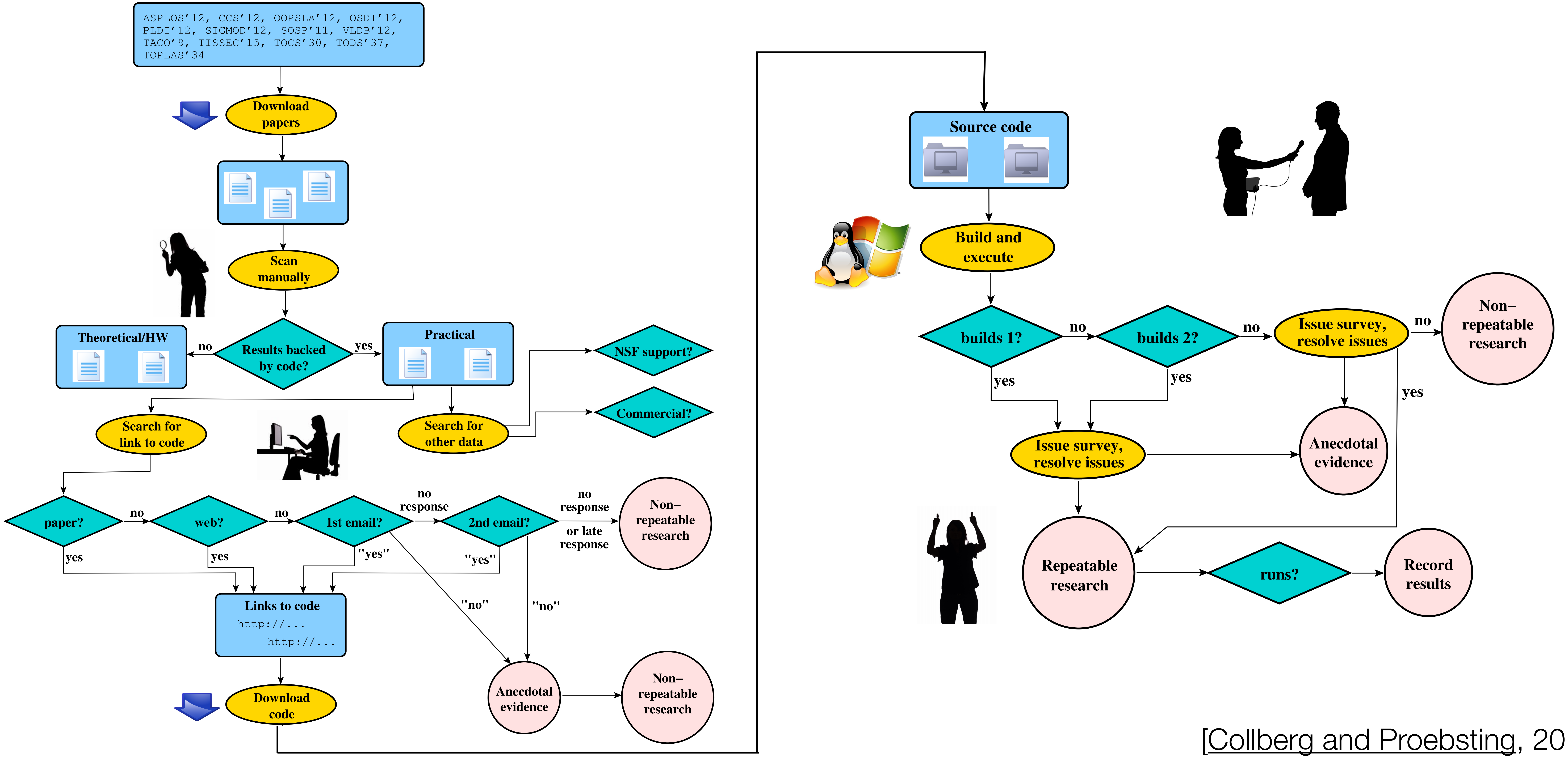
Advanced Data Management (CSCI 490/680)

Machine Learning in Databases

Dr. David Koop

Checking Computational Results in Systems



[Collberg and Proebsting, 2015]

