Microsoft

How to kill two birds with one stone

Leveraging AI investments to accelerate database systems

Matteo Interlandi

Microsoft Azure Data
Gray Systems Lab

The minds behind this talk:

TQP

Query Processing on Tensor Computation Runtimes

Dong He[1], Supun Nakandala[2], Dalitso Banda[3], Rathijit Sen[3], Karla Saur[3], Kwanghyun Park[3], Carlo Curino[3], Jesús Camacho-Rodríguez[3], Konstantinos Karanasos[3], Matteo Interlandi[3]
[1]University of Washington, [2]University of California, San Diego, [3]Microsoft
[1]donghe@cs.washington.edu, [2]snakanda@eng.ucsd.edu, [3]<firstname>.<lastname>@microsoft.com

Tensors: An abstraction for general data processing

Dimitrios Koutsoukos[1], Supun Nakandala[2], Konstantinos Karanasos[3], Karla Saur[3], Gustavo Alonso[1], Matteo Interlandi[3]

Share the Tensor Tea: How Databases can Leverage the Machine Learning Ecosystem

Yuki Asada*[2], Victor Fu*[2], Apurva Gandhi*[1], Advitya Gemawat*[1], Lihao Zhang*[2], Dong He[4], Vivek Gupta[3], Ehi Nosakhare[1], Dalitso Banda[1], Rathijit Sen[1], Matteo Interlandi[1]
[1,2,3]Microsoft, [4]University of Washington

HB

A Tensor Compiler for Unified Machine Learning Prediction Serving

Supun Nakandala[c,], Karla Saur[m], Gyeong-In Yu[s,*], Konstantinos Karanasos[m], Carlo Curino[m], Markus Weimer[m], Matteo Interlandi[m]*
[m]Microsoft, [c]UC San Diego, [s]Seoul National University
{<name>.<surname>}@microsoft.com, snakanda@eng.ucsd.edu, gyeongin@snu.ac.kr

WindTunnel: Towards Differentiable ML Pipelines Beyond a Single Model

Gyeong-In Yu*
Seoul National University
gyeongin@snu.ac.kr

Saeed Amizadeh
Microsoft
saamizad@microsoft.com

Sehoon Kim
UC Berkeley
sehoonkim@berkeley.edu

Artidoro Pagnoni*
Carnegie Mellon University
apagnoni@andrew.cmu.edu

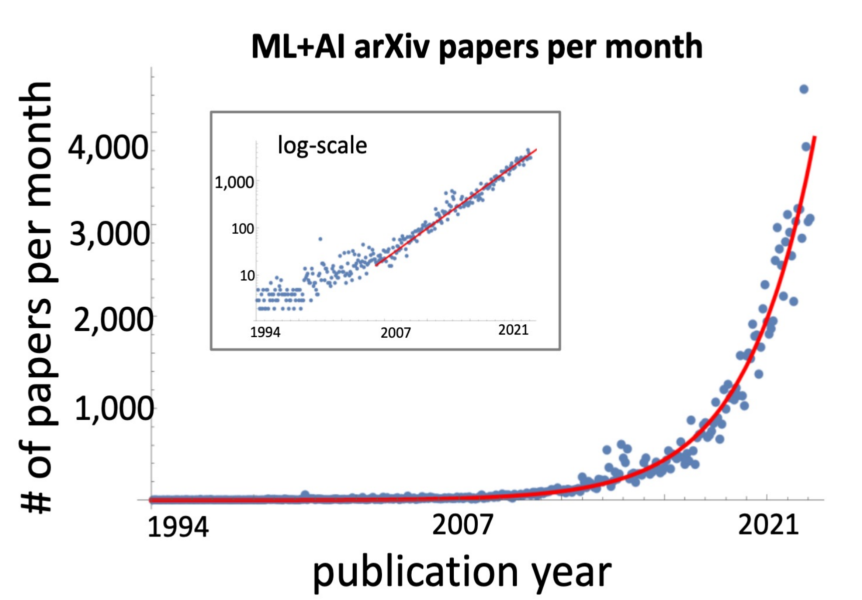
Ce Zhang
ETH Zurich
ce.zhang@inf.ethz.ch

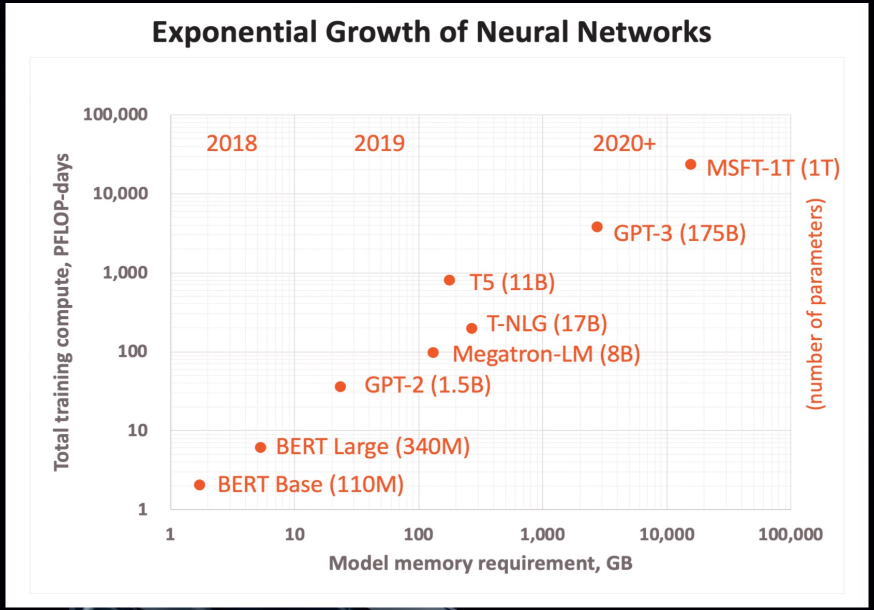
Byung-Gon Chun
Seoul National University
FriendliAI
bgchun@snu.ac.kr

Markus Weimer
Microsoft
mweimer@microsoft.com

Matteo Interlandi
Microsoft
mainterl@microsoft.com

AI is growing (especially NN)





Big $$$ spent on Special HW for NN

VCs are pouring $2B/quarter

Market expected to exceed $200B/year by 2025.

https://www.statista.com/statistics/1003890/worldwide-artificial-intelligence-hardware-market-revenues/

