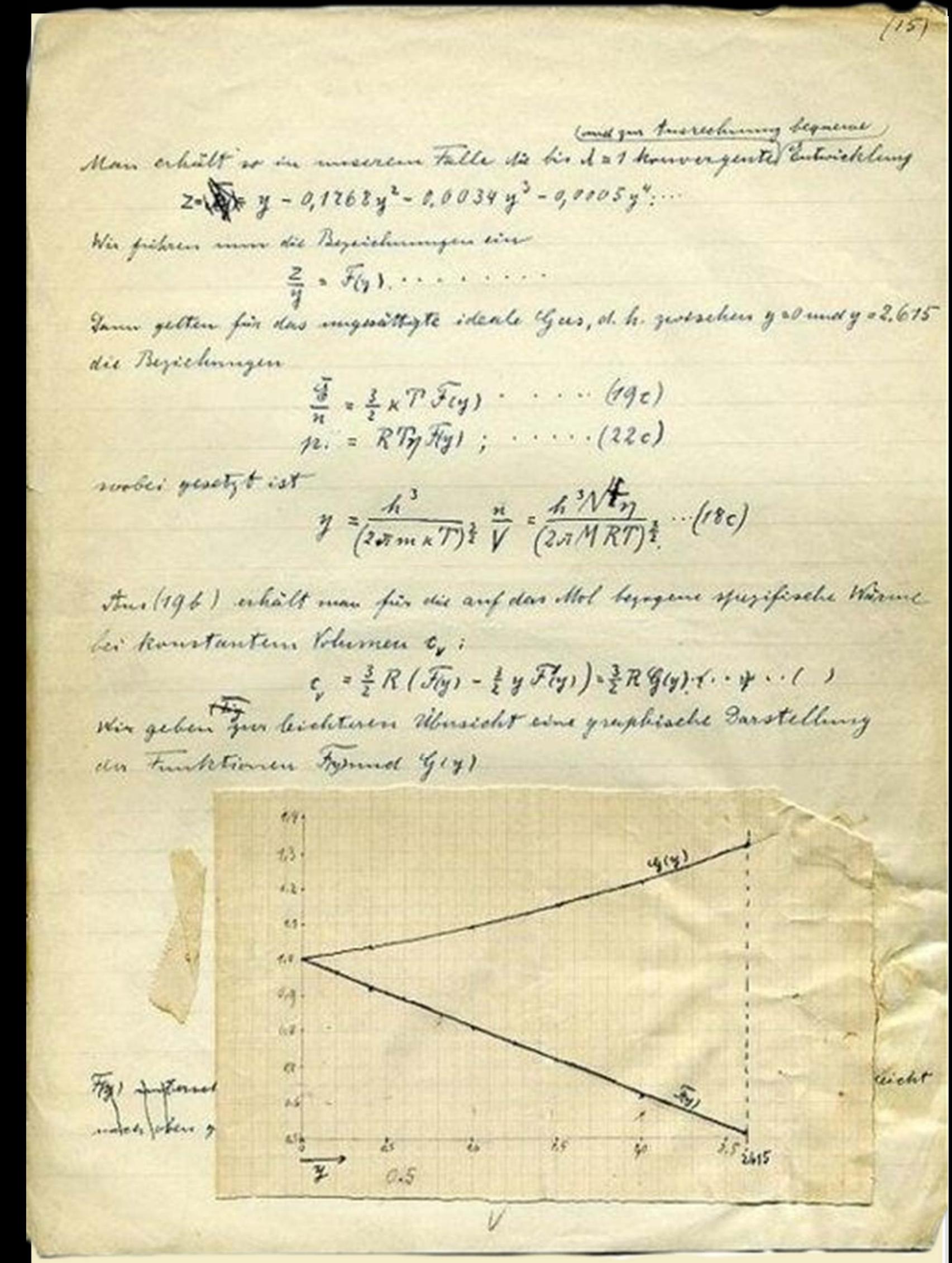
... and matplotlib, numpy, scikit-learn, NLTK, Pygments, py2neo, requests, BeautifulSoup, Pygal ...

