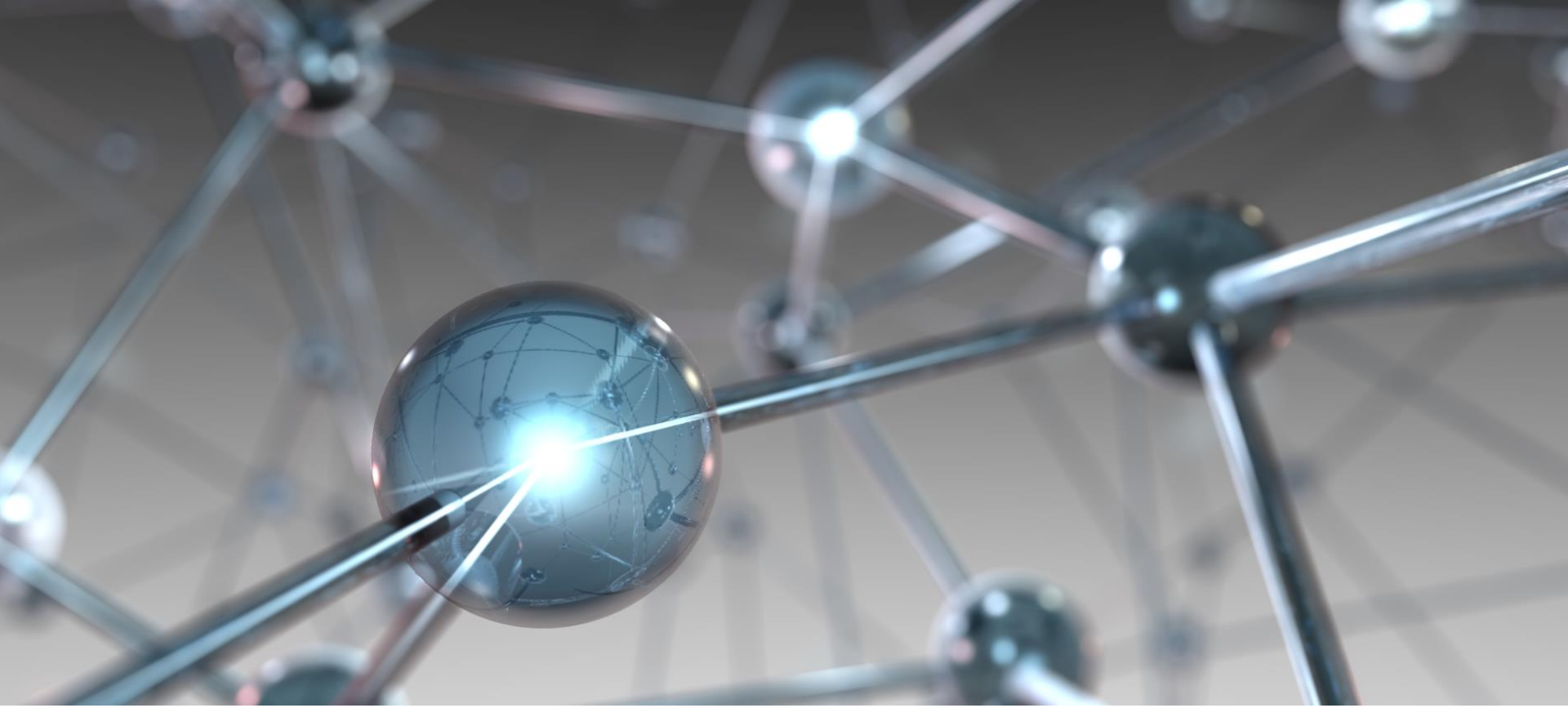
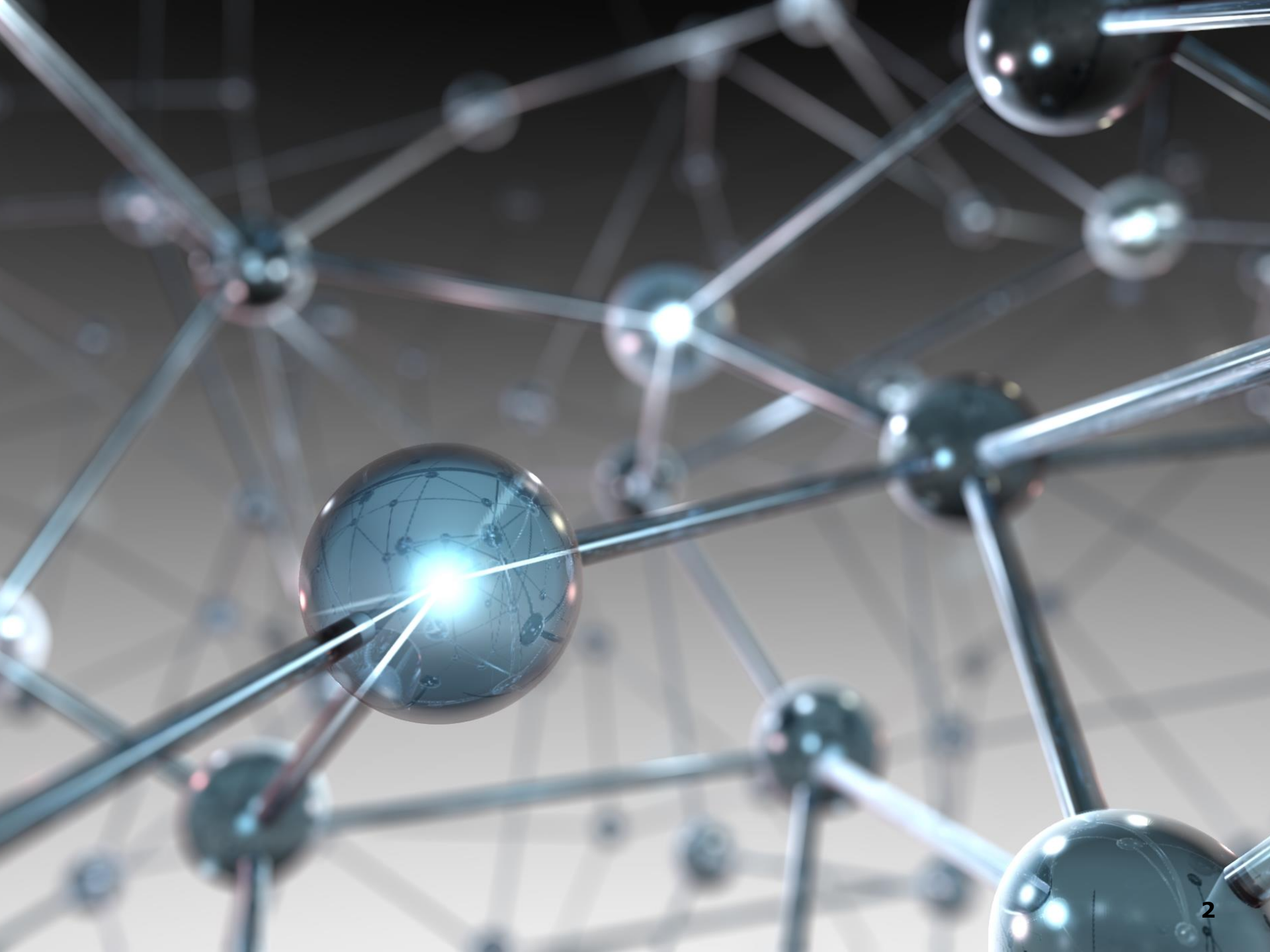


