CWI Centrum Wiskunde & Informatica

Data Systems Research at CWI

Peter Boncz
CWI + VU

+ *all members of the Database Architectures group..*

CWI
Centrum Wiskunde & Informatica

Outline

- **Intro**
  - Data systems research @ CWI DA
- **Past**
  - MonetDB & VectorWise + impact
- **Present**
  - DuckDB ("in-process analytics")
- **Future**
  - Learned Data Formats
  - SQL:2023 (Property Graph Queries)
  - Responsible Decentralized Data Architectures

Data Systems Research

- data systems: of the most **complex software**
  - 10-100Ks of source code, takes 5-10y and 100sFTE
  - compression, data structures, algorithms, optimization, machine learning, compilers, operating systems, hardware co-design

Data Systems Research

- data systems: of the most complex software
  - 10-100Ks of source code, takes 5-10y and 100sFTE
  - compression, data structures, algorithms, optimization, machine learning, compilers, operating systems, hardware co-design
- architecture = reconciling **conflicting interests**
  - writes⇔reads; size⇔computation; flexibility⇔efficiency; etc

Data Systems Research

- data systems: of the most complex software
  - 10-100Ks of source code, takes 5-10y and 100sFTE
  - compression, data structures, algorithms, optimization, machine learning, compilers, operating systems, hardware co-design
- architecture = reconciling conflicting interests
  - writes⇔reads; datasize⇔computation; flexibility⇔efficiency; etc
- **systems research**
  - PVLDB, SIGMOD, CIDR, ICDE, DaMoN
  - Turing Awards: Jim Gray, Michael Stonebraker

CWI
Centrum Wiskunde & Informatica

Data Systems Research at CWI

- data systems: of the most complex software
  - 10-100Ks of source code, takes 5-10y and 100sFTE
  - compression, data structures, algorithms, optimization, machine learning, compilers, operating systems, hardware co-design
- architecture = reconciling conflicting interests
  - writes⇔reads; datasize⇔computation; flexibility⇔efficiency; etc
- systems research
  - PVLDB, SIGMOD, CIDR, ICDE, DaMoN
  - Turing Awards: Jim Gray, Michael Stonebraker

Martin
Kersten

Data Systems Research at CWI

- Database Architectures (now)
    - tenured: Peter Boncz, Stefan Manegold, Hannes Mühleisen
    - post-doc: Mark Raasveldt, Pedro Holanda, Gabor Szarnyas
    - PhD: ~~Tim Gubner~~, Diego Tomé, Azim Afroozeh, Laurens Kuiper, Orson Peters, Ilaria Battiston + 7 interns + 2 startup employees

Outline

- **Intro**
  - Data systems research @ CWI DA
- **Past**
  - MonetDB & VectorWise + impact
- **Present**
  - DuckDB (“in-process analytics”)
- **Future**
  - Learned Data Formats
  - SQL:2023 (Property Graph Queries)
  - Responsible Decentralized Data Architectures

Main Memory Column Stores

- Hardware Trends
    - Databases ⇒ Big Data
    - big memories ⇔ memory wall

Main Memory Column Stores

- Hardware Trends
  - Databases ⇒ Big Data
  - big memories ⇔ memory wall
- Conflict: analytics ⇔ transactions
  - Traditional RDBMS optimized for transactions
  - Research Question: database architecture for *data mining*?

Main Memory Column Stores

- Hardware Trends
  - Databases ⇒ Big Data
  - big memories ⇔ memory wall
- Conflict: analytics ⇔ transactions
  - Traditional RDBMS optimized for transactions
  - Research Question: database architecture for *data mining*?

Data Distilleries
driving e-business personalization

Main Memory Column Stores

- Hardware Trends
  - Databases ⇒ Big Data
  - big memories ⇔ memory wall
- Conflict: analytics ⇔ transactions
  - Traditional RDBMS optimized for transactions
  - Research Question: database architecture for *data mining*?
- MonetDB (1994-) *Martin Kersten, Niels Nes, Sjoerd Mullender, ..*
  - column-at-a-time processing
  - CPU & *cache-conscious* algorithms

Data Distilleries
driving e-business personalization

monetdb

Main Memory Column Stores

- Hardware Trends
  - Databases ⇒ Big Data
  - big memories ⇔ memory wall
- Conflict: analytics ⇔ transactions
  - Traditional RDBMS optimized for transactions
  - Research Question: database architecture for *data mining*?
- MonetDB (1994-) *Martin Kersten, Niels Nes, Sjoerd Mullender, ..*
  - column-at-a-time processing
  - CPU & *cache-conscious* algorithms

Data Distilleries
driving e-business personalization

monetdb

Impact ⇒ SAP HANA (2010) + column store revolution (2005-)

Vectorized Query Processing

- MonetDB scalability..
  - all operations materialize full table columns ⇒ “memory hungry”
  - what if data > memory?