# Computational Notebook

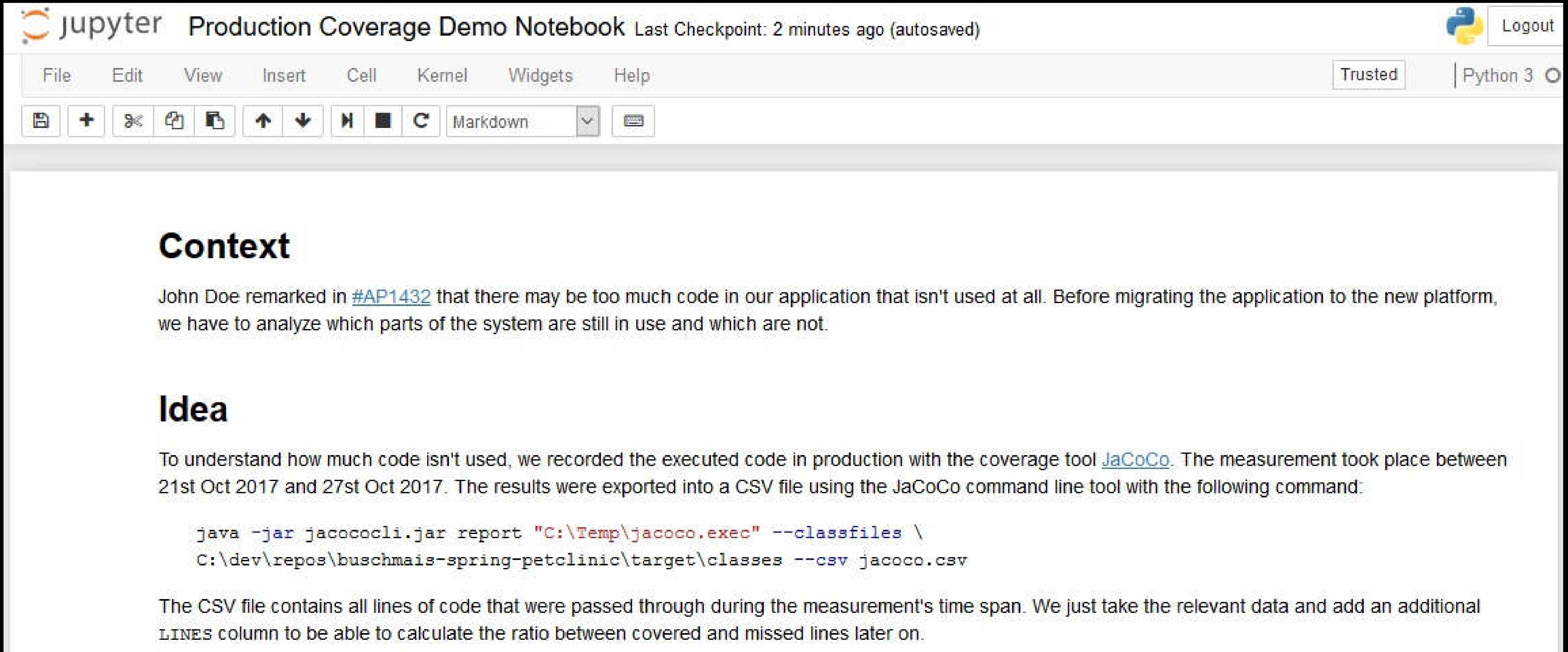
code and data in love

# Computational Notebook

## Jupyter Notebook



# Literate Programming with Jupyter Notebook



The screenshot shows a Jupyter Notebook interface with the following details:

- Title Bar:** "jupyter Production Coverage Demo Notebook Last Checkpoint: 2 minutes ago (autosaved)"
- Toolbar:** Includes File, Edit, View, Insert, Cell, Kernel, Widgets, Help, Trusted, Python 3, and a Python logo icon.
- Cell Buttons:** Standard Jupyter cell controls (New, Run, Stop, etc.) and a "Markdown" button.
- Content Area:** Contains two sections:
  - Context:** A text block stating: "John Doe remarked in [#AP1432](#) that there may be too much code in our application that isn't used at all. Before migrating the application to the new platform, we have to analyze which parts of the system are still in use and which are not."
  - Idea:** A text block stating: "To understand how much code isn't used, we recorded the executed code in production with the coverage tool [JaCoCo](#). The measurement took place between 21st Oct 2017 and 27st Oct 2017. The results were exported into a CSV file using the JaCoCo command line tool with the following command:
- Code Block:**

```
java -jar jacococli.jar report "C:\Temp\jacoco.exec" --classfiles \
C:\dev\repos\buschmais-spring-petclinic\target\classes --csv jacoco.csv
```
- Text Description:** "The CSV file contains all lines of code that were passed through during the measurement's time span. We just take the relevant data and add an additional LINES column to be able to calculate the ratio between covered and missed lines later on."