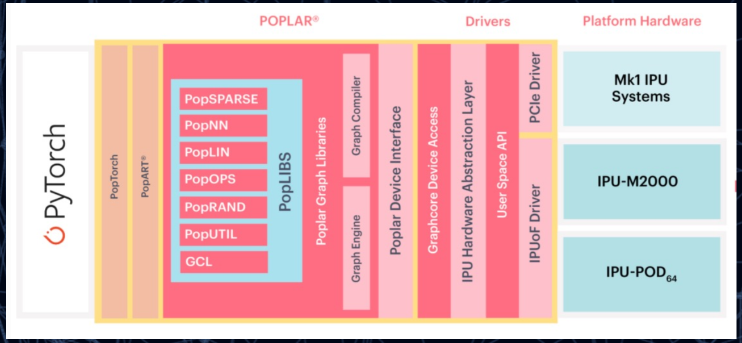
Tensor Runtime

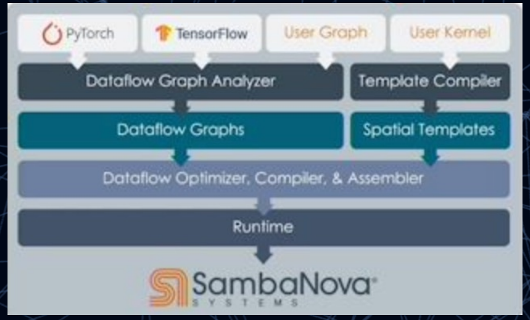
Very large/active communities

Tensor as de-facto API

PyTorch for AMD ROCm™ Platform now available

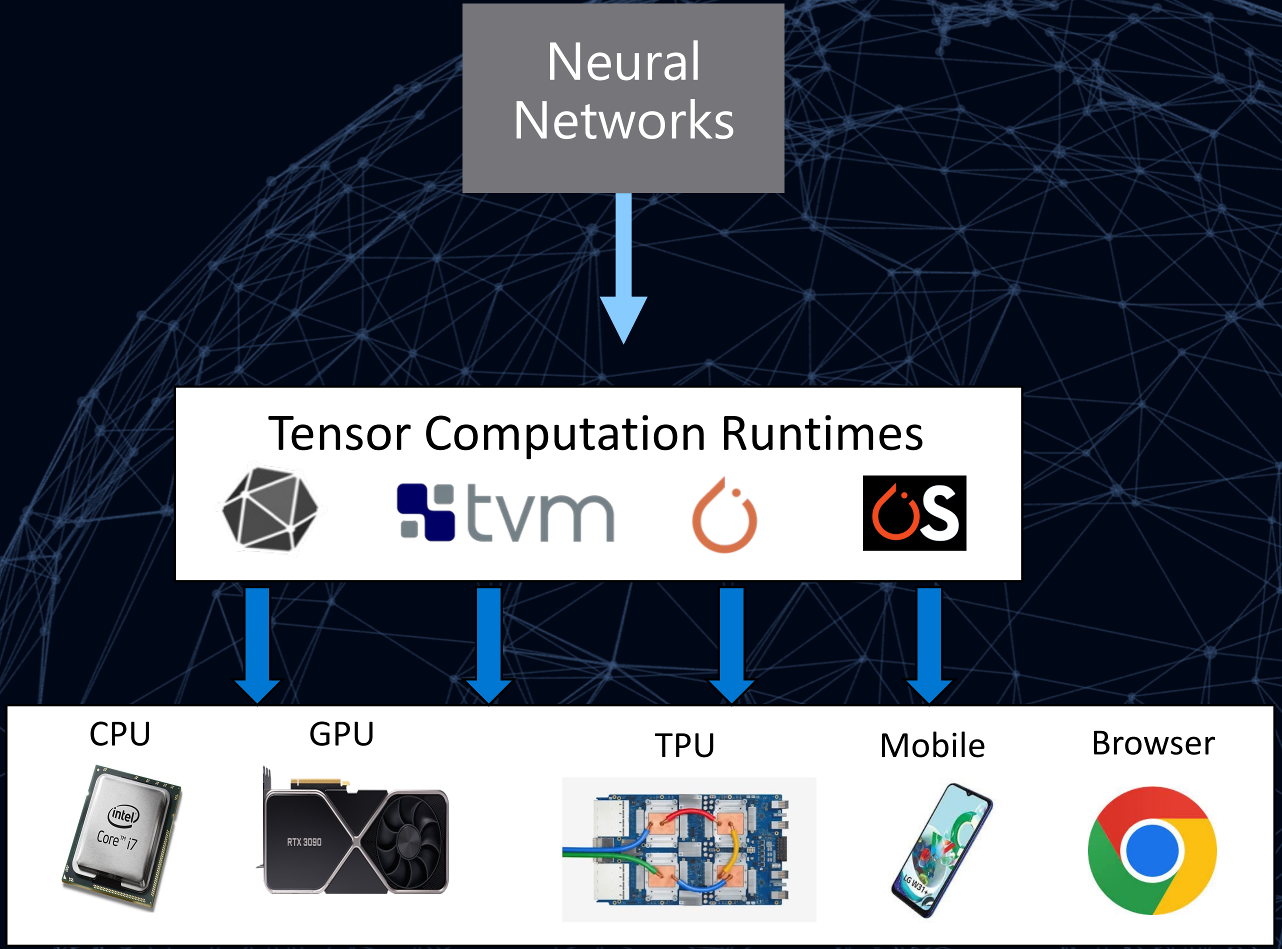
Introducing Accelerated PyTorch Training on Mac