Transparenz durch Visualisierung: KI verständlich machen

Prof. Dr. Fabian Beck

4. Juli 2019

PALUNO
The Ruhr Institute for Software Technology



Die Realität

```python
import numpy as np

def nonlin(x,deriv=False):
    if(deriv==True):
        return x*(1-x)

    return 1/(1+np.exp(-x))

X = np.array([[0,0,1],
              [0,1,1],
              [1,0,1],
              [1,1,1]])

y = np.array([[0],
              [1],
              [1],
              [0]])

np.random.seed(1)

# randomly initialize our weights with mean 0
syn0 = 2*np.random.random((3,4)) - 1
syn1 = 2*np.random.random((4,1)) - 1

for j in xrange(60000):

    # Feed forward through layers 0, 1, and 2
    l0 = X
    l1 = nonlin(np.dot(l0,syn0))
    l2 = nonlin(np.dot(l1,syn1))

    # how much did we miss the target value?
    l2_error = y - l2

    if (j% 10000) == 0:
        print "Error:" + str(np.mean(np.abs(l2_error)))

    # in what direction is the target value?
    # were we really sure? if so, don't change too much.
    l2_delta = l2_error*nonlin(l2,deriv=True)

    # how much did each l1 value contribute to the l2 error (according
       to the weights)?
    l1_error = l2_delta.dot(syn1.T)

    # in what direction is the target l1?
    # were we really sure? if so, don't change too much.
    l1_delta = l1_error * nonlin(l1,deriv=True)

    syn1 += l1.T.dot(l2_delta)
    syn0 += l0.T.dot(l1_delta)
```

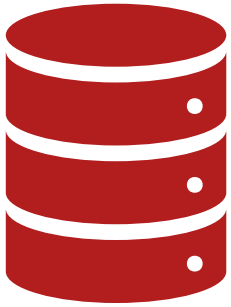
Programmcode (Python) für ein einfaches neuronales Netz

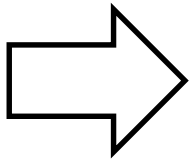
https://iamtrask.github.io/2015/07/12/basic-python-network/