Vectorized Query Processing

- MonetDB scalability..
  - all operations materialize full table columns ⇒ “memory hungry”
  - what if data > memory?
- Idea: **incremental** column processing
  - manipulate small column slices (“**vectors**”) at the time
  - VectorWise system (2005-) *Marcin Zukowski, Sándor Héman, ..*
  - scaling beyond RAM (“cooperative scans”)

vectorwise

Vectorized Query Processing

- MonetDB scalability..
  - all operations materialize full table columns ⇒ “memory hungry”
  - what if data > memory?
- Idea: **incremental** column processing
  - manipulate small column slices (“vectors”) at the time
  - VectorWise system (2005-) *Marcin Żukowski, Sandor Heman, ..*
  - scaling beyond RAM (“cooperative scans”)

vectorwise

Impact ⇒ Google Dremel (F1, BigQuery), Apache Drill, Apache Impala, Dremio, CockroachDB, Databricks Delta Engine, Microsoft SQLserver, IBM DB2 Blu, MySQL Heatwave (Oracle), ClickHouse, Snowflake, …, …, …,

VectorWise Data Format

Compressed columnar storage

- **vector**-at-a-time **decoding** algorithms
  - decompress small vectors into **CPU cache**
  - **Run-Length, Zero Truncation (FOR), Delta coding, Bit-Packing**
  - remove outliers (“patching”)

VectorWise Data Format

Compressed columnar storage

- vector-at-a-time decoding algorithms
  - decompress small vectors into CPU cache
  - Run-Length, Zero Truncation (FOR), Delta coding, Bit-Packing
  - remove outliers (“patching”)
- **Skippable Row-groups**
  - **= large table chunks** with inside columnar storage (PAX)
  - **Min/Max** for each column in header ⇒ exploit natural orderings

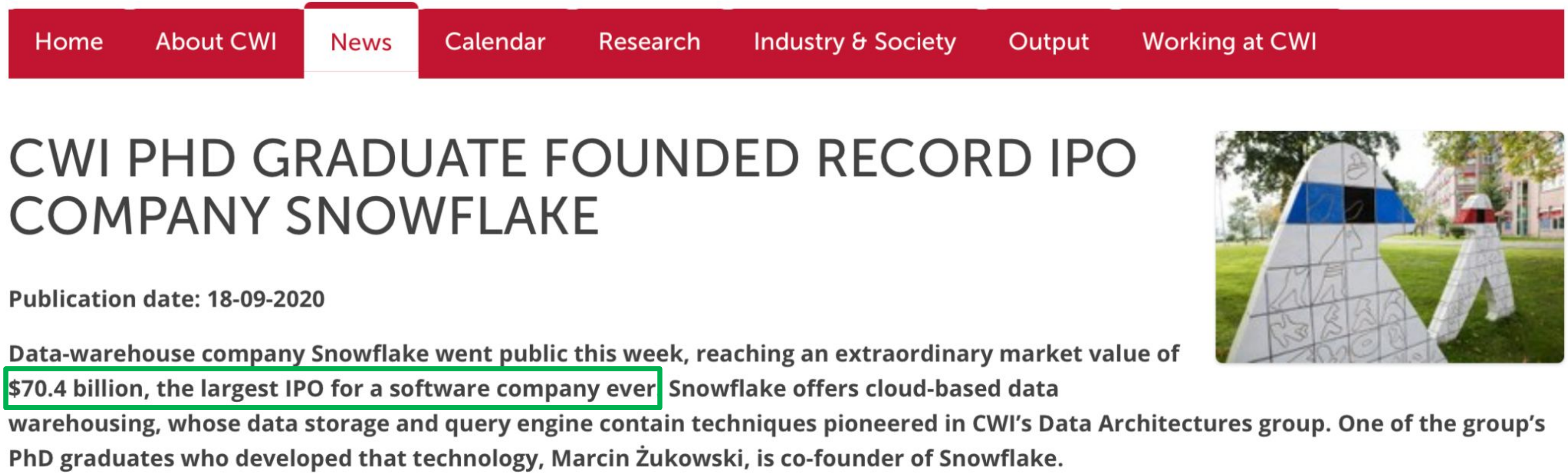
VectorWise Data Format

Compressed columnar storage

- Vectorizable encoding algorithms
    - decompress small vectors into CPU L1 cache
    - Run-Length, Zero Truncation (FOR), Delta coding, Bit-Packing
    - remove outliers (“patching”)
- Skippable Row-groups
    - Columnar storage per large table chunk
    - Min/Max for each column in header ⇒ exploit natural orderings