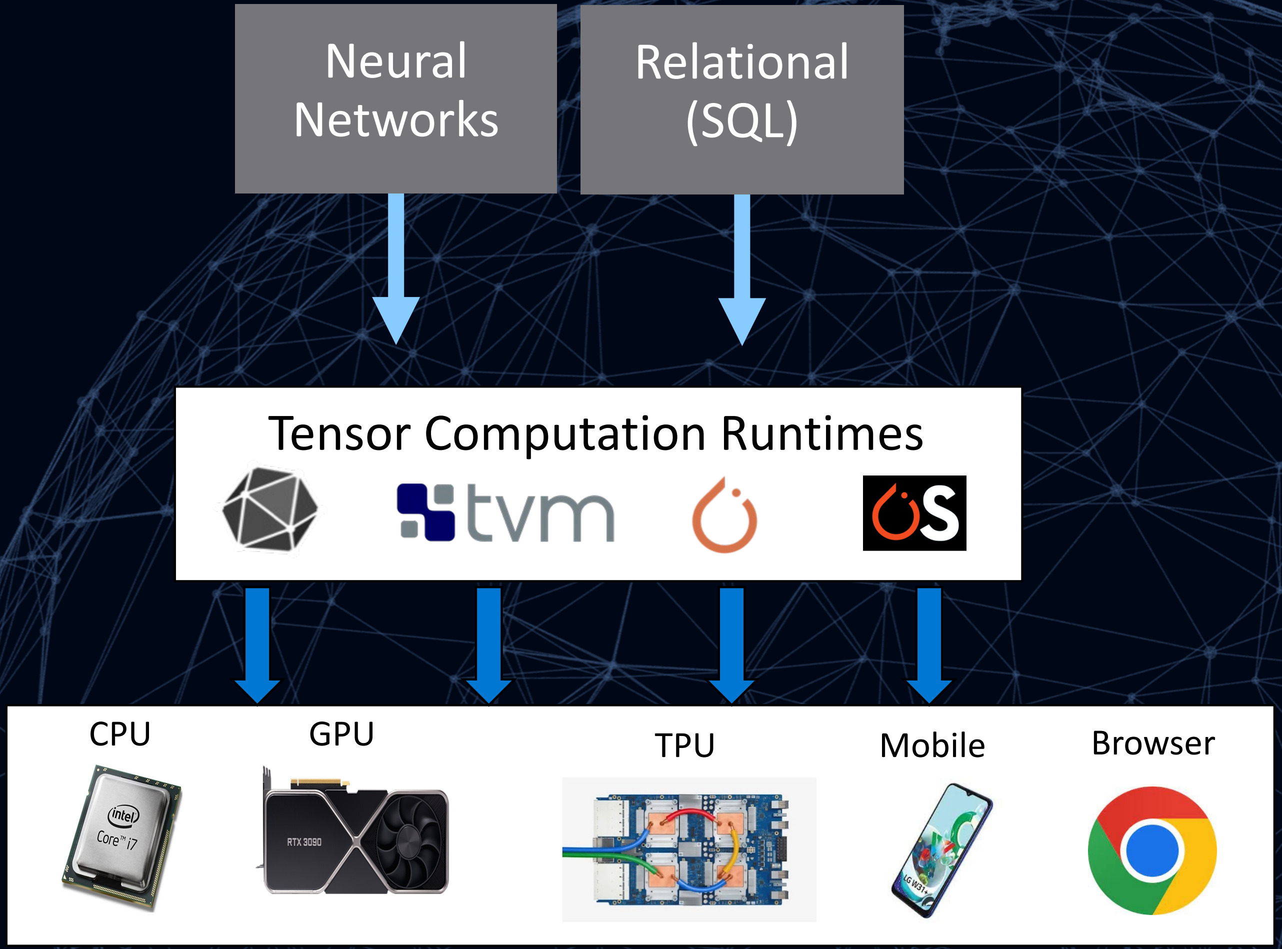
How can Databases take advantage of this?

Core Idea: Tensor Query Processing

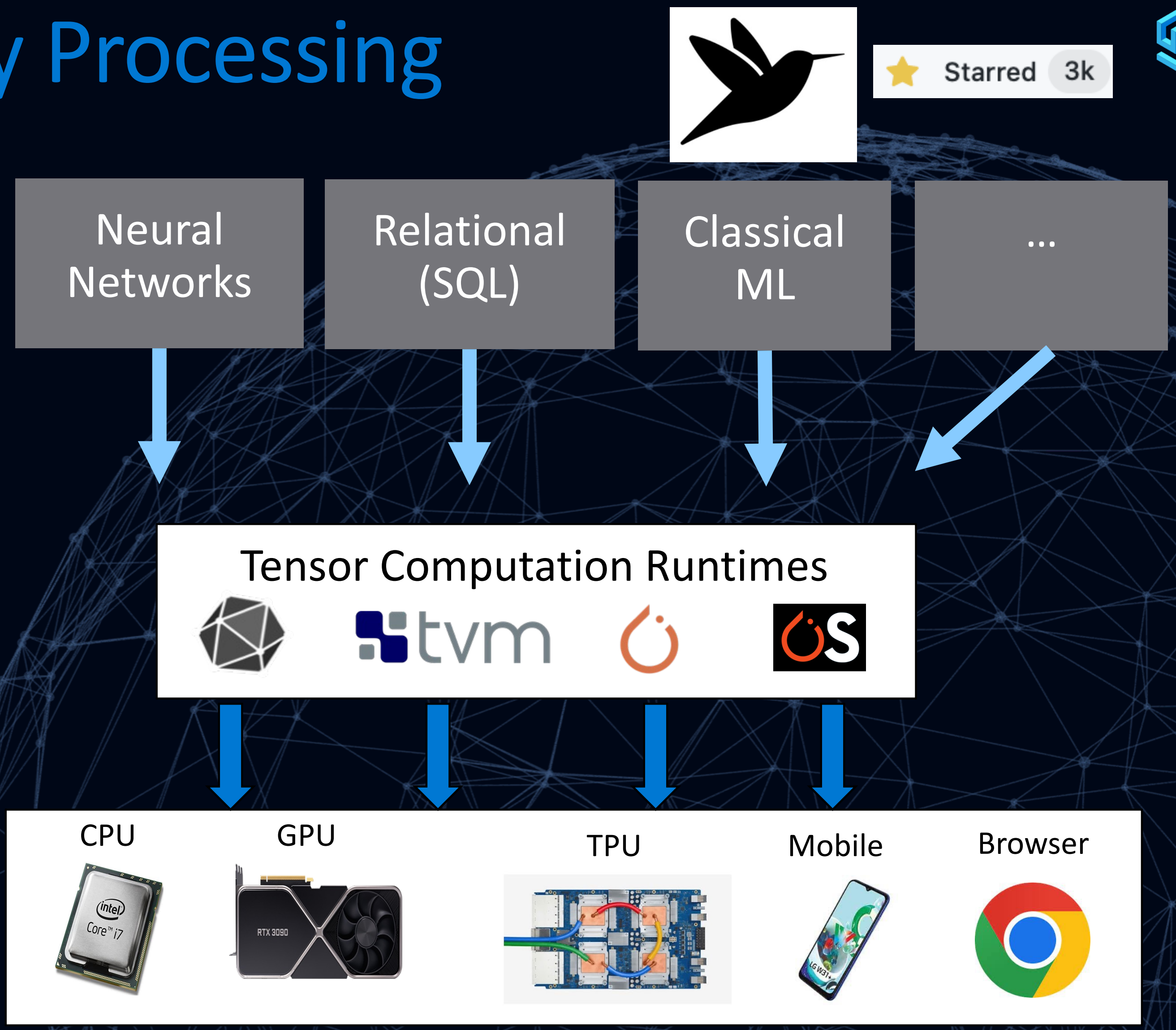


Core Idea: Tensor Query Processing

Compile SQL to the popular tensor abstraction!