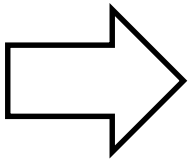
Verfahren der Künstlichen Intelligenz

Visualisierung schafft Verständlichkeit und Transparenz

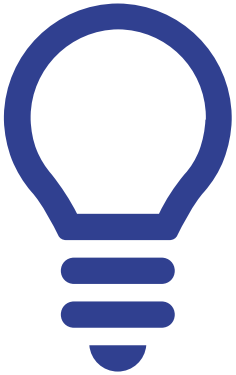
Daten





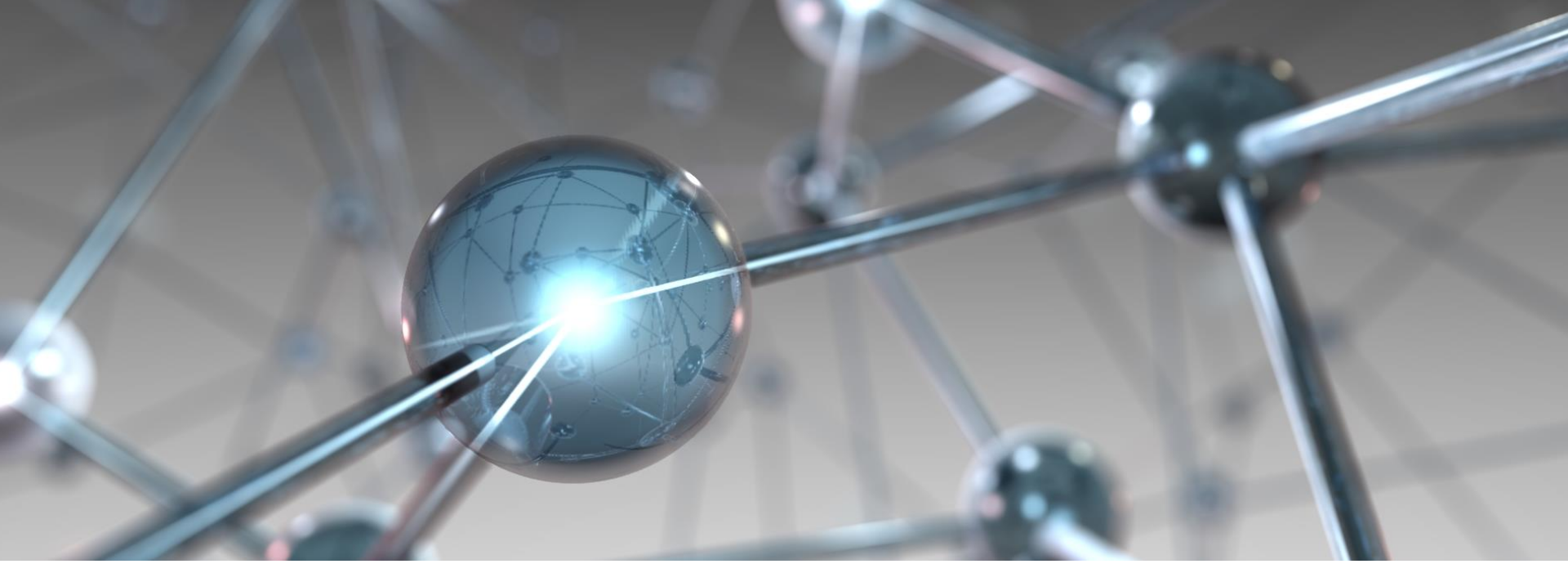


Ergebnisse



Die „Black Box“
Künstliche Intelligenz

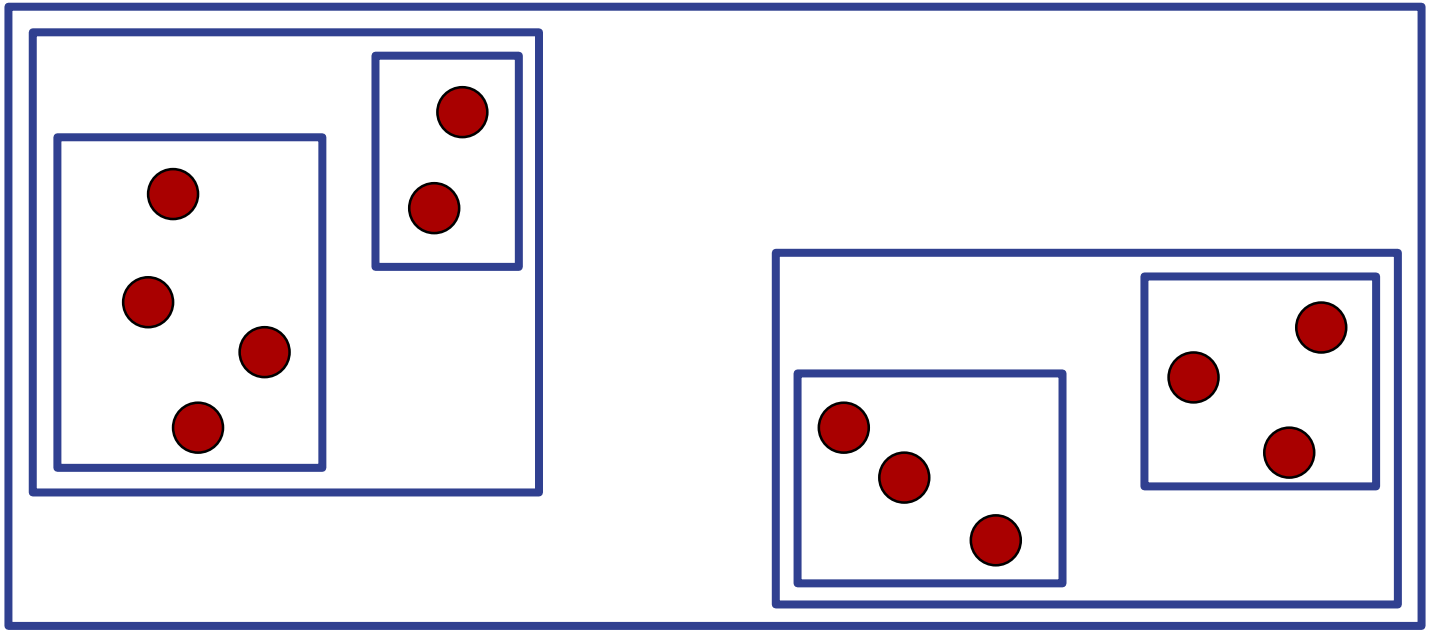
- komplex
- intransparent



Wie **maschinelles Lernen** und **Visualisierung** bei der **Entwicklung von Software** helfen

Clustering

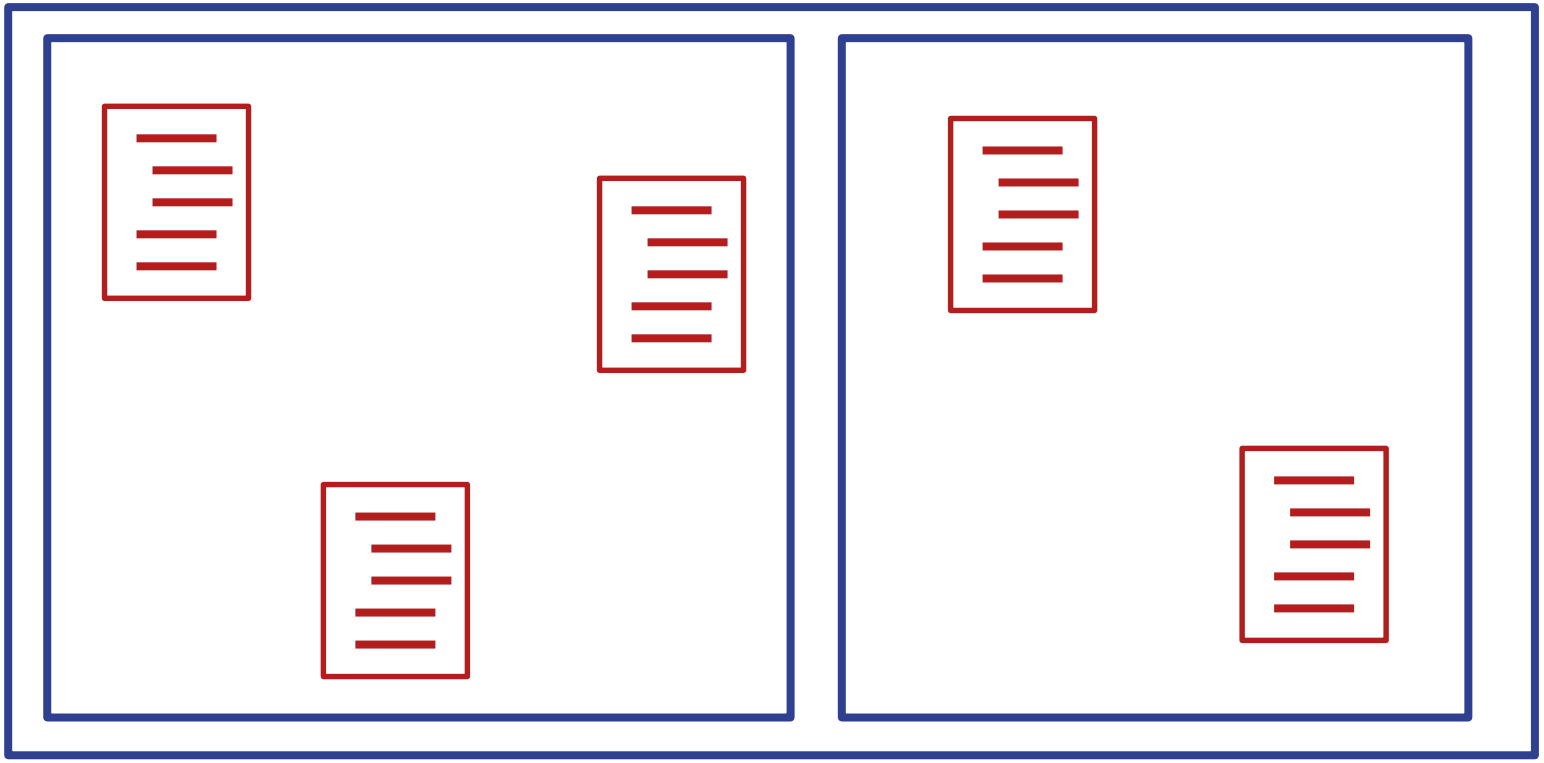
Ziel: Gruppierung von Elementen ● aufgrund ihrer Ähnlichkeit (hier: räumliche Nähe)



„Unüberwachtes Lernen“
(Kategorien sind nicht vorgegeben)

Modulare Software

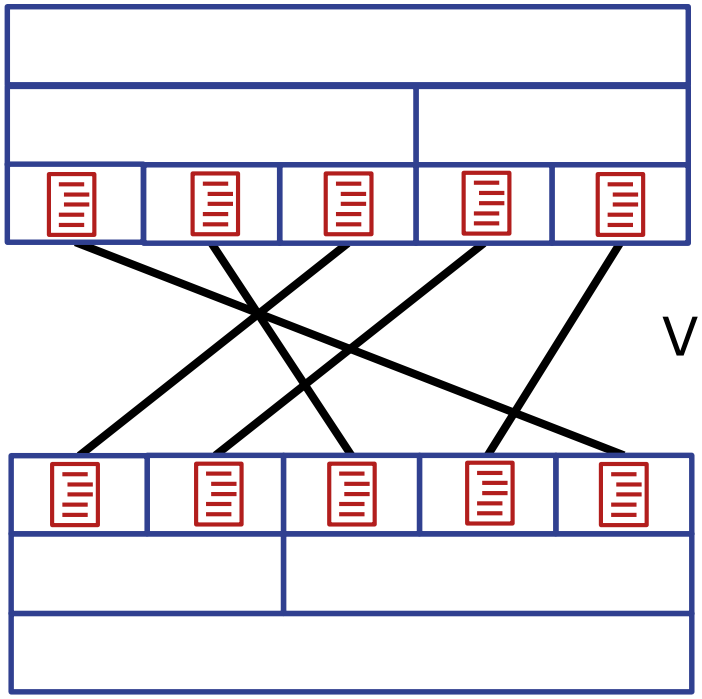
Module helfen Software-Entwicklern, einen Überblick über den großen Programmcode zu behalten



→ Finde eine gute Gruppierung (Modulstruktur)