Repeatability Results

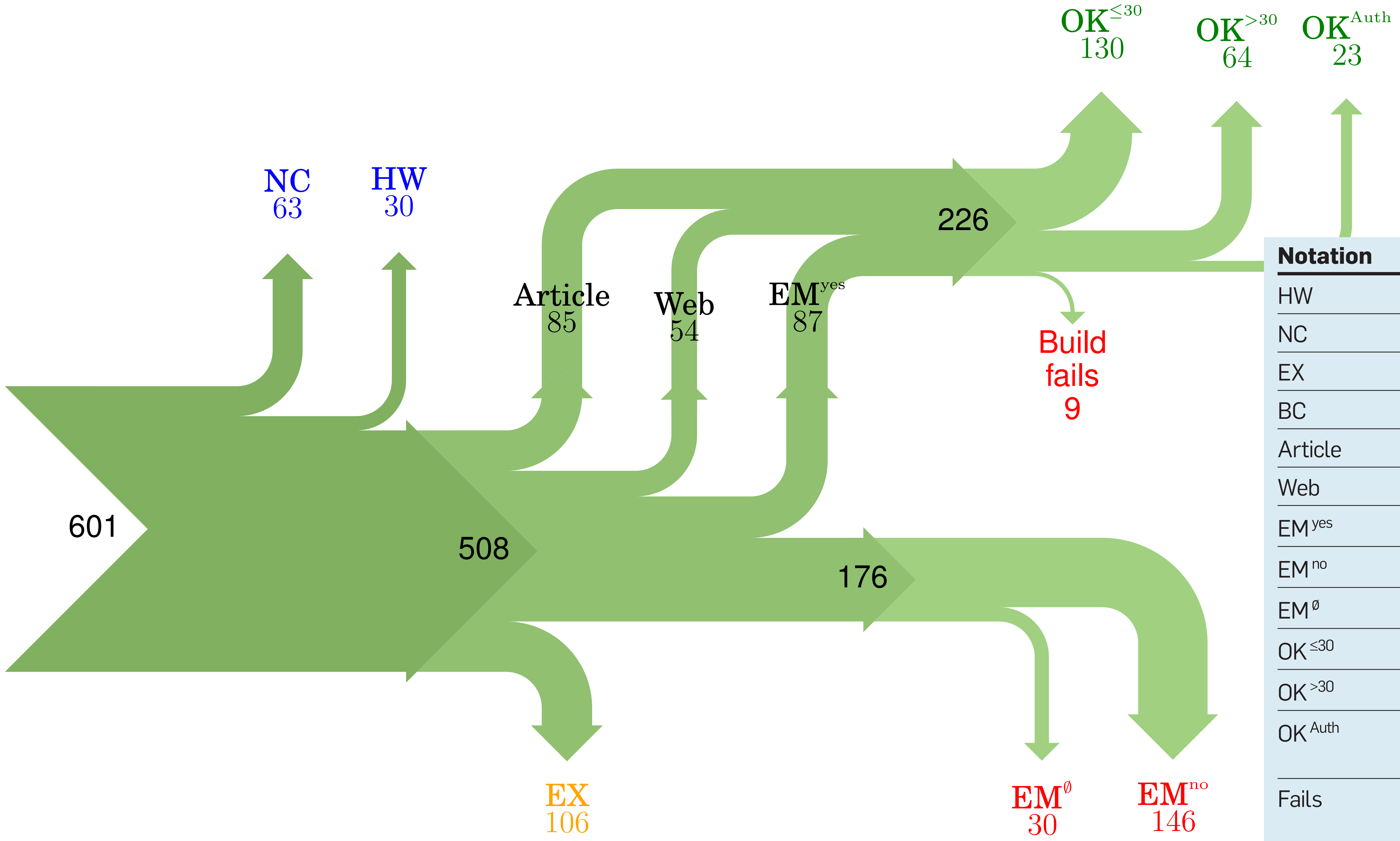


Figure 11: Study result. Blue numbers represent papers that were excluded from consideration, green numbers papers that are weakly repeatable, red numbers papers that are non-weakly repeatable, and orange numbers represent papers that were excluded (due to our restriction of sending at most one email to each author).

Notation	Number of papers ...
HW	excluded due to replication requiring special hardware
NC	excluded due to results not being backed by code
EX	excluded due to overlapping author lists
BC	where the results are backed by code
Article	where code was found in the paper itself
Web	where code was found through a Web search
EM^{yes}	where the author provides code after receiving an email message
EM^{no}	where the author responds to an email message saying code cannot be provided
$EM^{\emptyset}$	where the author does not respond to email requests within two months
$OK^{\leq 30}$	where code is available and we succeed in building the system in ≤30 minutes
$OK^{>30}$	where code is available and we succeed in building the system in >30 minutes
OK^{Auth}	where code is available and we fail to build, and the author says the code builds with reasonable effort
Fails	where code is available and we fail to build, and the author says the code may have problems building

[Collberg and Proebsting, 2015]

Excuses for not sharing

- Versioning
- Available Soon
- No Intention to Share
- Personnel Issues
- Lost Code
- Academic Tradeoffs
- Industrial Lab Tradeoffs
- Obsolete HW/SW
- Controlled Usage
- Privacy/Security
- Design Issues

[Collberg and Proebsting, 2015]

Examining 'Reproducibility in Computer Science'

- Repeat the experiment in reproducibility!
- Differences from original
- Shows issues with trying to classify experiments

Purported Not Building; Disputed; Not Checked	6%	••••••
Purported Building; Disputed; Not Checked	2%	••
Conflicting Checks!	0%	
Misclassified	1%	•
Purported Not Building But Found Building	14%	••••••••••••••
Purported Building But Found Not Building	0%	
Purported Not Building; Confirmed	0%	•
Purported Building; Confirmed	0%	•
All Others Purported Not	27%	•••••••••••••••••••••••••••

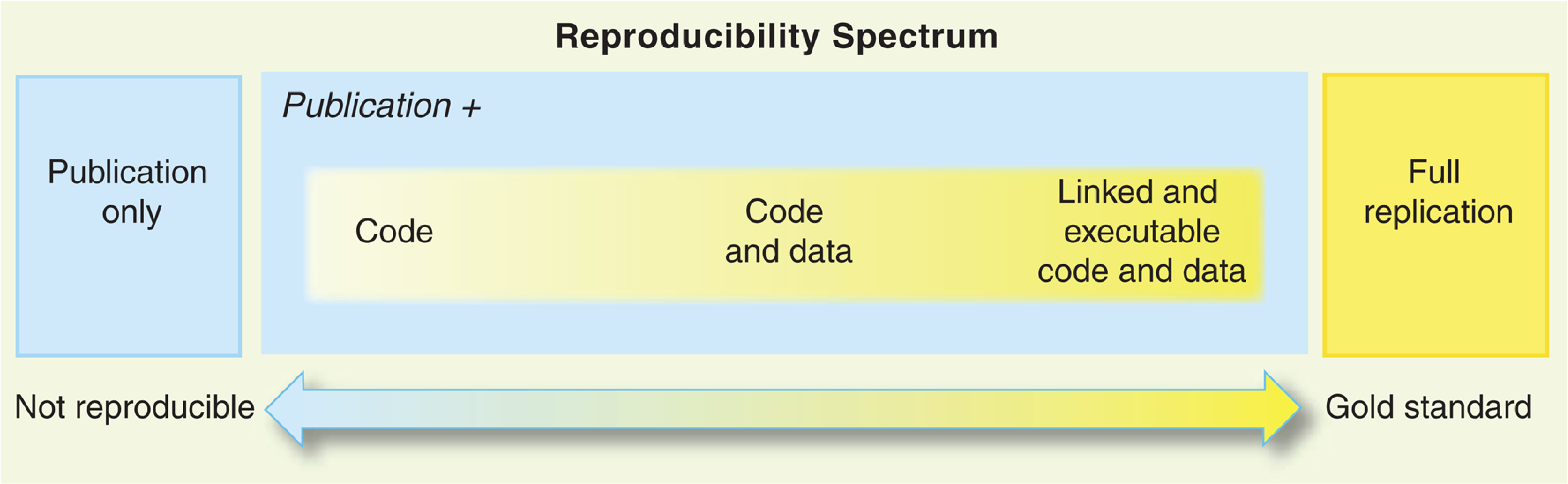
[S. Krishnamurthi et al.]