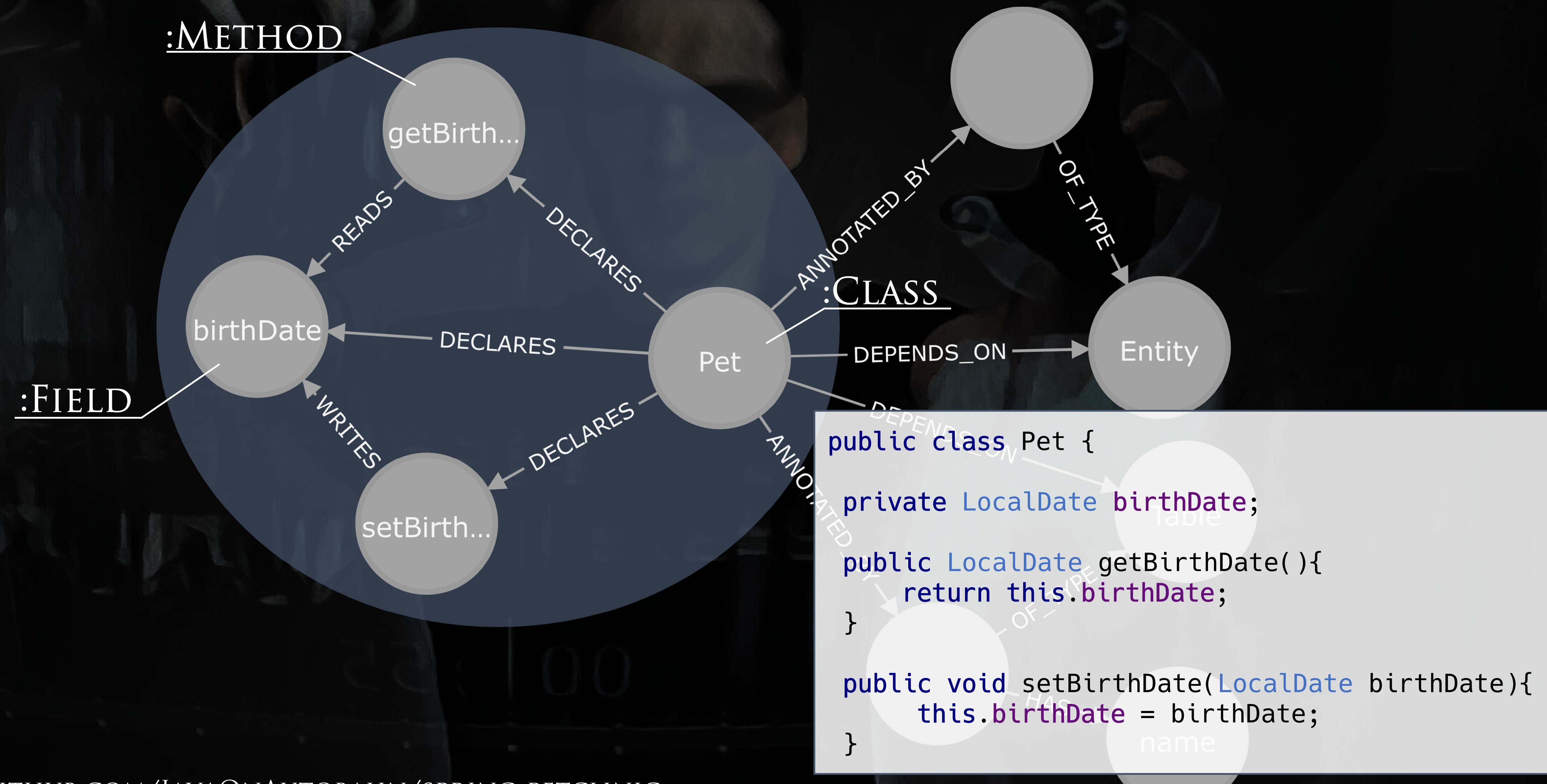


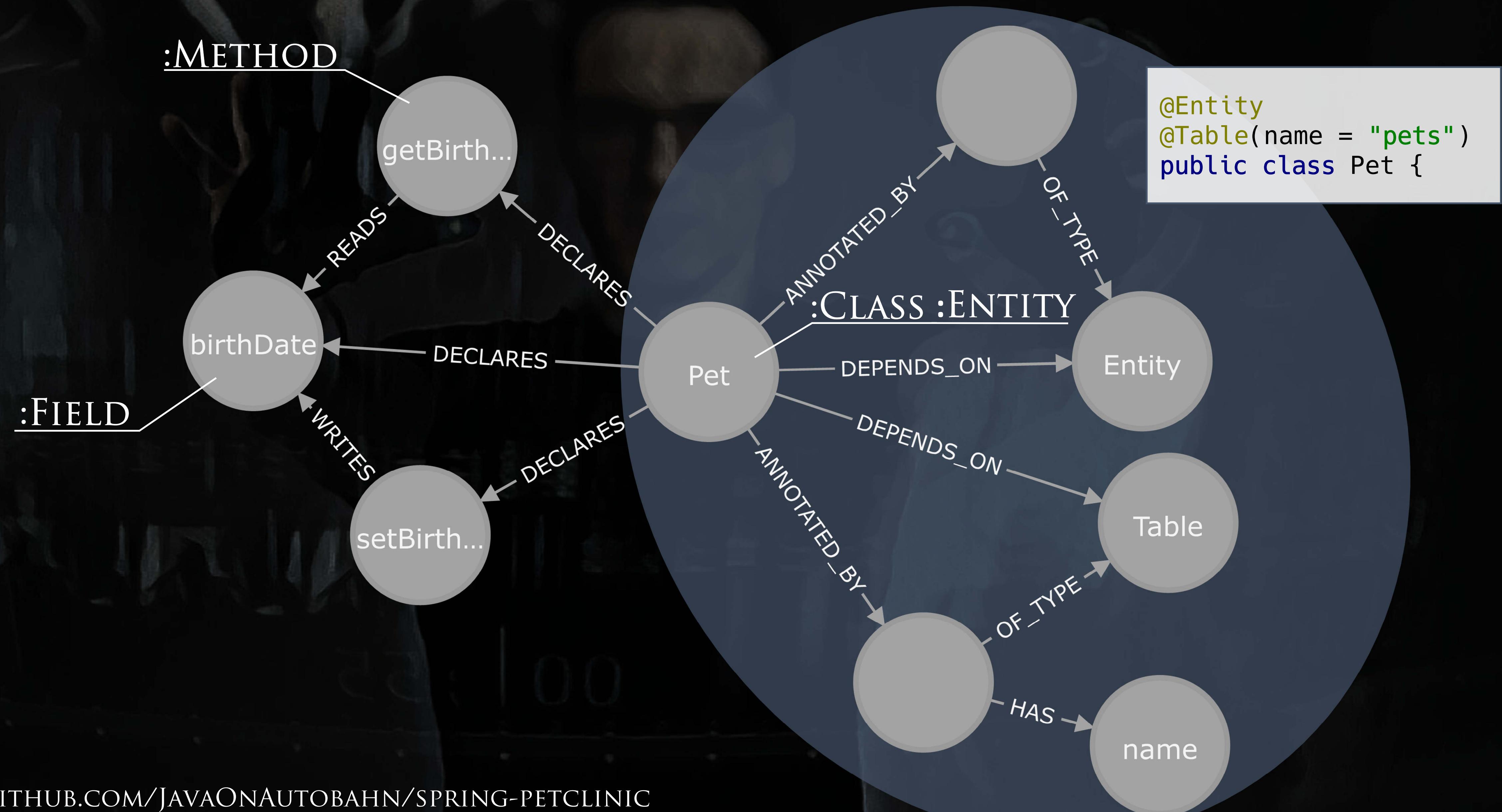
# GRAPH DATA SCIENCE

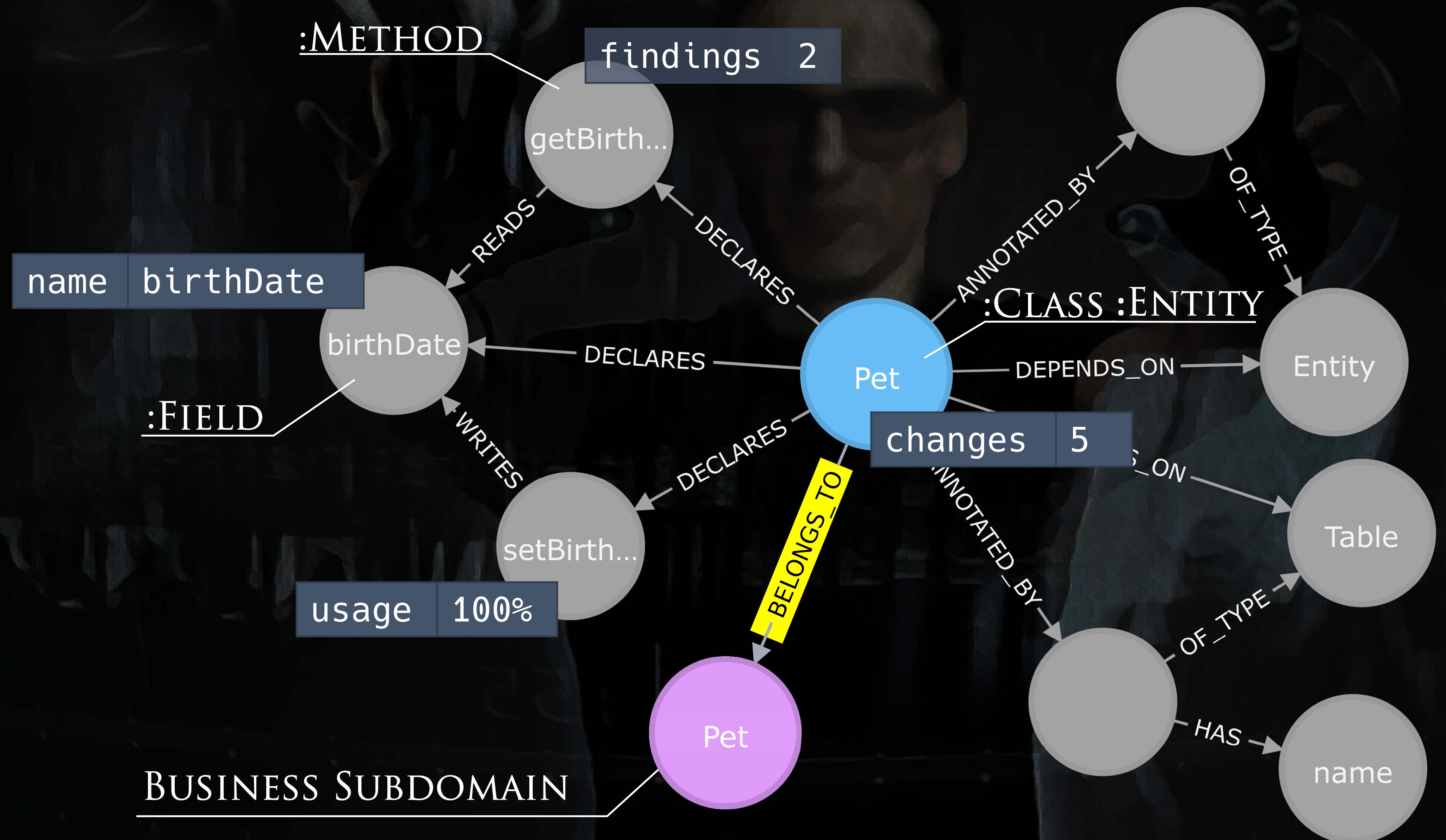


JQASSISTANT

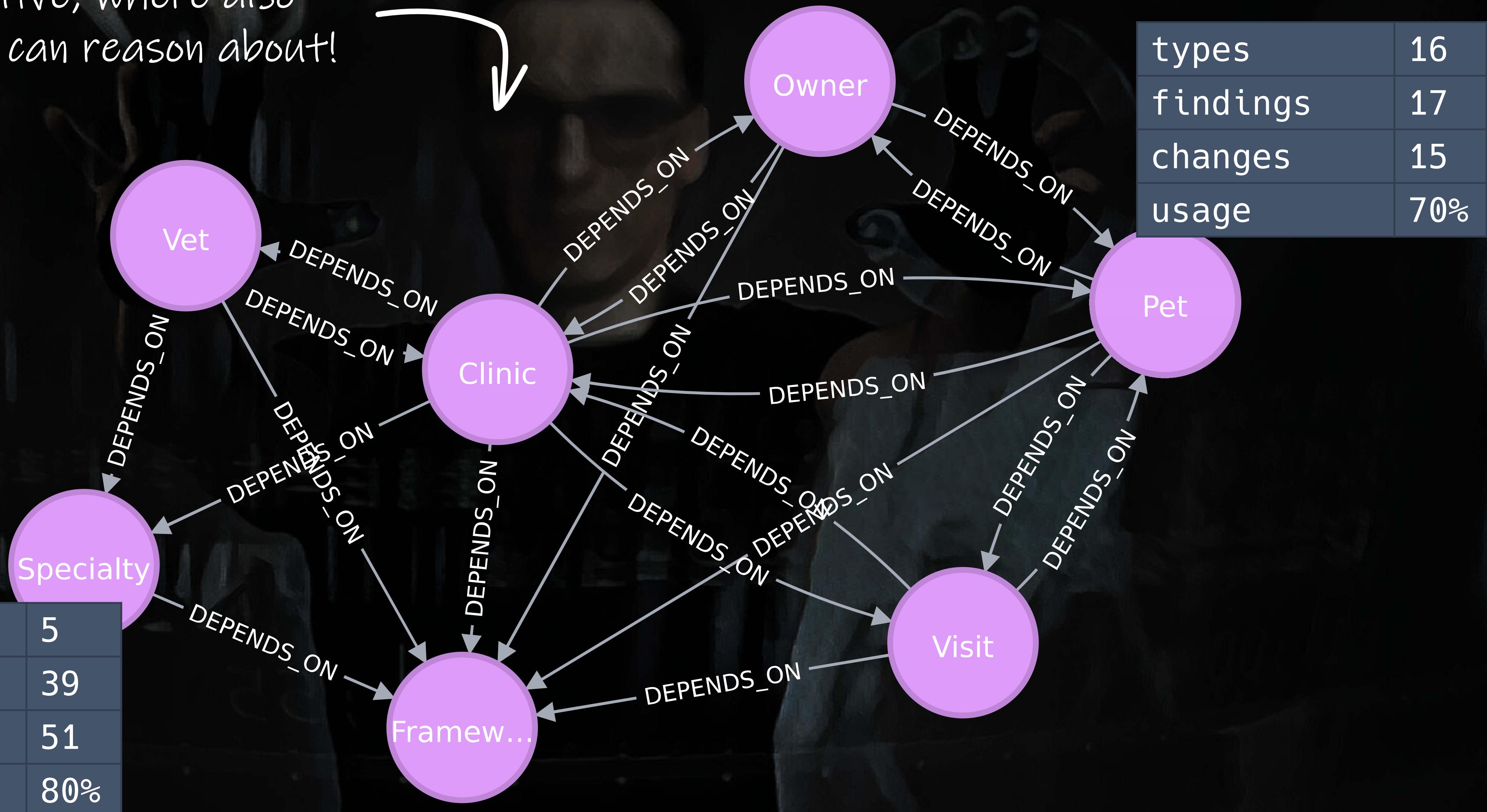
NEO4J







A perspective, where also  
managers can reason about!





BECOME THE  
**LORD** OF THE **THINGS**  
BY ANALYZING SOFTWARE IN A DATA-DRIVEN WAY