Core Idea: Tensor Query Processing



Compile SQL to the popular tensor abstraction!

Pros and Cons of “Tensor Query Processing”

Pros

- ✓ Leverage the massive investments in special HW
- ✓ Scalable Approach (tensor runtimes are getting ported to each new HW)

Cons

- ? Is this even possible?
- ? What about performance?
- ? How expensive is it going to be? (engineering wise)

System Design

SQL Query

```sql
SELECT
  MAX(p_supplycost)
  AS price,
  s_name AS supp
FROM supplier
JOIN partsupp
ON
  ps_suppkey=s_suppkey
GROUP BY
  supplier.s_name
ORDER BY
  price DESC;
```

Parsing Layer

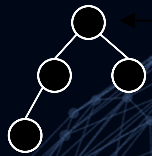
System Design

SQL Query

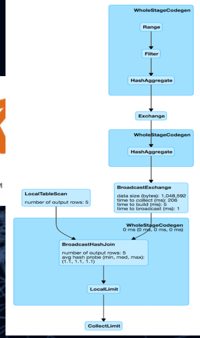
```
SELECT
  MAX(p_supplycost)
  AS price,
  s_name AS supp
FROM supplier
JOIN partsupp
ON
  ps_suppkey=s_suppkey
GROUP BY
  supplier.s_name
ORDER BY
  price DESC;
```