Reproducible Research

- Science is verified by replicating work independently
- Replication Issues:
  - Requires many resources to replicate (Sloan Digital Sky Survey)
  - Requires significant computing power (Climate Model Simulation)
  - Requires too much time or very specific circumstances (Environment Epidemiology)
- Reproducibility
  - Replication of the analysis based on the collected data (not replicating the data collection itself)
  - Better if we have the actual code or available executables

[R. D. Peng]

Reproducibility Spectrum



[R. D. Peng]

10 Rules for Reproducible Computational Research

- Rule 1: For Every Result, Keep Track of How It Was Produced
- Rule 2: Avoid Manual Data Manipulation Steps
- Rule 3: Archive the Exact Versions of All External Programs Used
- Rule 4: Version Control All Custom Scripts
- Rule 5: Record All Intermediate Results, When Possible in Standardized Formats

[Sandve et al., 2013]

10 Rules for Reproducible Computational Research

- Rule 6: For Analyses That Include Randomness, Note Underlying Random Seeds
- Rule 7: Always Store Raw Data behind Plots
- Rule 8: Generate Hierarchical Analysis Output, Allowing Layers of Increasing Detail to Be Inspected
- Rule 9: Connect Textual Statements to Underlying Results
- Rule 10: Provide Public Access to Scripts, Runs, and Results

[Sandve et al., 2013]

(Database) Reproducibility Research Topics

- Design and Management of Experiment Repositories
- Querying and Searching Experiments
- Mining Experiments

[J. Freire et al.]

Notebook Reproducibility

- Use notebooks from Github (~1 million)
  - Unambiguous cell order? 81.99%
- Study notebook dependencies
  - Dependencies Available? 13.72%
  - Dependencies Install? 5.03%
- Study notebook executability
  - Execute: 24.11% of unambiguous cell order
  - Matched results: 4.03%

[Pimentel et al., 2019]

Dataflow Notebooks: Resolve Notebook Ambiguities

In [d51f8eab]:
```python
import pandas as pd
df = pd.read_csv('guardian-top100-female-2019.csv')
```

df:

	Name	Rank	Position	Age on 1 Dec 2019	Nationality
0	Sam Kerr	1	Forward	26	Australia
...	...	...	...	...	...
99	Ludmila	100	Forward	25	Brazil

100 rows × 5 columns

In [full]:
```python
df = df.rename(columns={'Age on 1 Dec 2019': 'Age'})
```

df:

	Name	Rank	Position	Age	Nationality
0	Sam Kerr	1	Forward	26	Australia
...	...	...	...	...	...
99	Ludmila	100	Forward	25	Brazil

100 rows × 5 columns

In [over30]:
```python
df = df$full[df$full.Age >= 31]
```

df:

	Name	Rank	Position	Age	Nationality
2	Megan Rapinoe	3	Midfielder	34	USA
...	...	...	...	...	...
96	Cláudia Neto	97	Midfielder	31	Portugal

19 rows × 5 columns

In [under25]:
```python
df = df$full[df$full.Age <= 24]
```

df:

	Name	Rank	Position	Age	Nationality
3	Ada Hegerberg	4	Forward	24	Norway
...	...	...	...	...	...
98	Lena Oberdorf	99	Midfielder	17	Germany

25 rows × 5 columns

Assignment 5

- Four parts
  - Loading Data
  - Spatial Analysis
  - Graph Analysis
  - Temporal Analysis
- Due at the end of the semester (April 22, 2021)
- Questions?

Final Exam

- Monday, April 26, 4:00-5:50pm, Online (Blackboard)
- Similar format
- More comprehensive (questions from topics covered in Test 1 & 2)
- Will also have questions from temporal data, provenance, reproducibility, machine learning
- Bring questions on Wednesday

Improving Databases

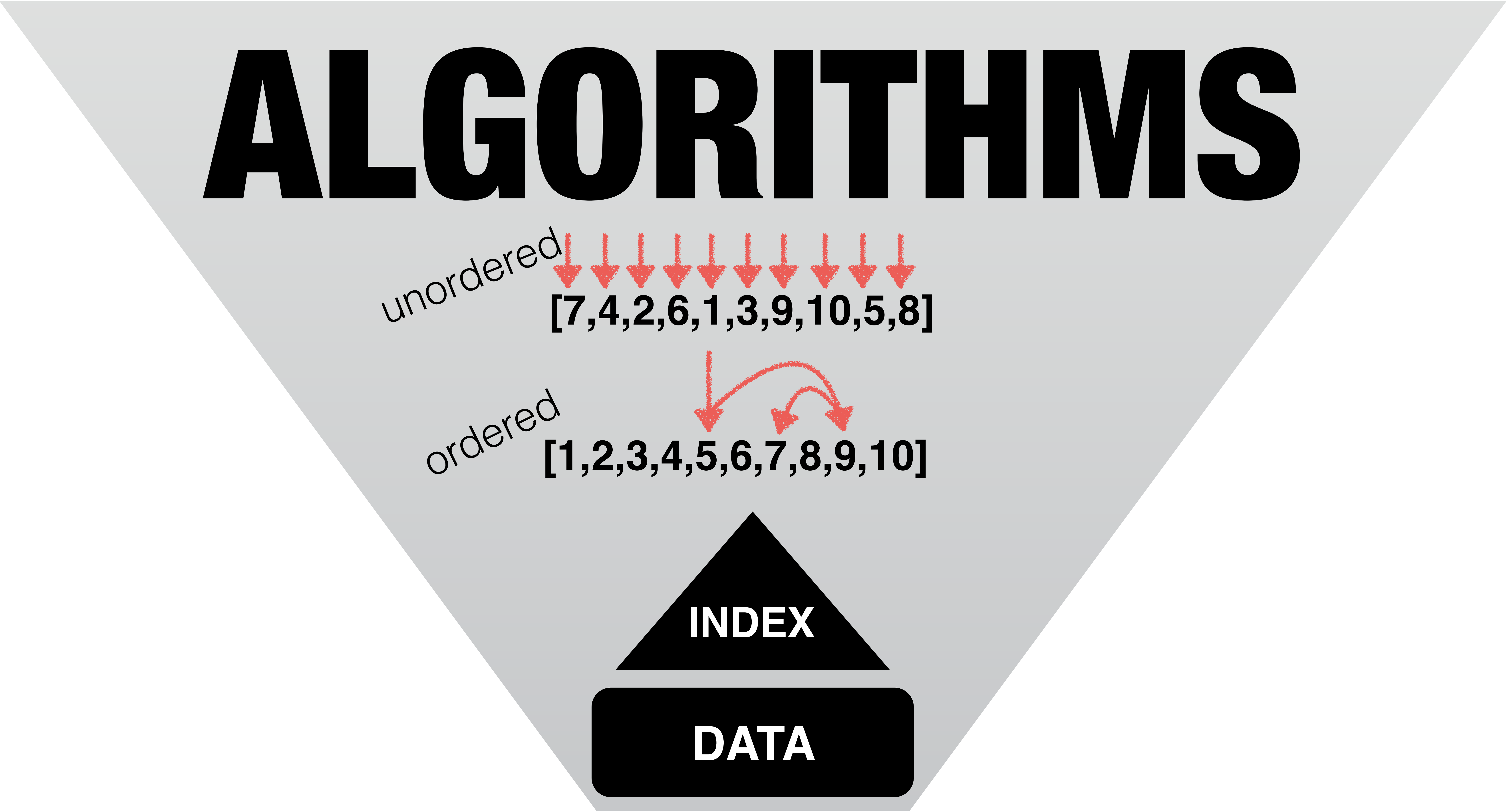
LEARNED AND **SELF-DESIGNING** DATA STRUCTURES