NO TOOL TO  
RULE THEM ALL

THE FELLOWSHIP  
OF THE BLING

THE TWO TIPS

O P E N E S S

A U T O M A T I O N

THE RETURN  
OF REASON



NUMBER OF  
SOLVED PROBLEMS



ASK ' EM ALL



@feststelltaste

# Appendix

# Demos

**Jupyter notebook, python, pandas, matplotlib**

Repo

[https://github.com/feststelltaste/software-analytics/tree/master/demos/20240315\\_BOBKonf\\_2024](https://github.com/feststelltaste/software-analytics/tree/master/demos/20240315_BOBKonf_2024)

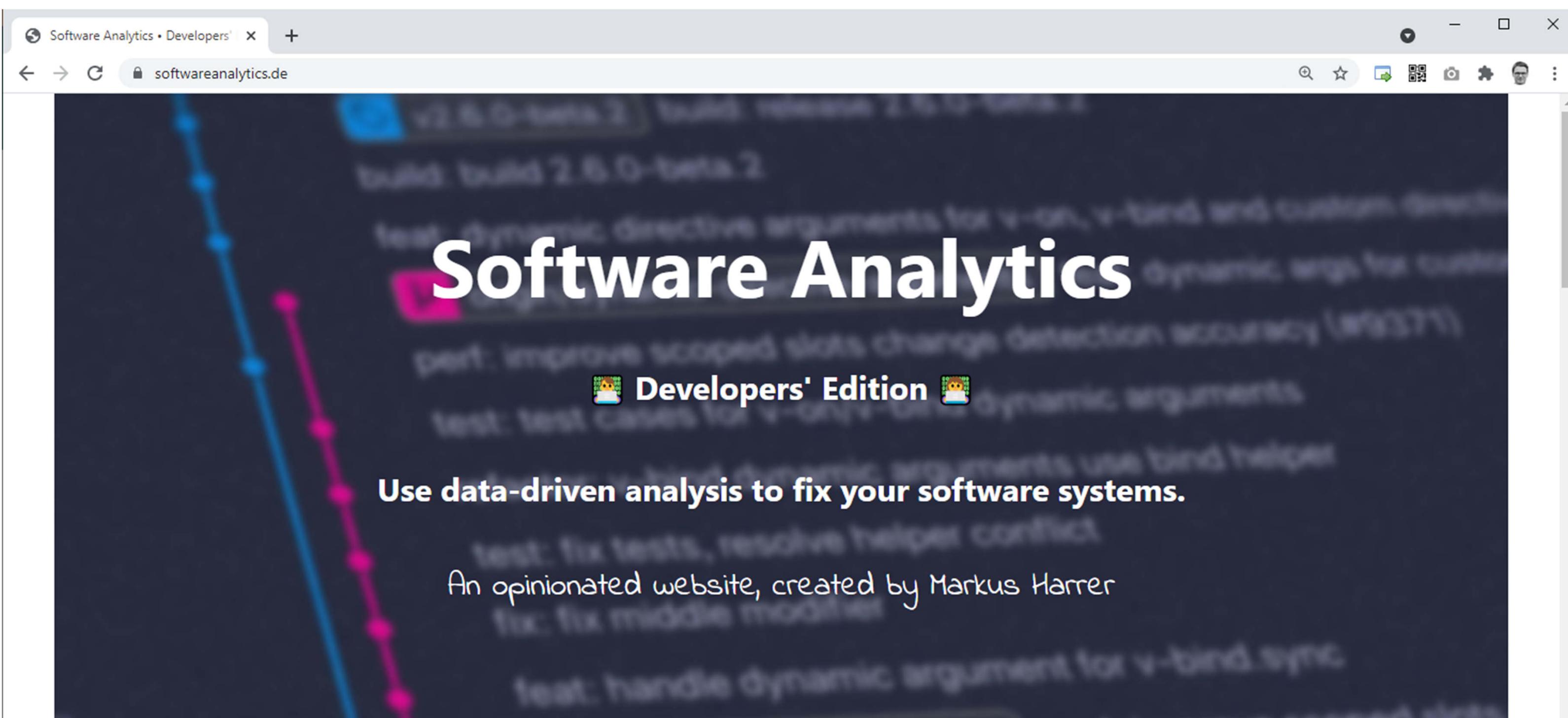
**jQAssistant & Neo4j**

Repo Spring PetClinic

<https://github.com/JavaOnAutobahn/spring-petclinic>

# More on Software Analytics

[softwareanalytics.de](http://softwareanalytics.de)