Impact ⇒ Lucene, Dremel, ORC, **Parquet** (yottabytes)

VectorWise ⇒ snowflake

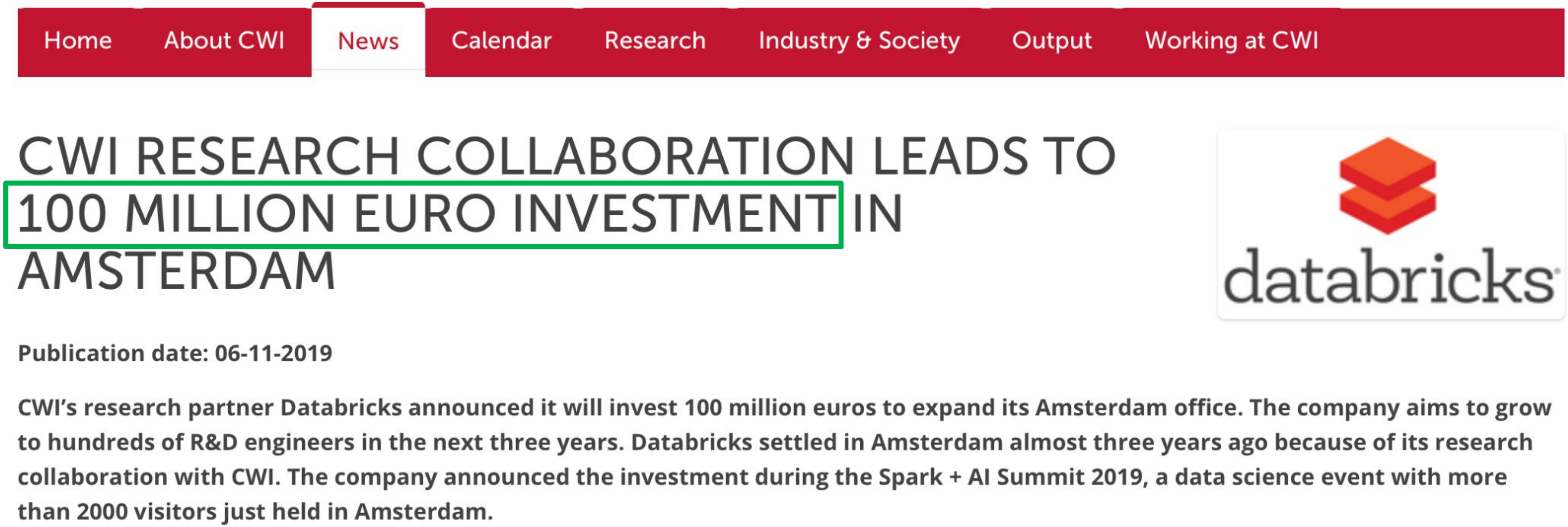


CWI PHD GRADUATE FOUNDED RECORD IPO COMPANY SNOWFLAKE

Publication date: 18-09-2020

Data-warehouse company Snowflake went public this week, reaching an extraordinary market value of $70.4 billion, the largest IPO for a software company ever. Snowflake offers cloud-based data warehousing, whose data storage and query engine contain techniques pioneered in CWI's Data Architectures group. One of the group's PhD graduates who developed that technology, Marcin Żukowski, is co-founder of Snowflake.