@ Harvard SEAS

Data Systems and AI Lab

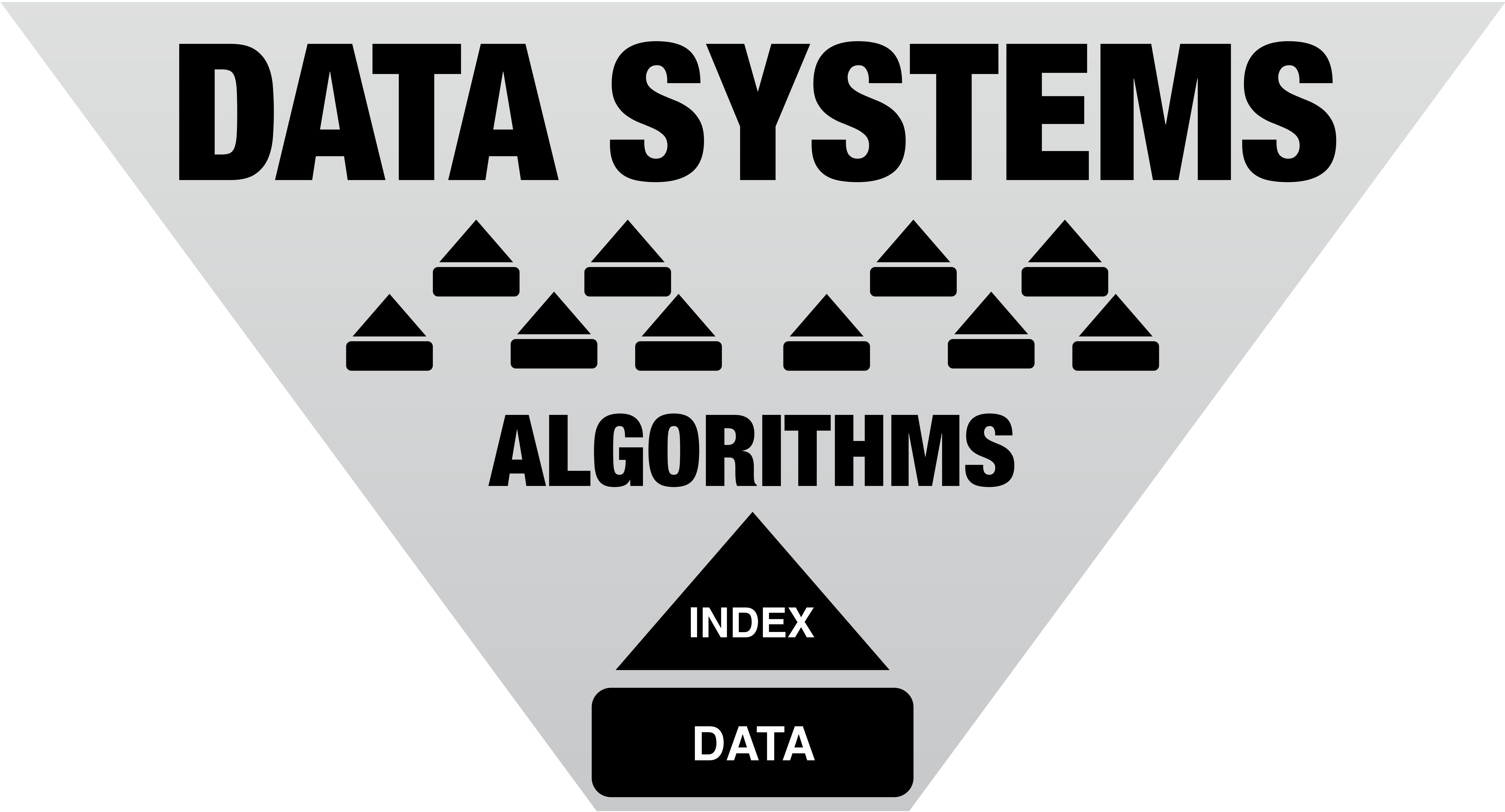
Stratos Idreos & Tim Kraska

Algorithms rely on the order of data



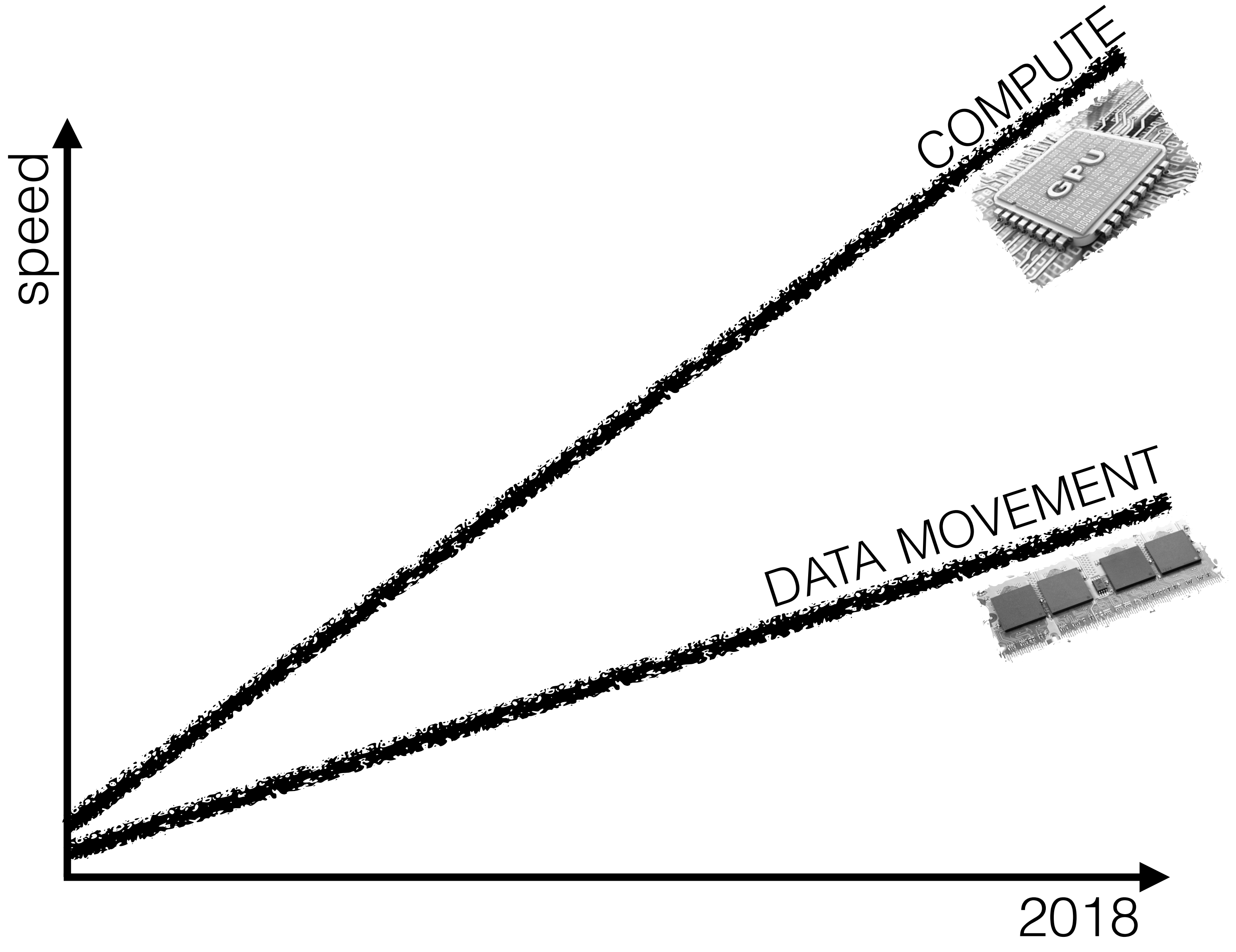
[S. Idreos, 2019]

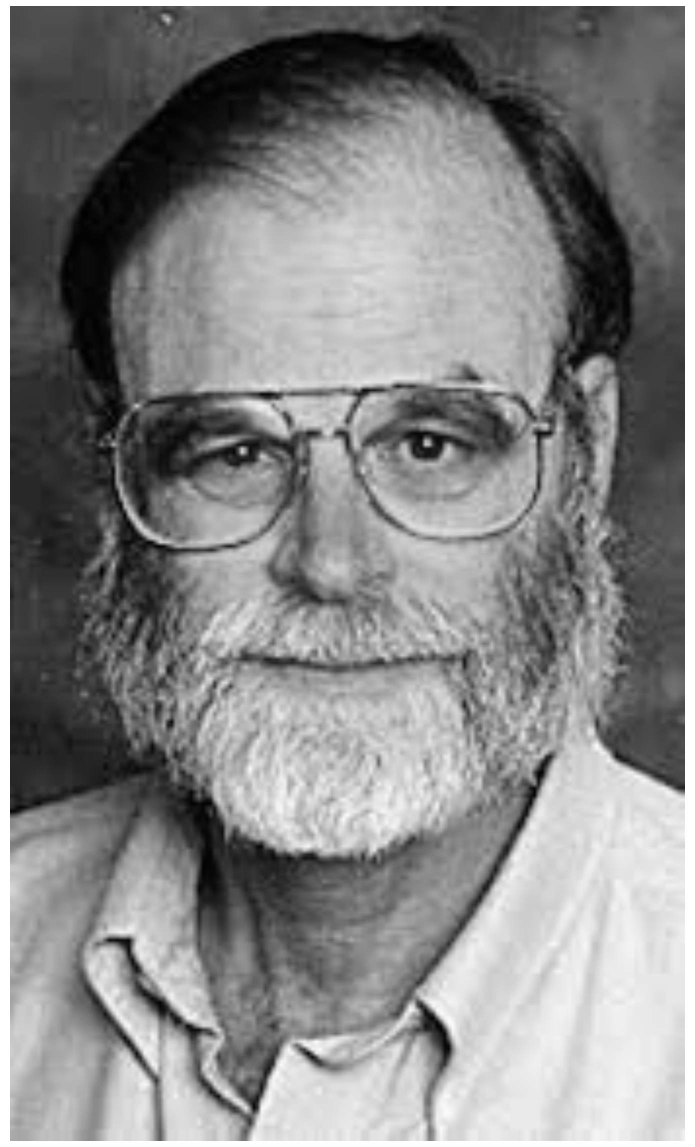
Data systems rely on algorithms



[S. Idreos, 2019]

Data structures define performance





register = this room
caches = this city
memory = nearby city
disk = Pluto

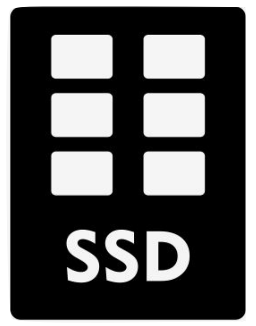
Jim Gray, Turing Award 1998

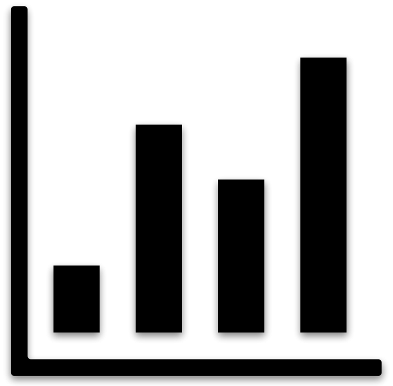
[S. Idreos, 2019]