Verbesserung einer Modulstruktur

Aktuelle Modulstruktur

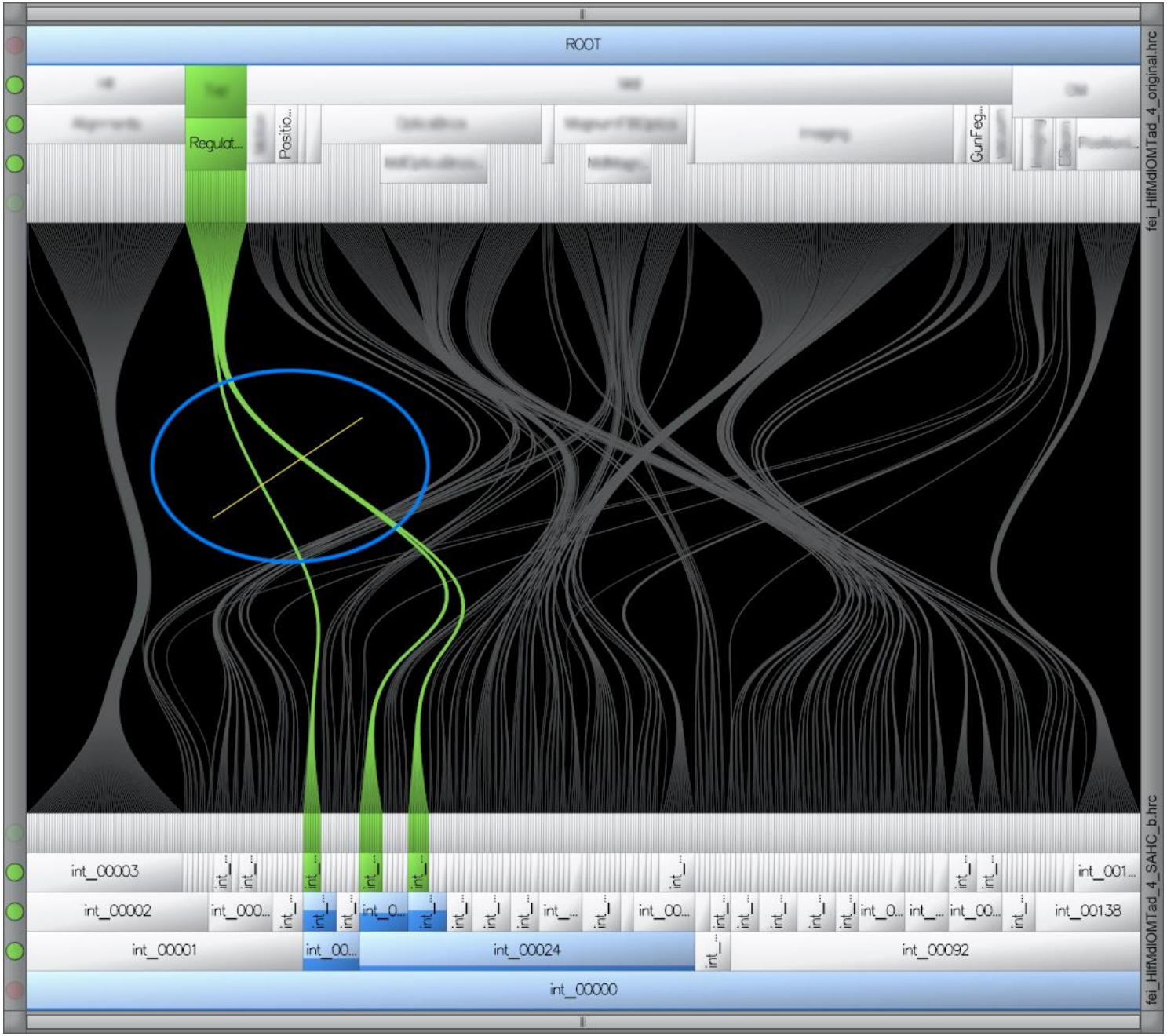


Clustering-Ergebnis

Es existiert bereits eine Modulstruktur

Visualisierung zeigt Unterschiede

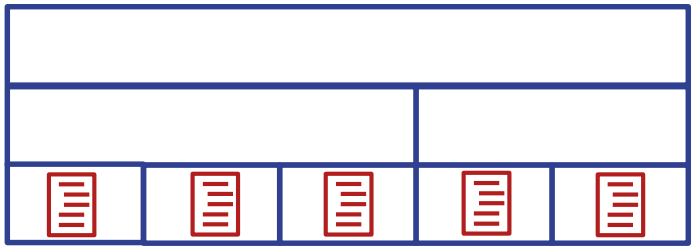
Clustering soll **Verbesserungsvorschläge** liefern, statt eine komplette Änderung



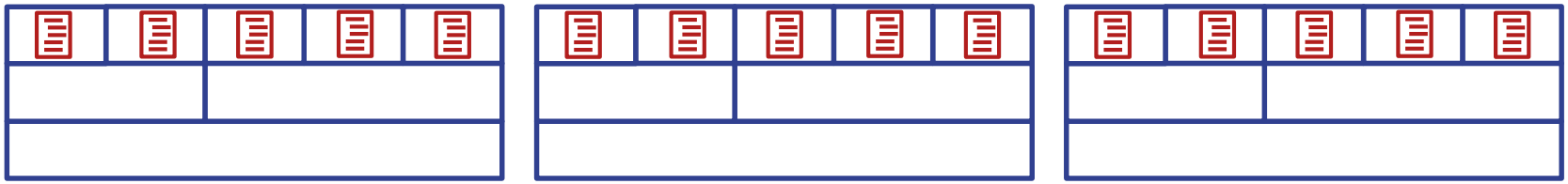
Holten, D. and van Wijk, J. J., 2008. Visual Comparison of Hierarchically Organized Data. In *Computer Graphics Forum* (Vol. 27, No. 3, pp. 759–766). DOI: 10.1111/j.1467-8659.2008.01205.x.

Verschiedene Kriterien zum Clustern

Aktuelle Modulstruktur



Visualisierung?