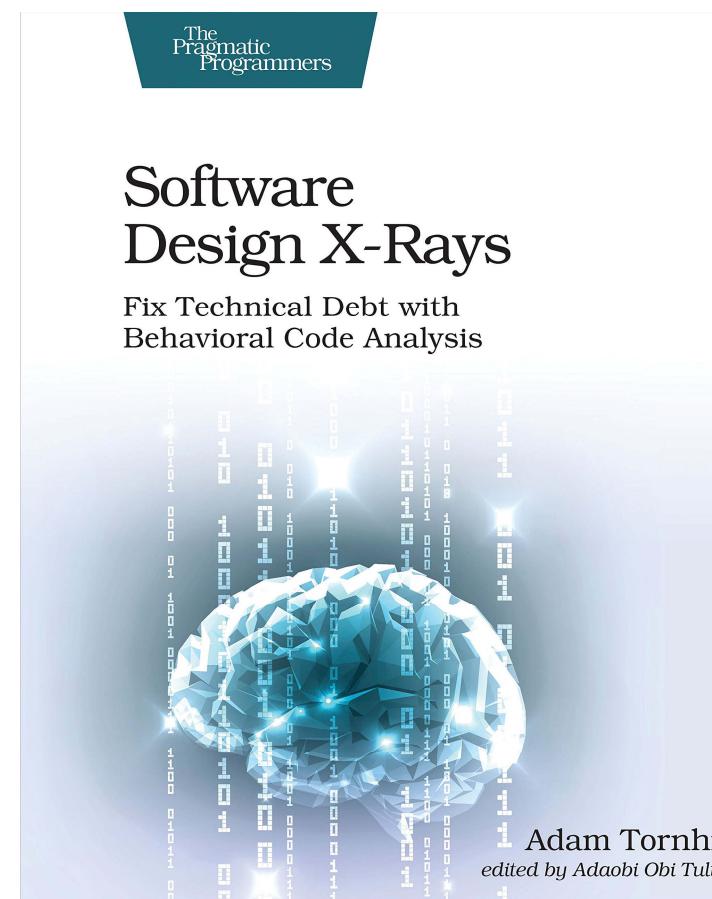


## What is "Software Analytics"?

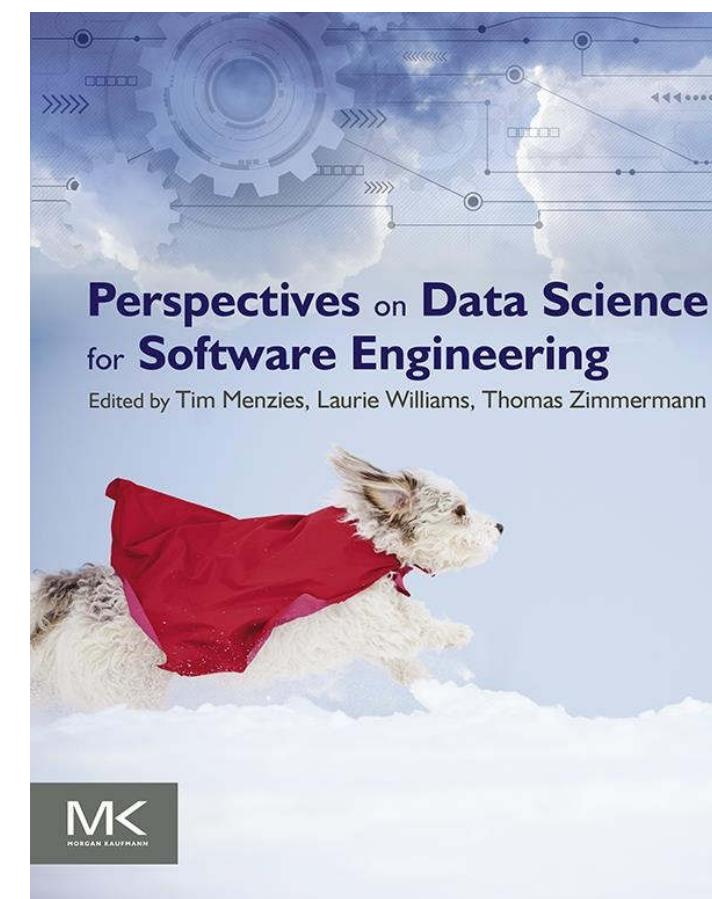
This is the best definition of Software Analytics I know so far:

Software analytics is analytics on software data for managers and software engineers with the aim of empowering software