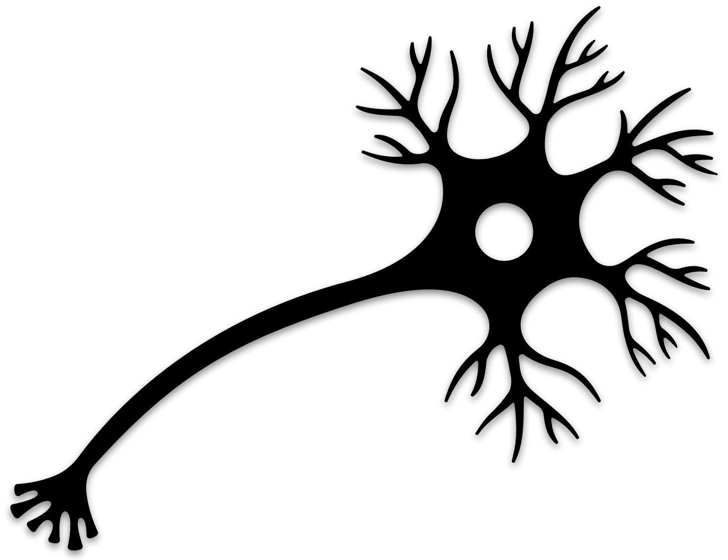
Database Questions

How do I make my **data system** run x times as fast?  (sql,nosql,bigdata, …)

 How do I minimize my **bill** in the **cloud**?

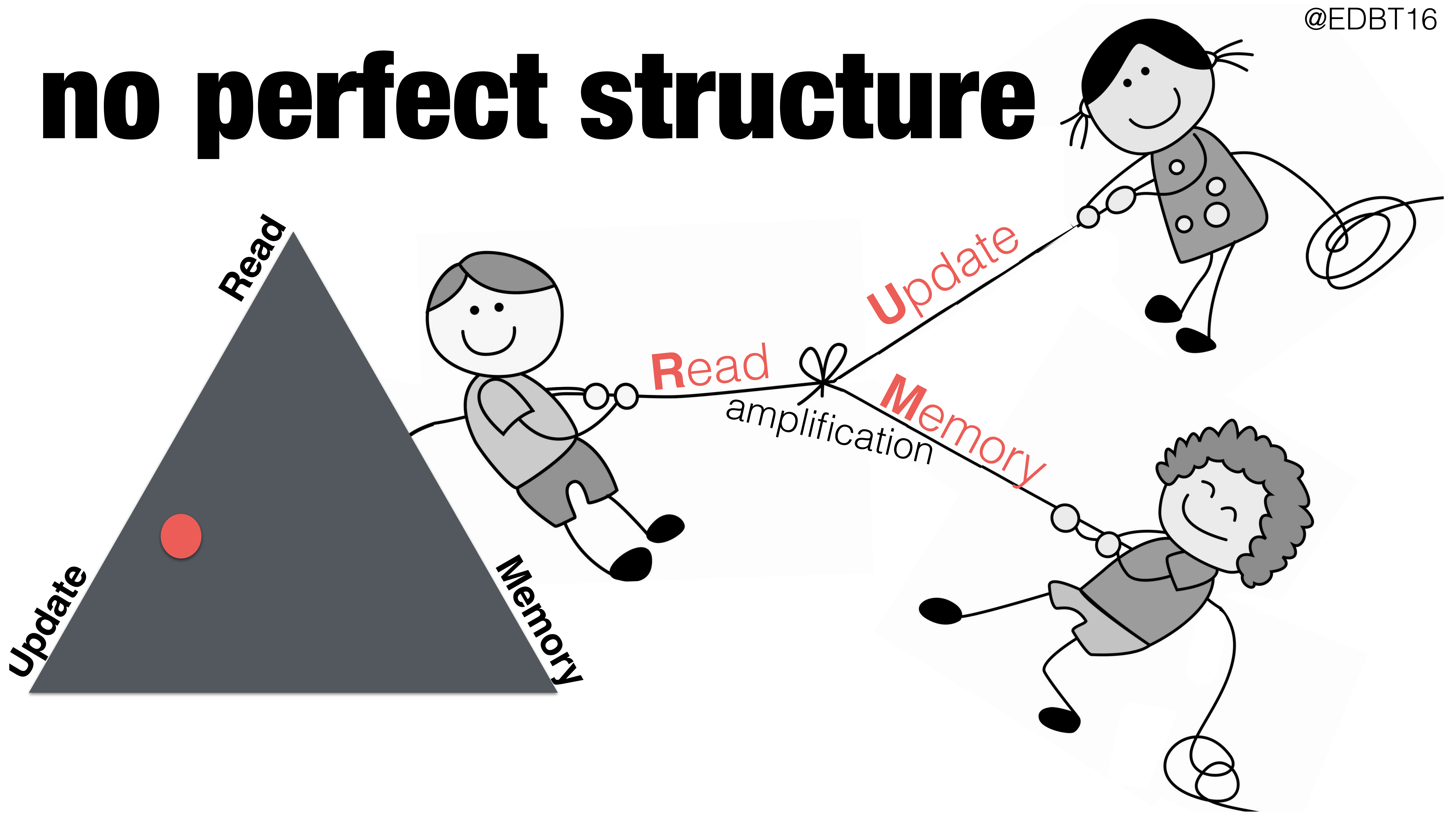
How do I extend the **lifetime** of my hardware? 

How to accelerate **statistics** computation for data science/ML? 

 How do I train my **neural network** x times faster?

[S. Idreos, 2019]