Alternative Clustering-Ergebnisse

Aktuelle Modulstruktur

ClusterCompare PMD https://clustercompare.jan-melcher.de

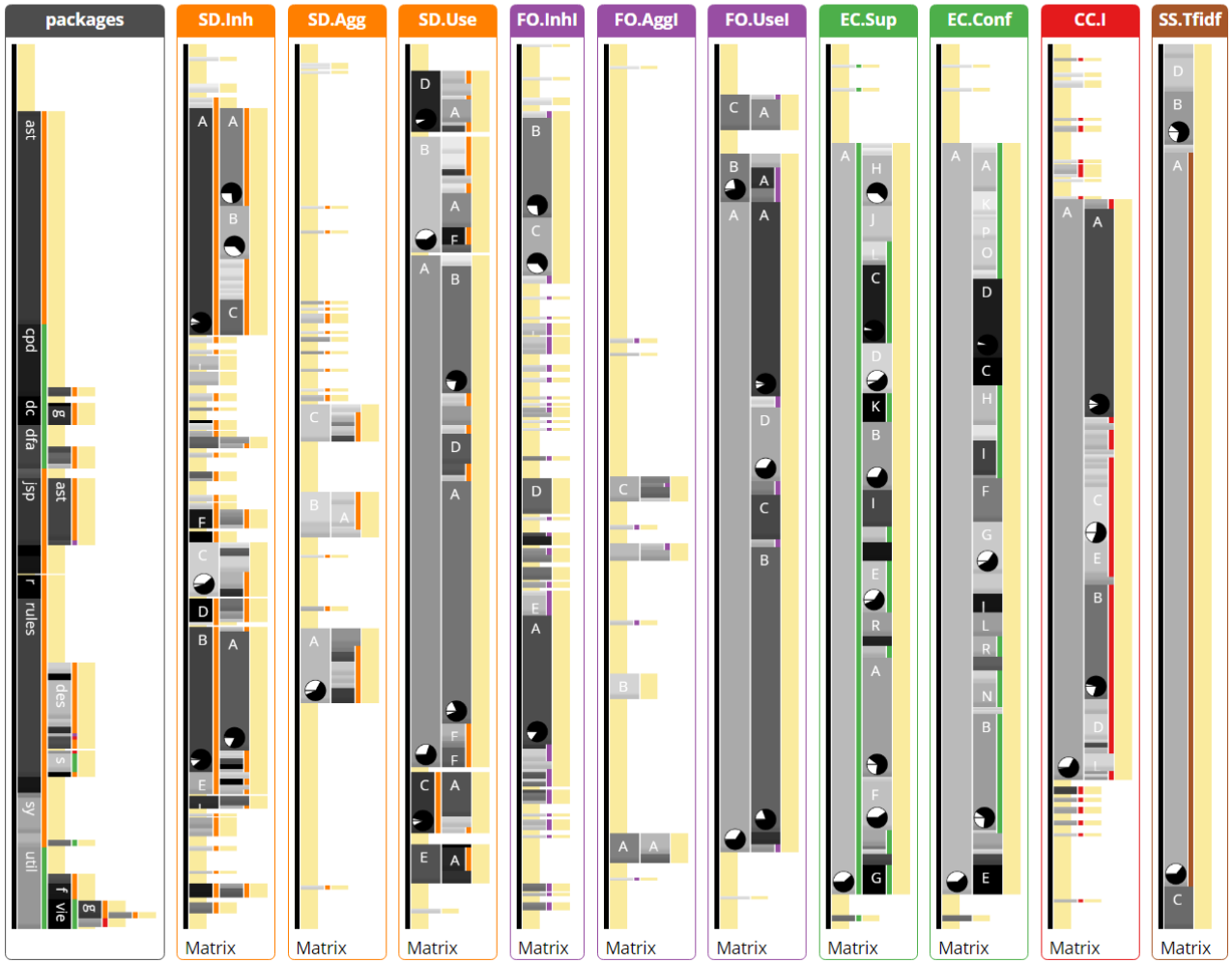
Beck, F., Melcher, J., and Weiskopf, D., 2016. Identifying Modularization Patterns by Visual Comparison of Multiple Hierarchies. In *2016 IEEE 24th International Conference on Program Comprehension*. DOI: 10.1109/ICPC.2016.7503712.

Clustering-Ergebnis

Beck, F., Melcher, J., and Weiskopf, D., 2016. Identifying Modularization Patterns by Visual Comparison of Multiple Hierarchies. In *2016 IEEE 24th International Conference on Program Comprehension*. DOI: 10.1109/ICPC.2016.7503712.

Alternative Clustering-Ergebnisse

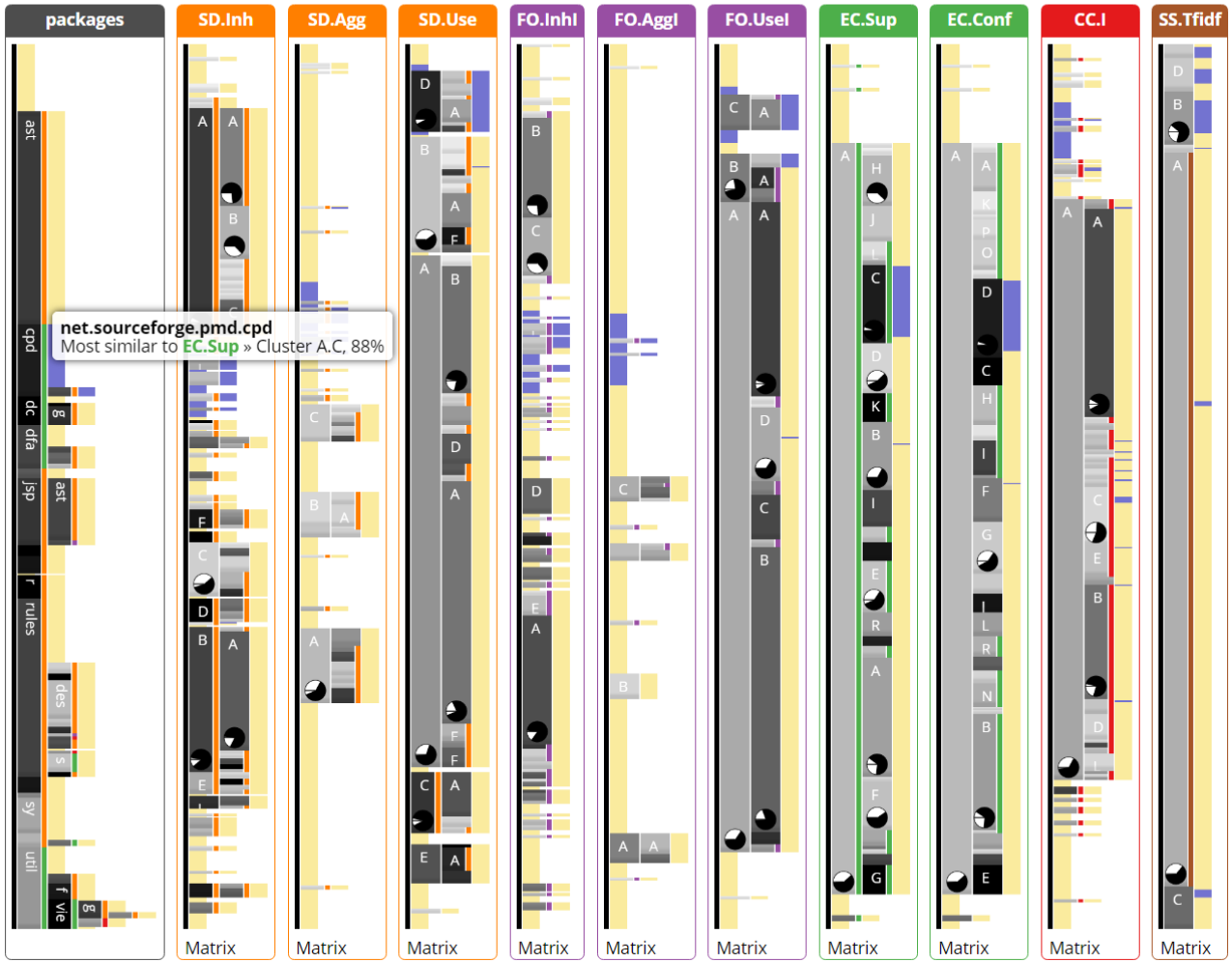
ClusterCompare PMD https://clustercompare.jan-melcher.de



Beck, F., Melcher, J., and Weiskopf, D., 2016. Identifying Modularization Patterns by Visual Comparison of Multiple Hierarchies. In *2016 IEEE 24th International Conference on Program Comprehension*. DOI: 10.1109/ICPC.2016.7503712.