Parsing Layer

IR Graph

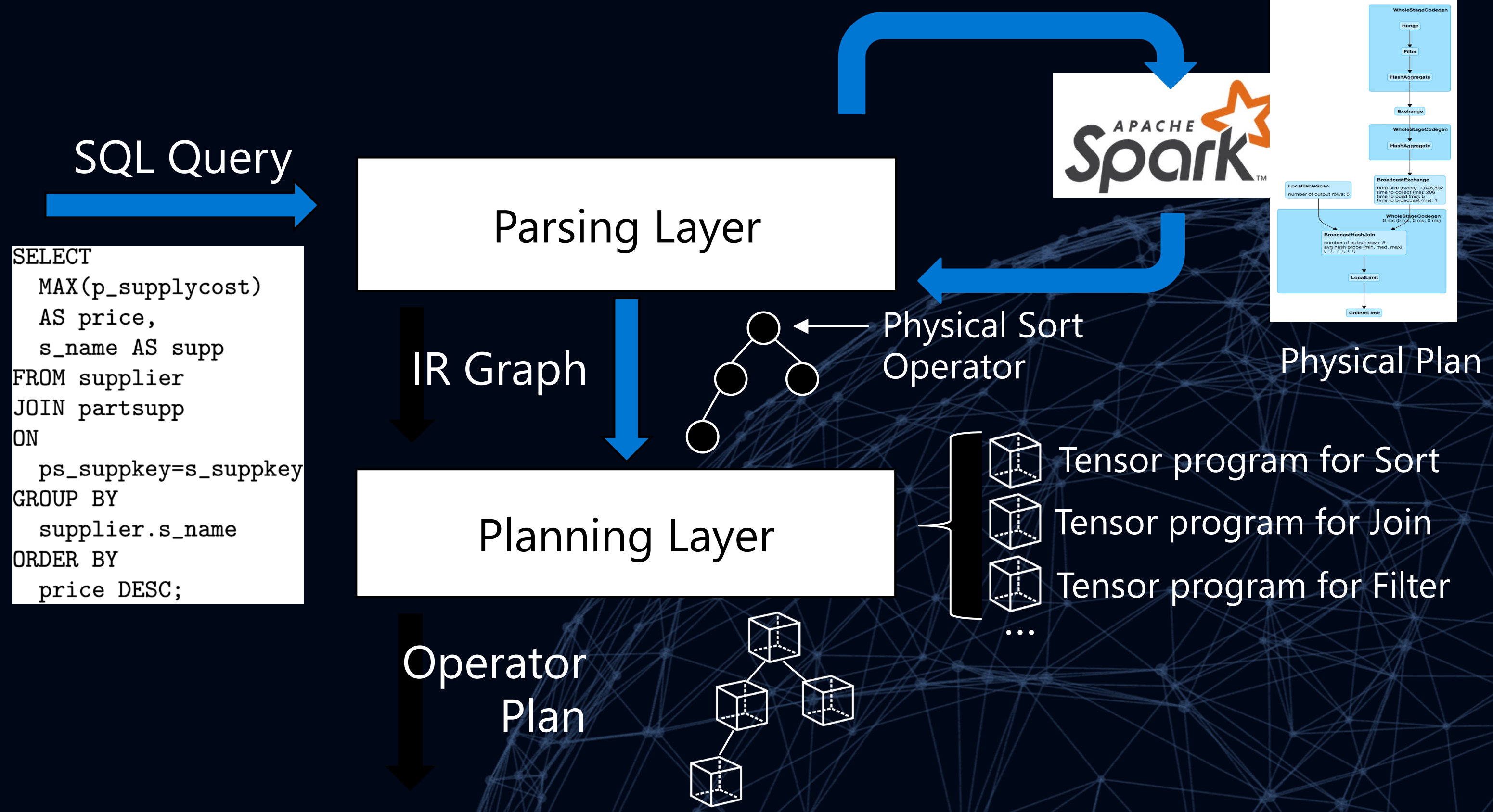


Physical Sort Operator

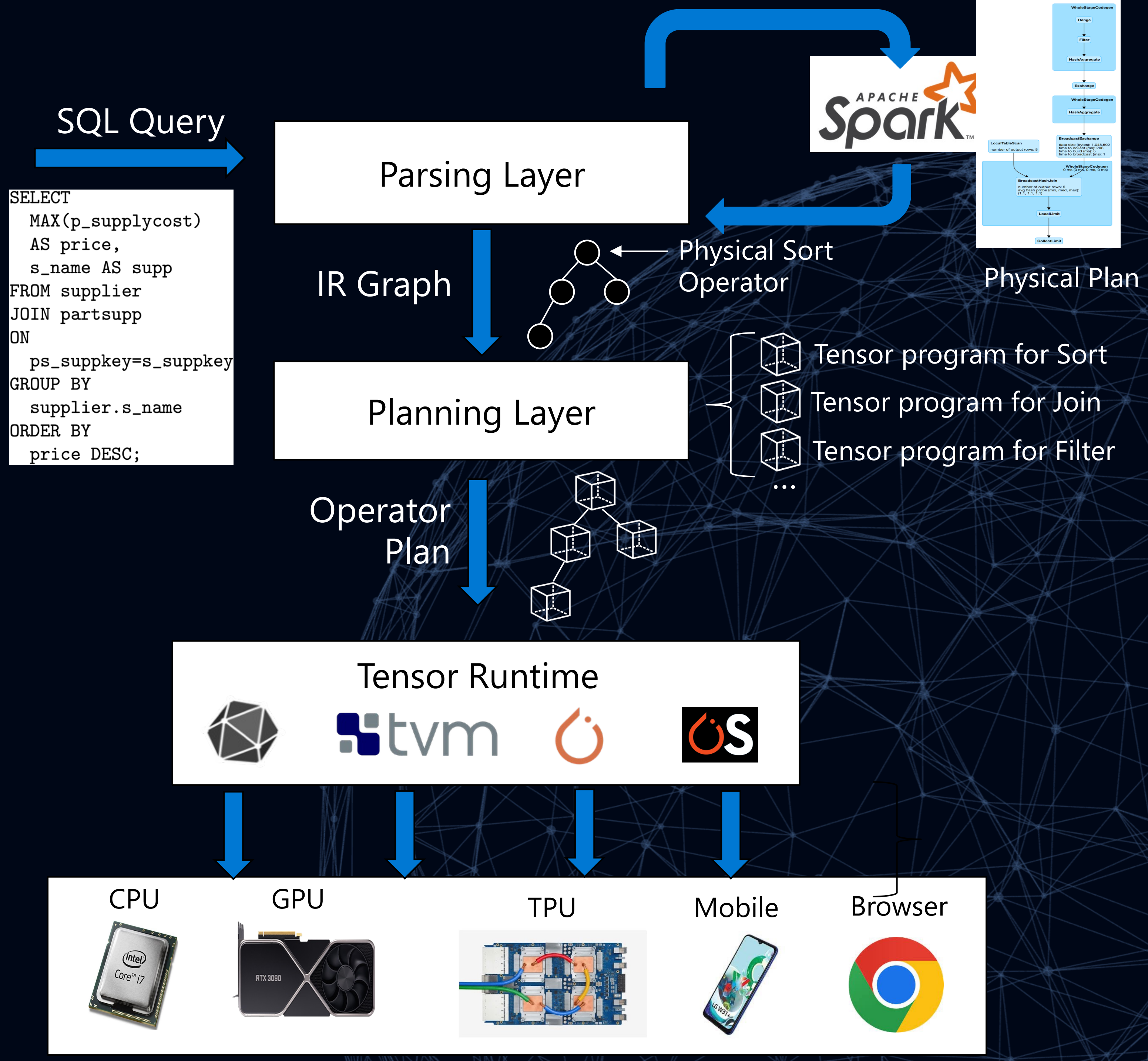


Physical Plan

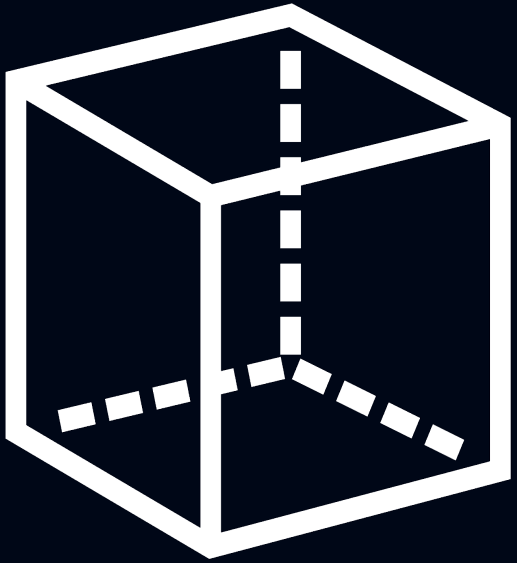
System Design



System Design



Example: Tensor Program for Filter



```
WHERE L_QUANTITY < 24
```

Opt 1:

```
mask = torch.lt(l_quantity, 24)
output = torch.masked_select(l_quantity, mask)
```

Opt 2:

```
mask = torch.lt(l_quantity, 24)
idx = torch.nonzero(mask)
output = torch.index_select(l_quantity, dim=0, mask)
```

Numeric as N x 1 tensors

Dates as N x 1 numeric

Strings as UTF-8 N x max_length

Pros and Cons of "Tensor Query Processing"

Pros

- ✓ Leverage the massive investments in special HW
- ✓ Scalable Approach (tensor runtimes are getting ported to each new HW)

Cons

- ? Is this even possible?
- ? What about performance? (as compared with state-of-the-art)
- ? How expensive is it going to be? (engineering wise)

Pros and Cons of “Tensor Query Processing”