Tradeoffs in each structure

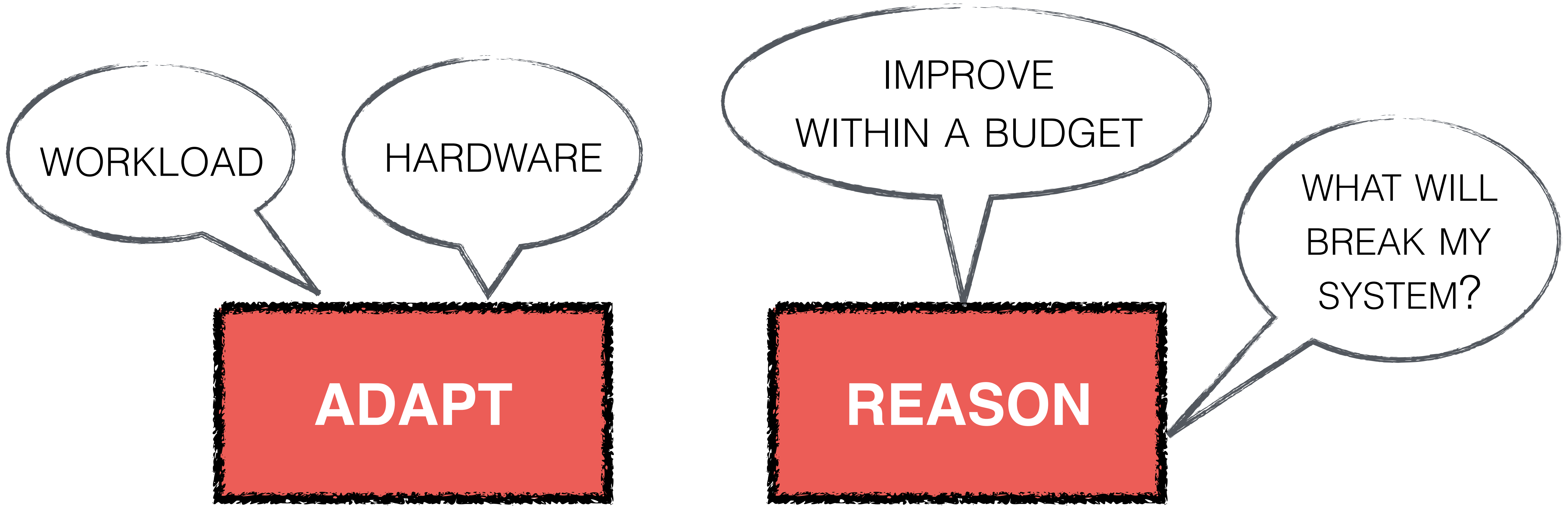


[S. Idreos, 2019]

New Applications Demand Change

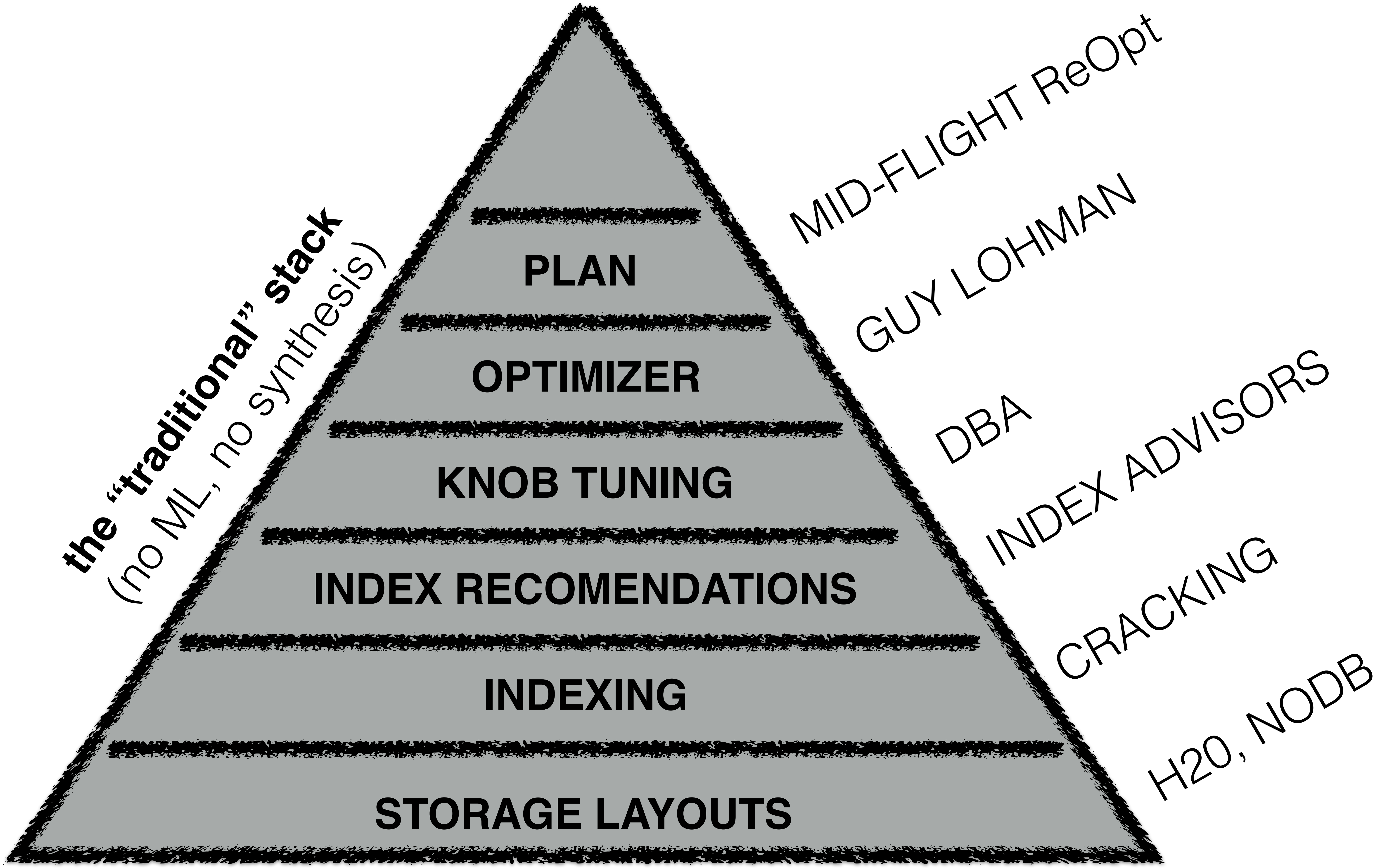
NEW APPLICATIONS

existing systems need to change too



[S. Idreos, 2019]