Interaktiver Vergleich

ClusterCompare PMD https://clustercompare.jan-melcher.de



46 classes selected

cpd.Match
cpd.cppast.TokenMgrError
cpd.MatchAlgorithm
cpd.AnyLanguage
cpd.CPD
cpd.CPDListener
cpd.CPDNullListener
cpd.CPDTask
cpd.CPPLanguage
cpd.CPPTokenizer
cpd.CSVRenderer

Most similar clusters:

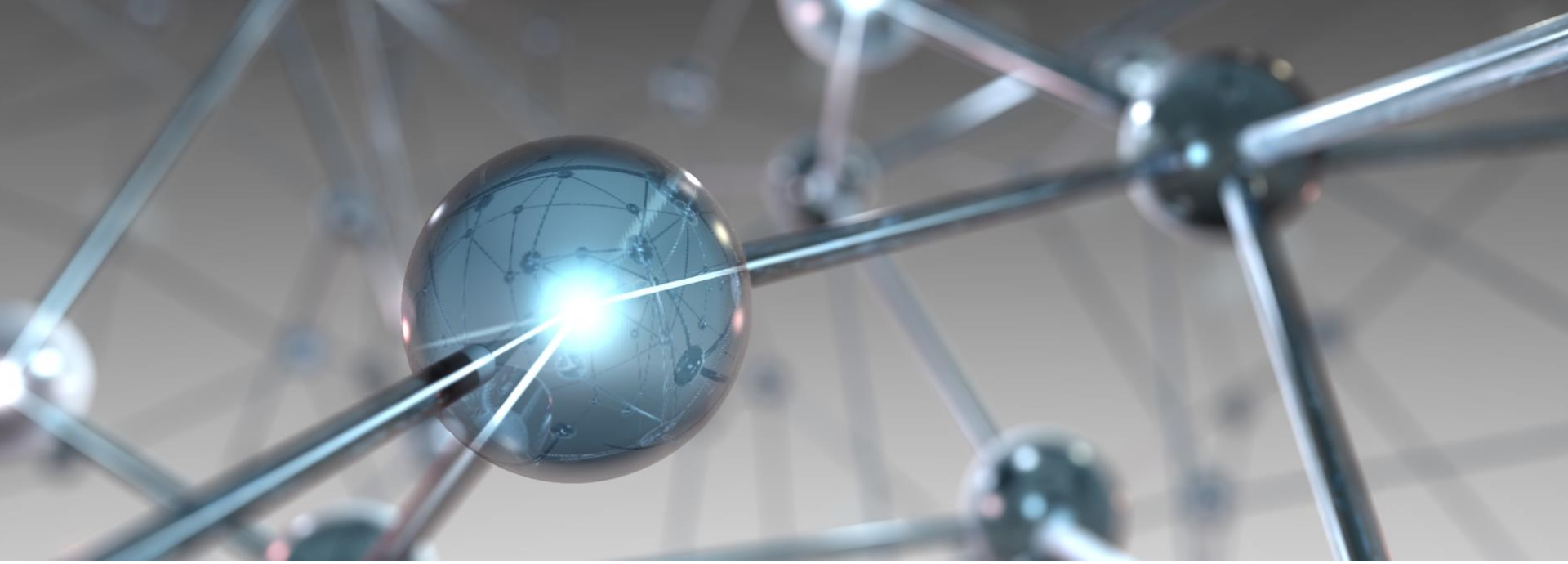
						#
100%	=	46	/	46	net.sourceforge.pmd.cpd	46
88%	=	45	/	51	EC.Conf » Cluster A.D	50
88%	=	45	/	51	EC.Sup » Cluster A.C	50
85%	=	39	/	46	SD.Use » Cluster D	39
50%	=	23	/	46	FO.Usel » Cluster C	23
43%	=	20	/	46	FO.Usel » Cluster C.A	20
36%	=	21	/	59	SS.Tfidf » Cluster B	34
20%	=	9	/	46	SD.Inh » Cluster I	9
17%	=	8	/	46	FO.Inhl » Cluster I	8
13%	=	6	/	46	SD.Agg » Cluster E	6
7%	=	3	/	46	FO.Aggl » Cluster G	3
5%	=	4	/	79	CC.I » Cluster A.C	37

Beck, F., Melcher, J., and Weiskopf, D., 2016. Identifying Modularization Patterns by Visual Comparison of Multiple Hierarchies. In *2016 IEEE 24th International Conference on Program Comprehension*. DOI: 10.1109/ICPC.2016.7503712.

Transparenz durch Visualisierung

Ergebnis I

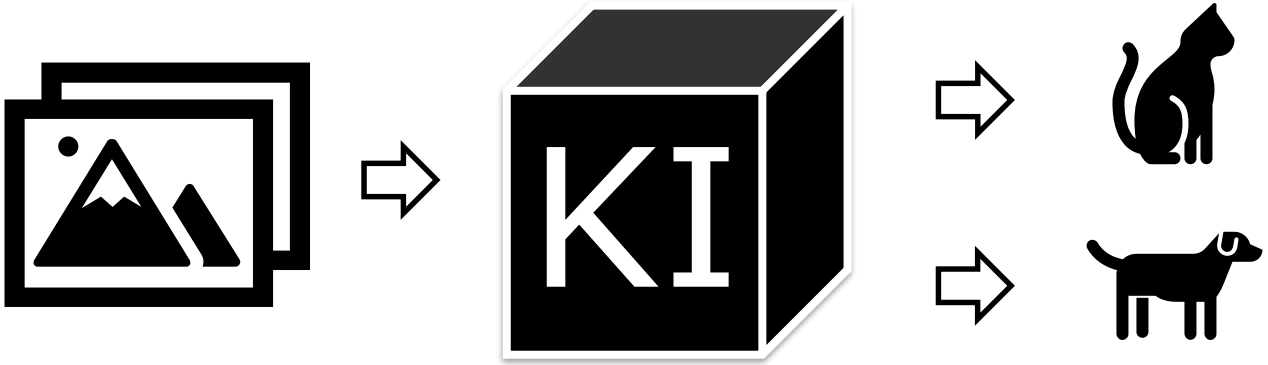
Visualisierung unterstützt Software-Entwickler, maschinelles Lernen zur Strukturierung des Programmcodes einzusetzen



Wie **Visualisierungen** bei der **Entwicklung einer KI-Anwendung** helfen

Klassifikation von Bildern

„Überwachtes Lernen“
(Kategorien sind vorgegeben)