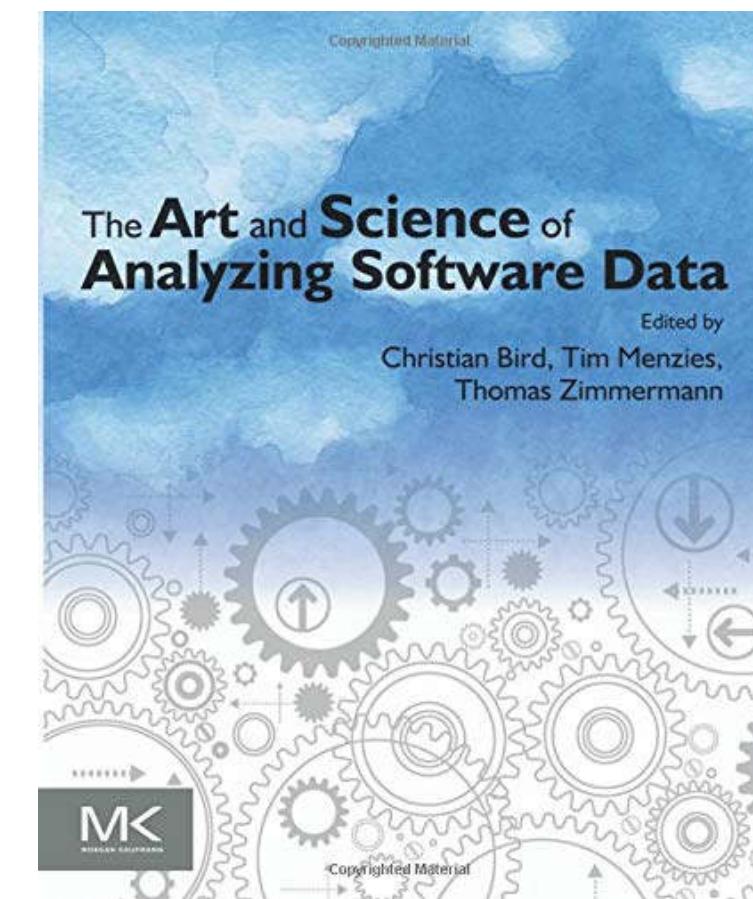
# More on Software Analytics



Adam Tornhill:  
*Software X-Ray*

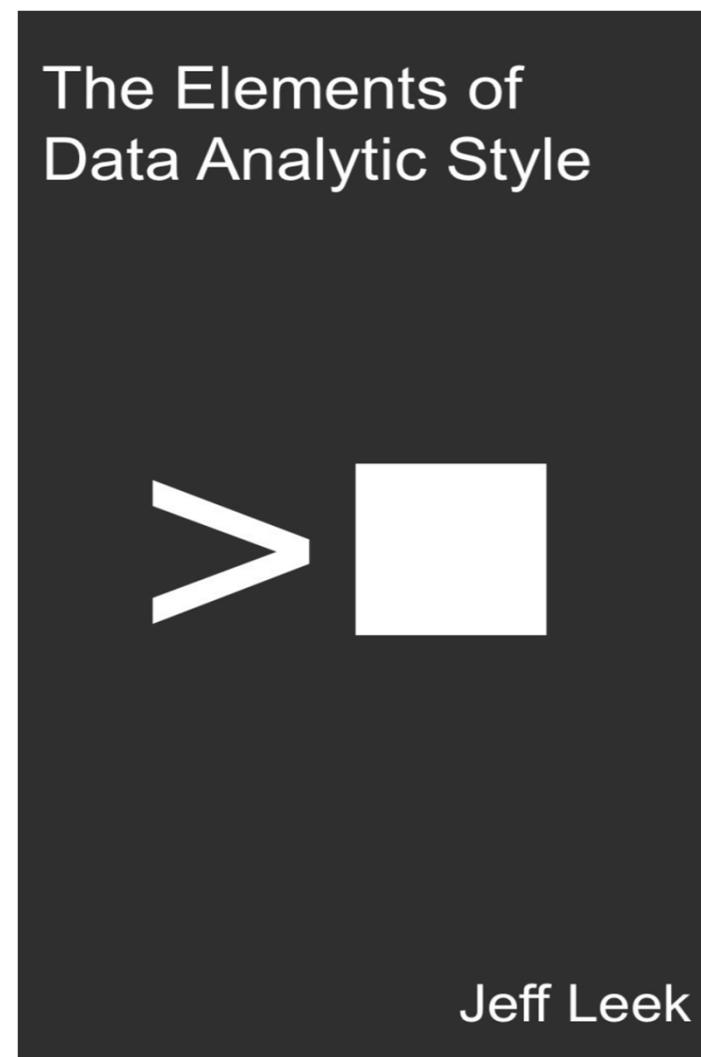


Tim Menzies, Laurie Williams,  
Thomas Zimmermann:  
*Perspectives on Data Science for  
Software Engineering*



Christian Bird, Tim Menzies,  
Thomas Zimmermann:  
*The Art and Science of Analyzing  
Software Data*