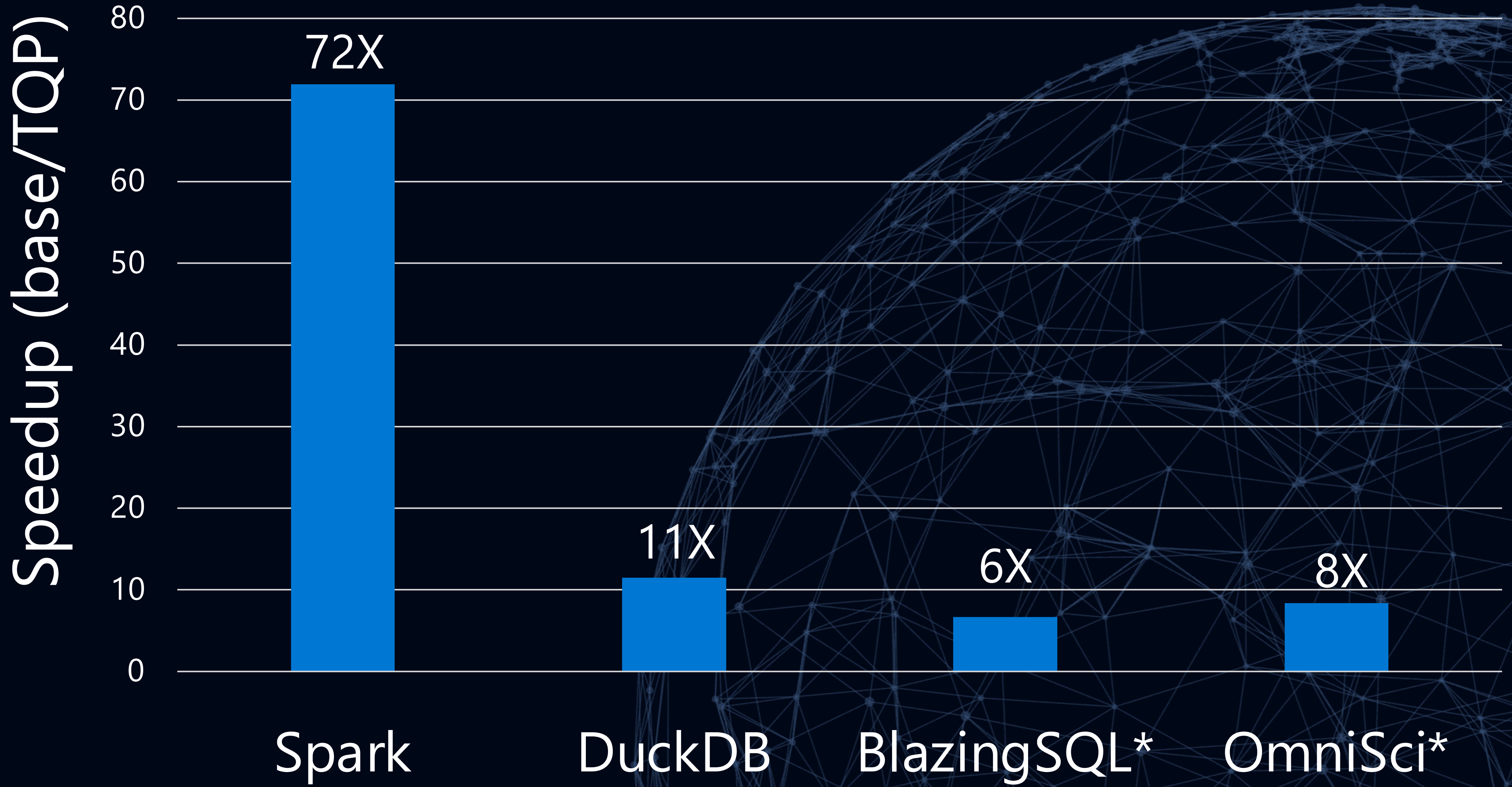
Pros

- ✓ Leverage the massive investments in special HW
- ✓ Scalable Approach (tensor runtimes are getting ported to each new HW)
- ✓ Is this even possible? → YES we can easily cover TPC-H

Cons

- ? What about performance?
- ? How expensive is it going to be? (engineering wise)