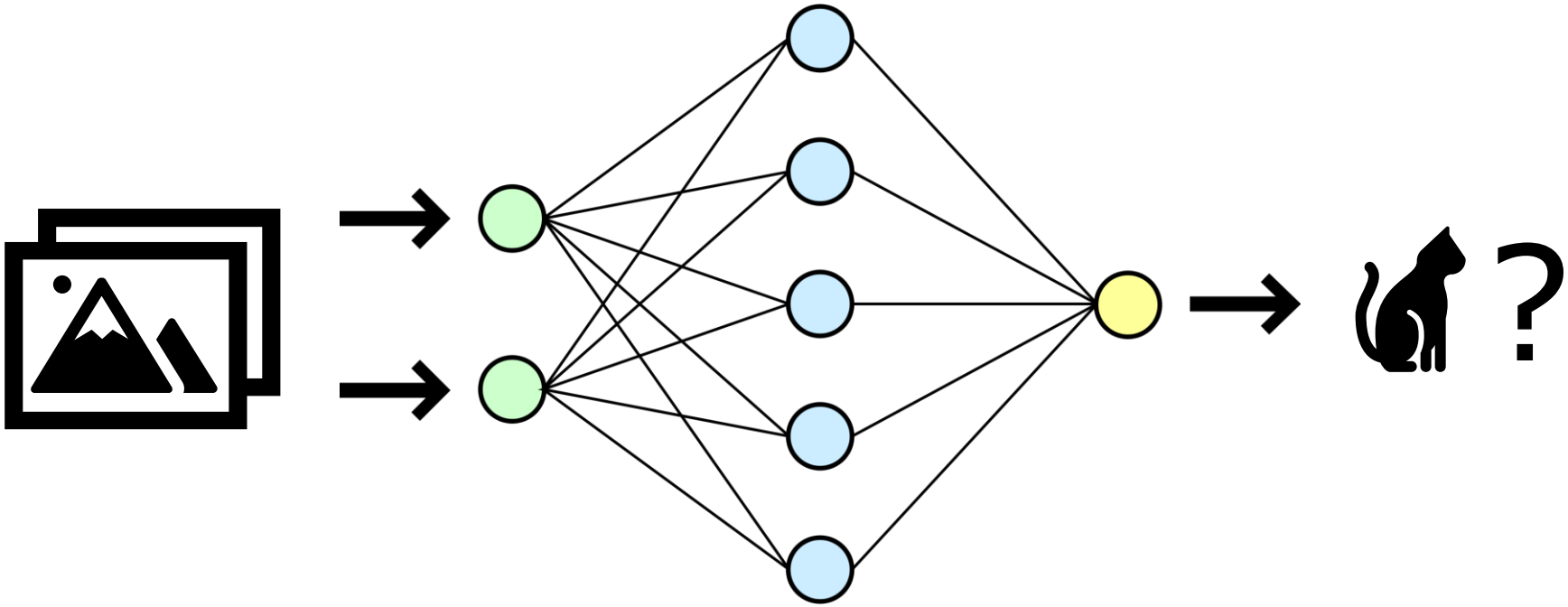


Bildersuche

Unterstützung
von Sehbehinderten

Autonomes
Fahren

Künstliches Neuronales Netz

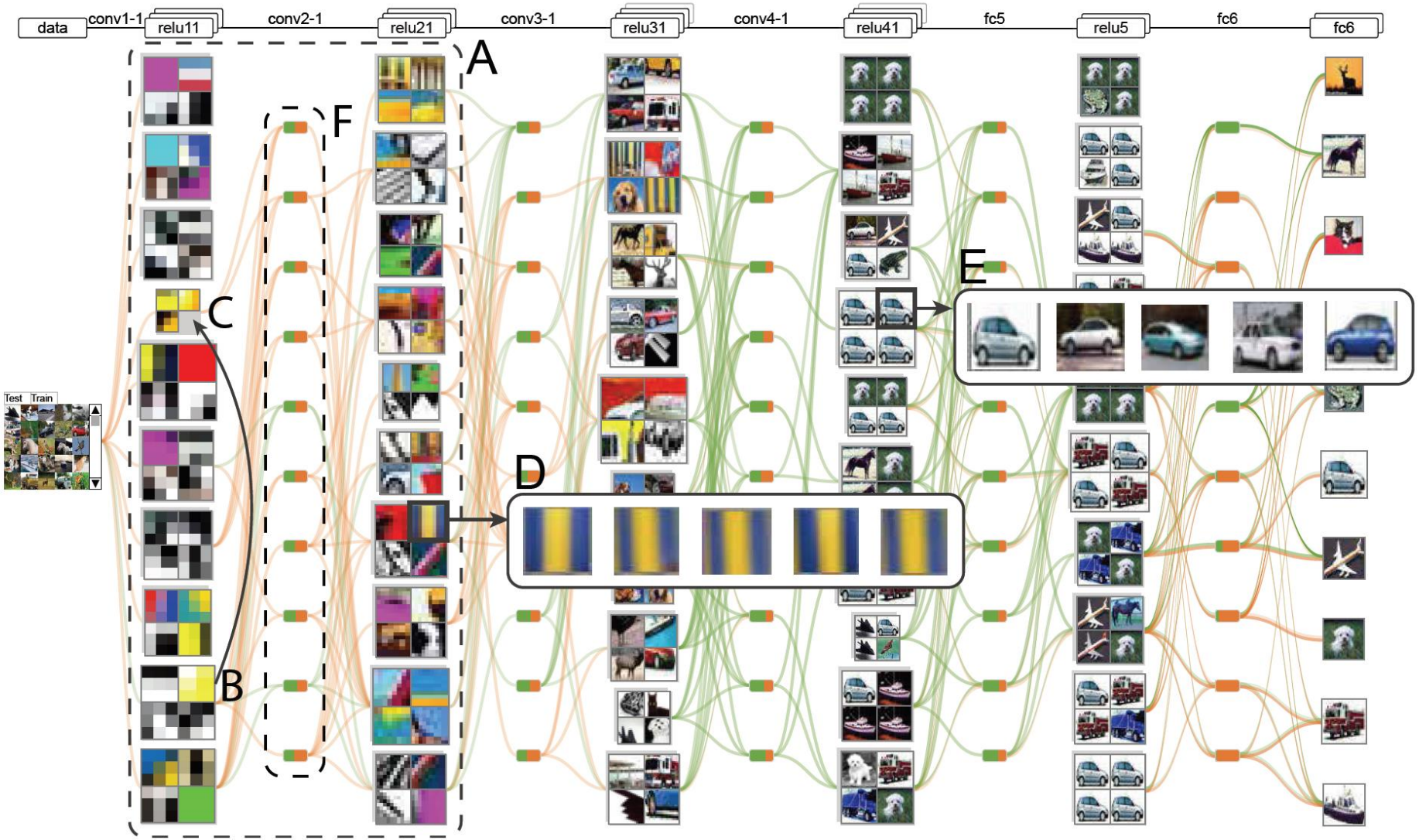


https://de.wikipedia.org/wiki/Neuronales_Netz#/media/Datei:Neural_network.svg

Deep Learning: viele Zwischenschichten (und viele Neuronen)