# More on Data Science

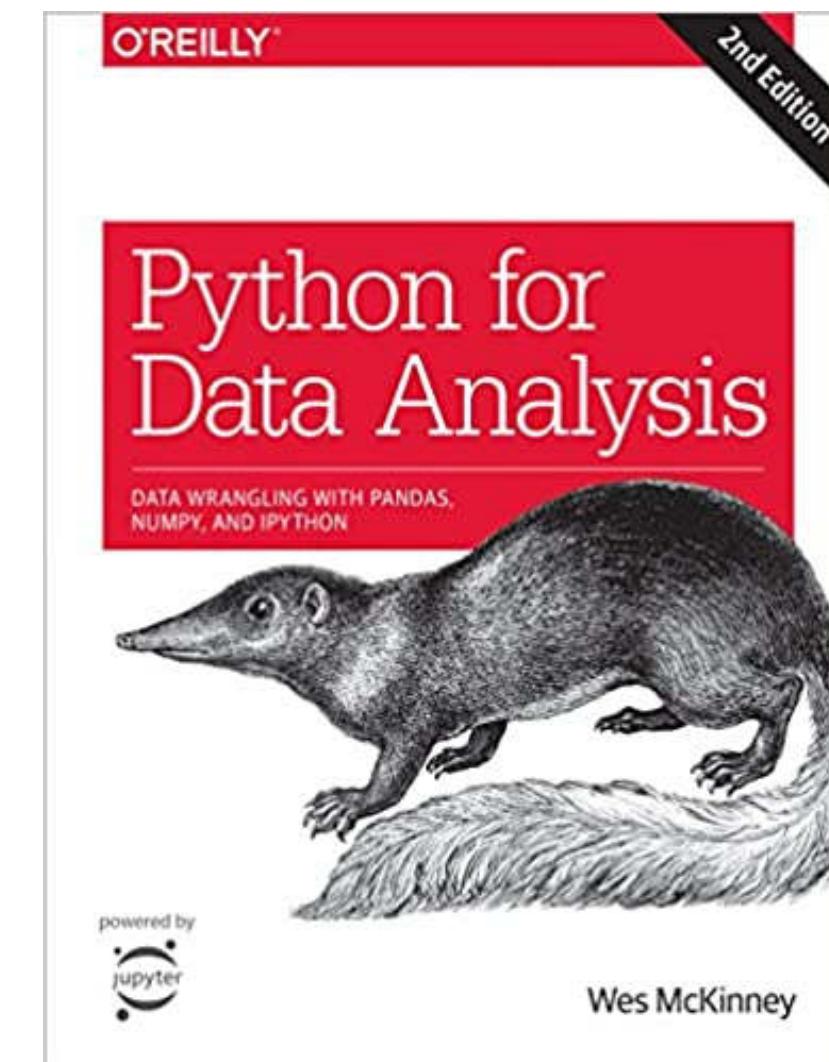


Jeff Leek:  
*The Elements of Data  
Analytic Style*

Report Writing for  
Data Science in R

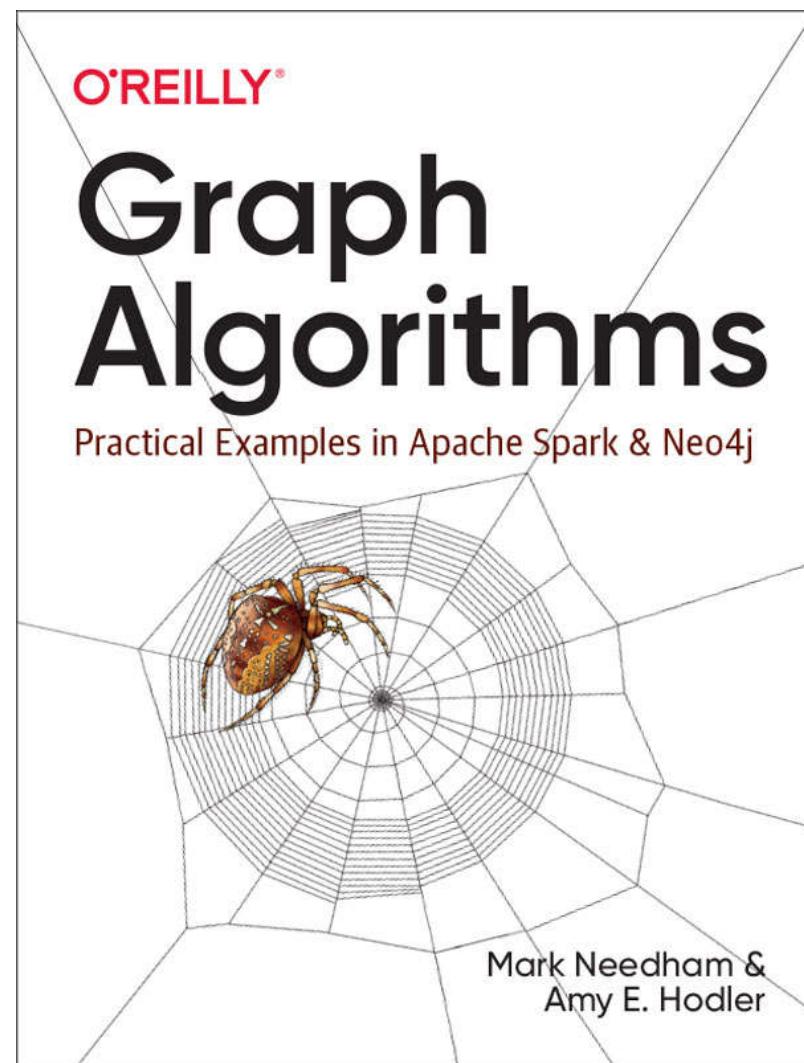


Roger D. Peng



Wes McKinney:  
*Python for Data Analysis*

# More on Graph Analytics



Mark Needham & Amy Hodler:  
*Graph Algorithms*

The image is a screenshot of the Neo4j website. At the top, there is a navigation bar with links for 'Products', 'Use Cases', 'Developers &amp; Data Scientists', 'Pricing', 'Learn', 'Contact Us', and a 'Get Started Free' button. The main content area has a light yellow background. On the left, there is a section titled 'WHAT IS IT?' with the heading 'Neo4j Graph Data Science'. Below this, a paragraph describes Graph Data Science as an analytics and machine learning solution. To the right of the text is a 3D isometric illustration of various data science and business intelligence components connected by lines, including a central cylinder labeled '2024', a bar chart with '+15%', a clock, a brain, a pie chart, and a network graph.

<https://neo4j.com/product/graph-data-science/>

# Paper about 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 important questions in this respect [2].

# Thank you very much!

Markus Harrer

markus.harrer@innoq.com

 @feststelltaste

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 Munich  
Germany

innoQ Schweiz GmbH

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

Albulastr. 55  
8048 Zurich  
Switzerland