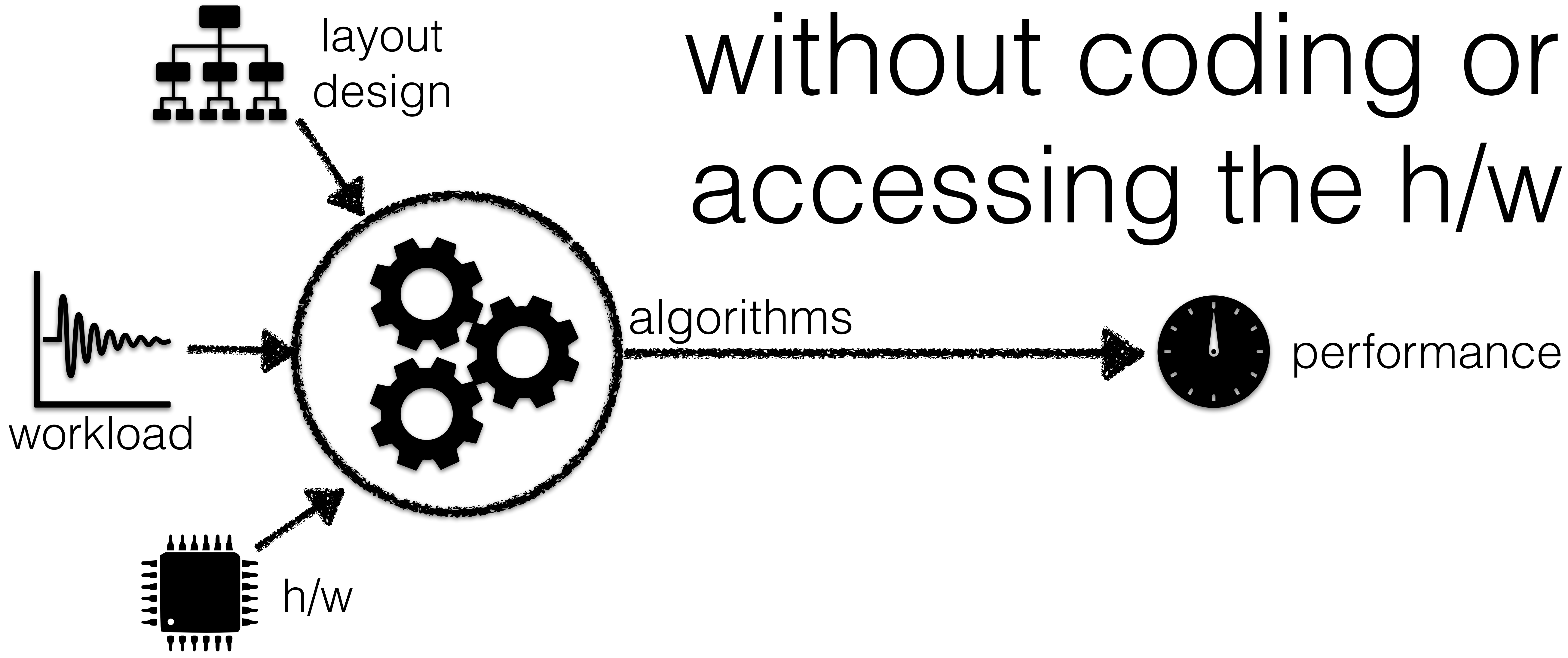
"Traditional" Database Research



[S. Idreos, 2019]

Self-designing systems

Data Calculator



without coding or accessing the h/w

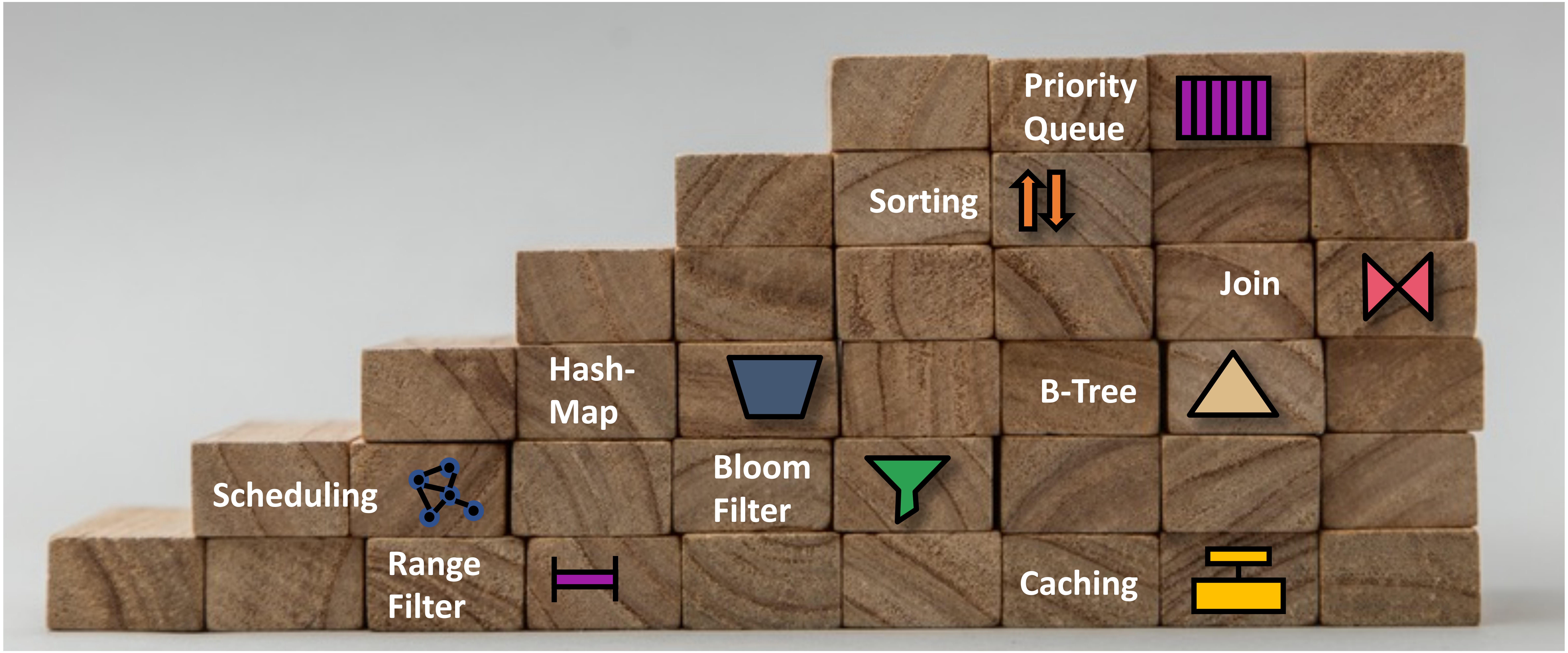
DASlab @ Harvard SEAS

[S. Idreos, 2019]

SageDB: a learned database system

T. Kraska, M. Alizadeh, A. Beutel, E. H. Chi, J. Ding, A. Kristo, G. Leclerc, S. Madden, H. Mao, and V. Nathan

Learned Data Structures and Algorithms



Discussion

- Is this the future?
- What about comparison baselines?
- Lots of work being done in this area

Reminders

- Assignment 5 Due Thursday
- Final Exam Review Wednesday (come with questions!)
- Final Exam on Monday, April 26 from 4-5:50pm (Online)