→ Entwickler: Probleme sind schwer nachvollziehbar

CNNVis: Convolutional Neural Network



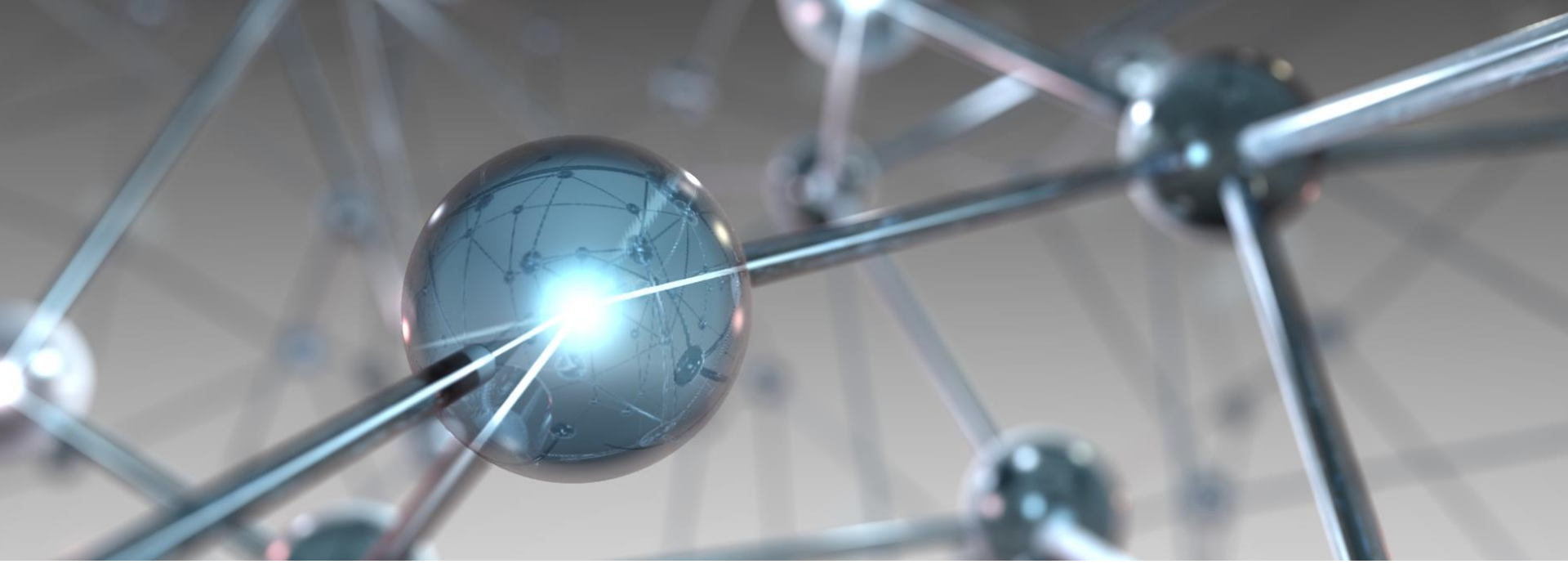
Liu, M., Shi, J., Li, Z., Li, C., Zhu, J., & Liu, S. (2016). Towards better analysis of deep convolutional neural networks. *IEEE Transactions on Visualization and Computer Graphics*, *23*(1), 91-100.

http://shixialiu.com/publications/cnnvis/demo/

Transparenz durch Visualisierung

Ergebnis II

Visualisierung hilft bei der Entwicklung und Optimierung von Verfahren der Künstlichen Intelligenz



Transparenz durch Visualisierung: **Chancen und Herausforderungen**

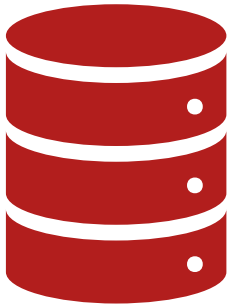
Chancen

Rechenpower

Menschliches
Urteilsvermögen
miteinbeziehen

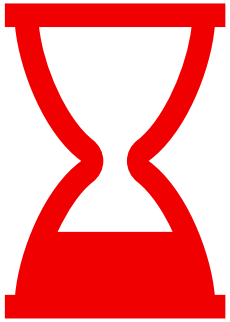
Entscheidungen
nachvollziehbar und
KI anwendbar machen

Herausforderungen



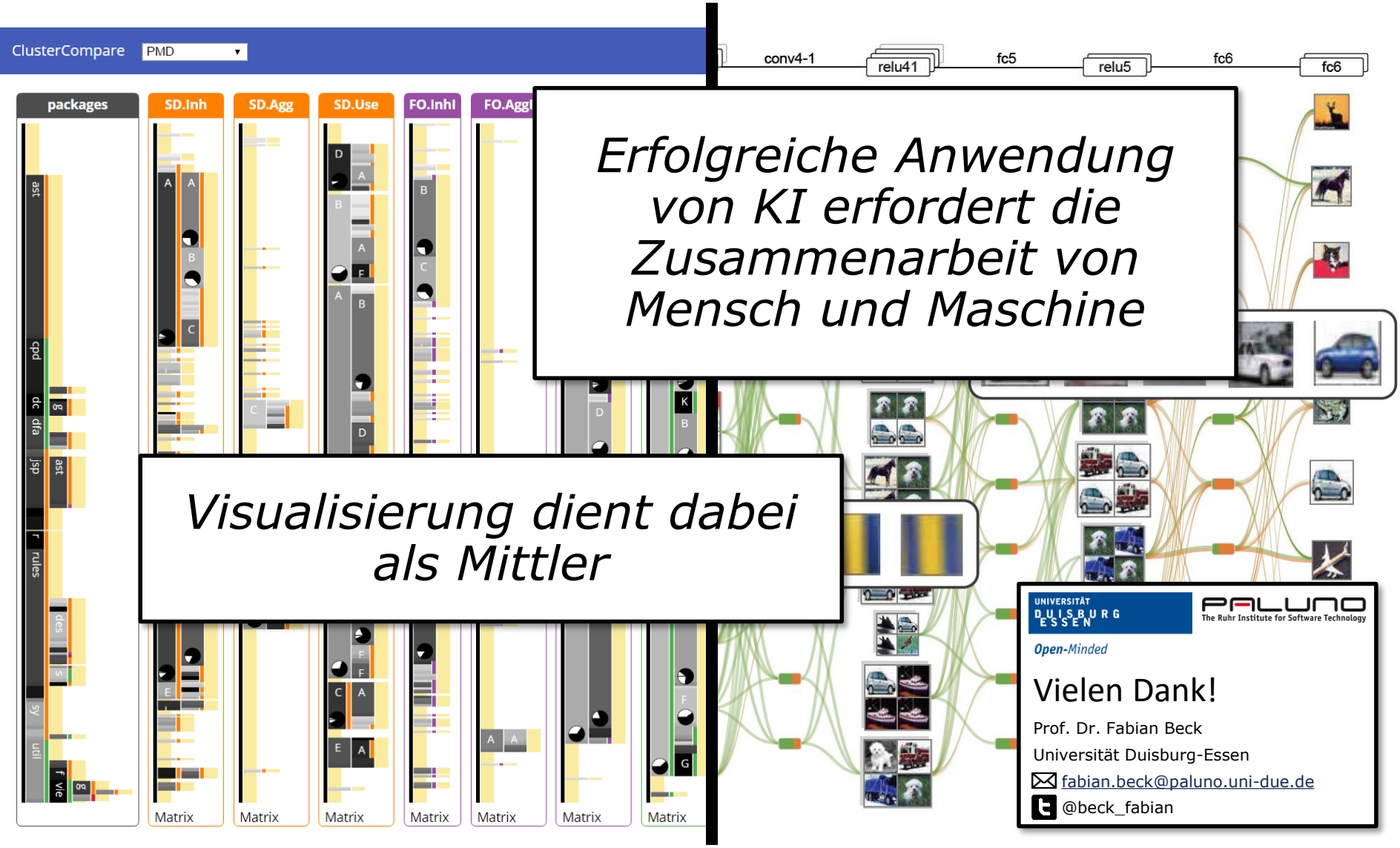
Große
Datenmengen

Menschliche
Informationsverarbeitung
begrenzt



Effizient
Entscheidungen
treffen

Transparenz durch Visualisierung: KI verständlich machen



Erfolgreiche Anwendung von KI erfordert die Zusammenarbeit von Mensch und Maschine

Visualisierung dient dabei als Mittler

UNIVERSITÄT DUISBURG ESSEN
Open-Minded

PALUNO
The Ruhr Institute for Software Technology

Vielen Dank!

Prof. Dr. Fabian Beck
Universität Duisburg-Essen
fabian.beck@paluno.uni-due.de
@beck_fabian