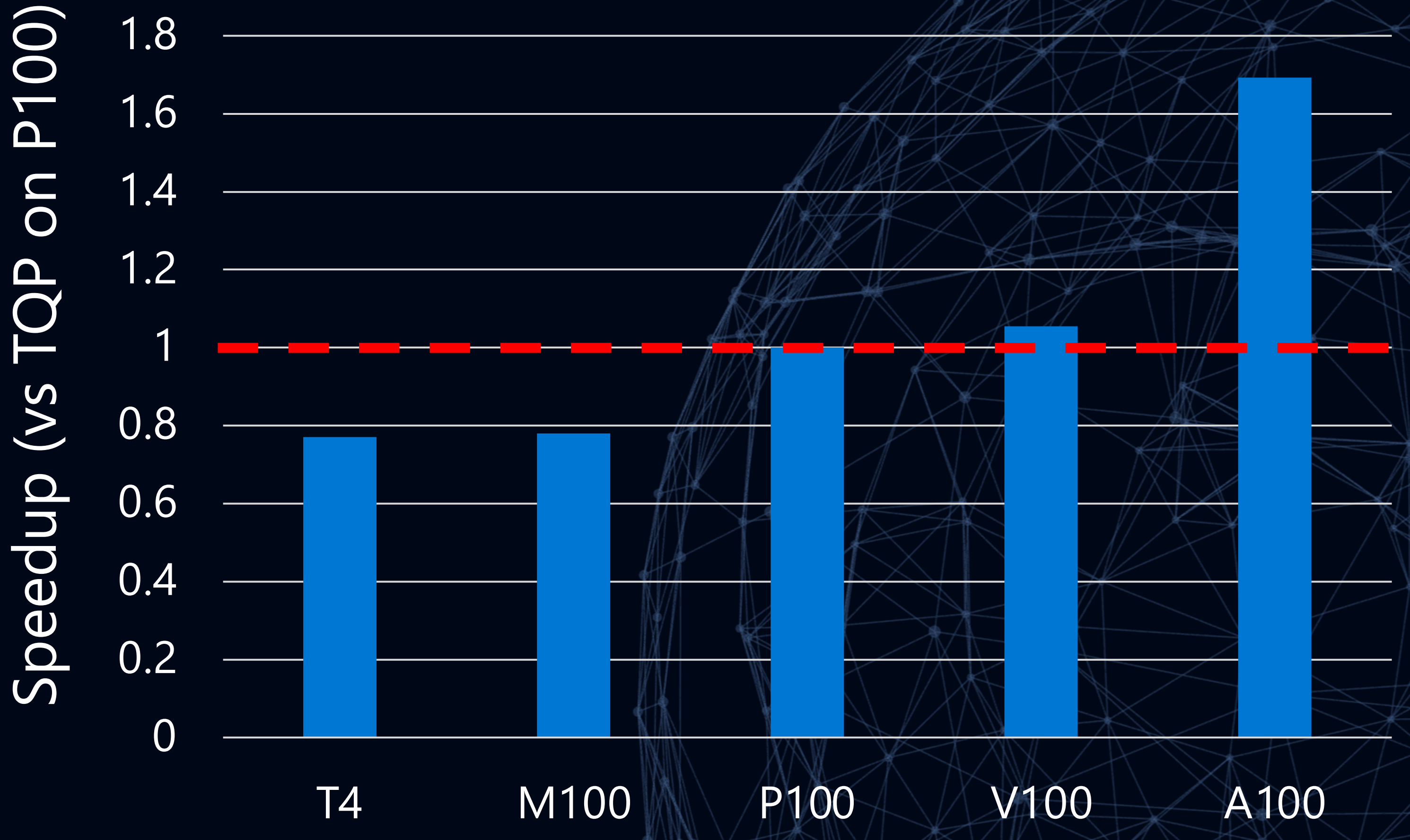
TQP Performance vs State of the Art



TPC-H Scale Factor 10GB, running on: 6 vCores, 112 GBRAM, P100 GPU

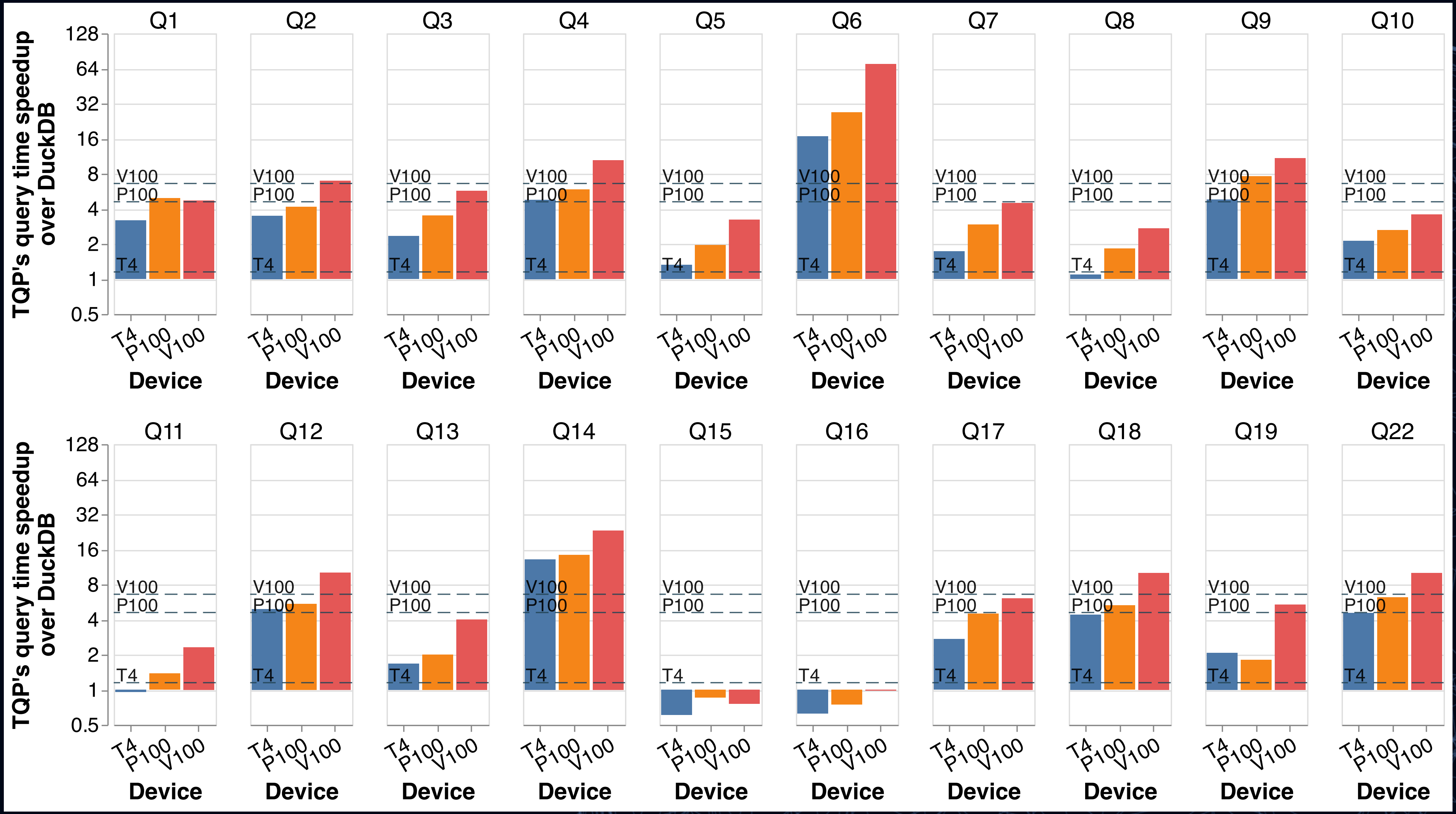
TQP Portability (different HW)

TPU Mobile Browser



TPC-H Scale Factor 1GB, running on: 6 vCores, 112 GBRAM (P100 relative speedup)

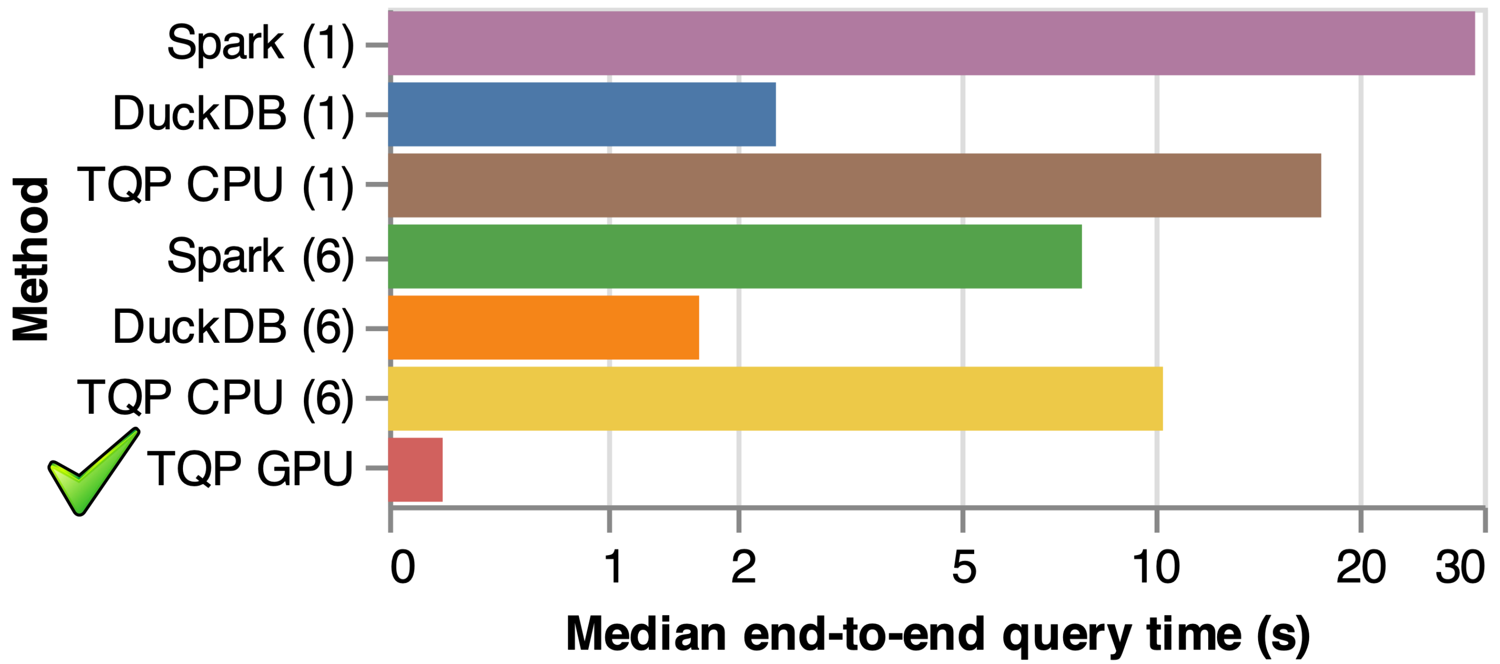
TQP Price/Perf Comparison (vs DuckDB)



TPC-H Scale Factor 10GB, running on: 8 vCores, 32 GBRAM (speed up vs DuckDB)

Performance on queries mixing ML and SQL

TQP supports mixing SQL and predictions of models built using PyTorch, scikit-learn and HuggingFace libraries.



Query time (in seconds) on a query mixing tree ensemble, one-hot encoding, feature scaling & concatenation with relational join, aggregation and filtering. In parentheses are the numbers of cores for CPU-based systems.

Pros and Cons of "Tensor Query Processing"

Pros

- ✓ Leverage the massive investments in special HW
- ✓ Scalable Approach (tensor runtimes are getting ported to each new HW)
- ✓ Is this even possible?

Cons

- ? What about performance?
- ? How expensive is it going to be? (engineering wise)

Pros and Cons of "Tensor Query Processing"

Pros

- ✓ Leverage the massive investments in special HW
- ✓ Scalable Approach (tensor runtimes are getting ported to each new HW)
- ✓ Is this even possible?
- ✓ What about performance?

Cons

- ? How expensive is it going to be? (engineering wise)

Pros and Cons of “Tensor Query Processing”

Pros

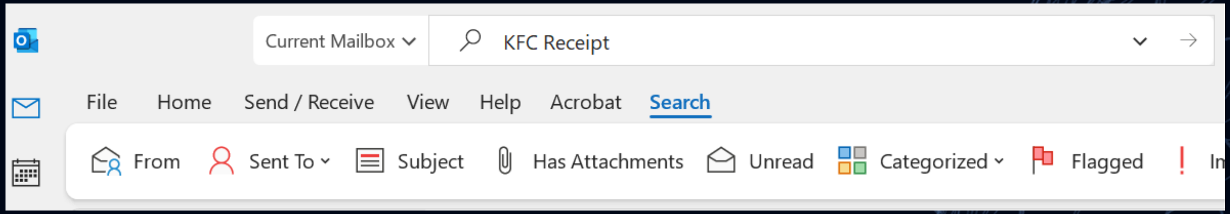
- ✓ Leverage the massive investments in special HW
- ✓ Scalable Approach (tensor runtimes are getting ported to each new HW)
- ✓ Is this even possible?
- ✓ What about performance?
- ✓ How expensive is it going to be? (engineering wise)

Less than 10k LoC for TQP+ HB

Future Directions: AI-Centric Database System

Broader implications of having a DBMS co-existing with an ML runtime

1. Multi-modal data support

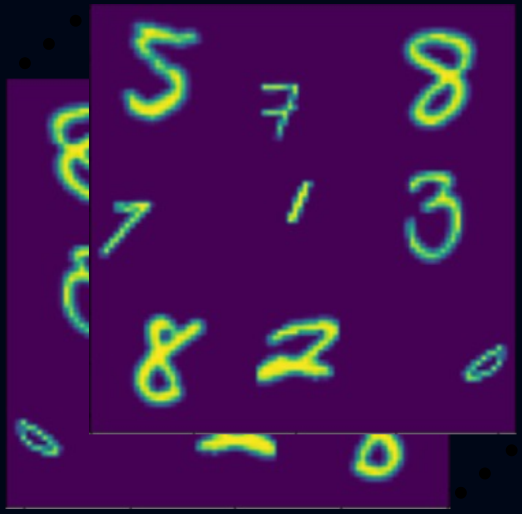



```sql
SELECT
    input AS images,
    image_text_similarity_model("KFC Receipt", input) AS score
FROM attachments
ORDER BY score DESC
LIMIT 1
```

Future Directions: AI-Centric Database System

Broader implications of having a DBMS co-existing with an ML runtime

1. Multi-modal data support
2. Automatic Differentiation



Example Query Inputs

```sql
SELECT Digit, Size, COUNT(*)
FROM parseImageToTable(Image)
GROUP BY Digit, Size
```

Trainable Table Value Function

Example Query Outputs

Digit	Size	Count
0	Small	1
	Large	0
1	Small	1
	Large	0
2	Small	0
	Large	1
3	Small	0
	Large	1
4	Small	0
	Large	0
5	Small	0
	Large	1
6	Small	0
	Large	0
7	Small	2
	Large	0
8	Small	0
	Large	2
9	Small	0
	Large	0

Digi		
0		
1		
2		
3		
4		
5		
6		
7		
8		
9	Small	0
	Large	2

Conclusion

- Compile SQL to Tensor Programs
- ✓ Free-ride on $B of dollars of HW/SW investments
- ✓ Performance looks great
- ✓ Low engineering costs
- ✓ Fun future directions

Microsoft Azure Data
Gray Systems Lab

Thanks!

https://aka.ms/gsl

Video credits: http://storyblocks.com