VectorWise ⇒ Databricks (Spark)

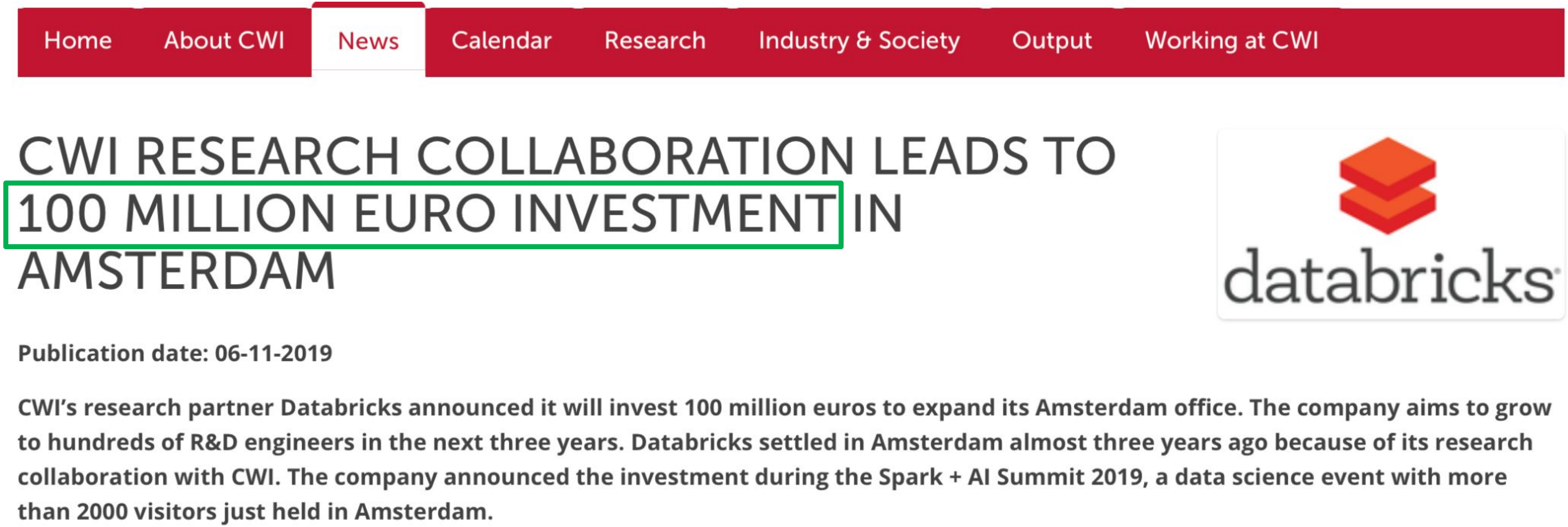


CWI RESEARCH COLLABORATION LEADS TO 100 MILLION EURO INVESTMENT IN AMSTERDAM

Publication date: 06-11-2019

CWI's research partner Databricks announced it will invest 100 million euros to expand its Amsterdam office. The company aims to grow to hundreds of R&D engineers in the next three years. Databricks settled in Amsterdam almost three years ago because of its research collaboration with CWI. The company announced the investment during the Spark + AI Summit 2019, a data science event with more than 2000 visitors just held in Amsterdam.

VectorWise ⇒ Databricks (Spark)



CWI RESEARCH COLLABORATION LEADS TO 100 MILLION EURO INVESTMENT IN AMSTERDAM

Publication date: 06-11-2019

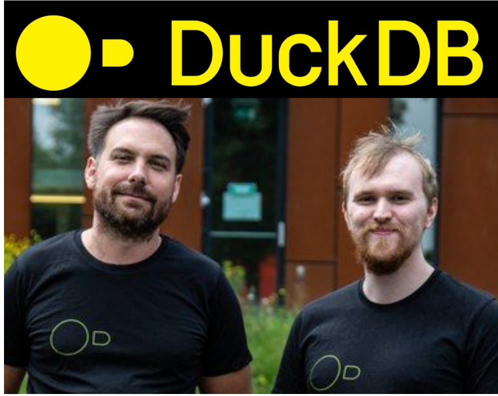
CWI's research partner Databricks announced it will invest 100 million euros to expand its Amsterdam office. The company aims to grow to hundreds of R&D engineers in the next three years. Databricks settled in Amsterdam almost three years ago because of its research collaboration with CWI. The company announced the investment during the Spark + AI Summit 2019, a data science event with more than 2000 visitors just held in Amsterdam.

RelationalAI ORACLE elasticsearch DuckDB

Outline

- **Intro**
  - Data systems research @ CWI DA
- **Past**
  - MonetDB & VectorWise + impact
- **Present**
  - DuckDB (“in-process analytics”)
- **Future**
  - Learned Data Formats
  - SQL:2023 (Property Graph Queries)
  - Responsible Decentralized Data Architectures

CWI
Centrum Wiskunde & Informatica



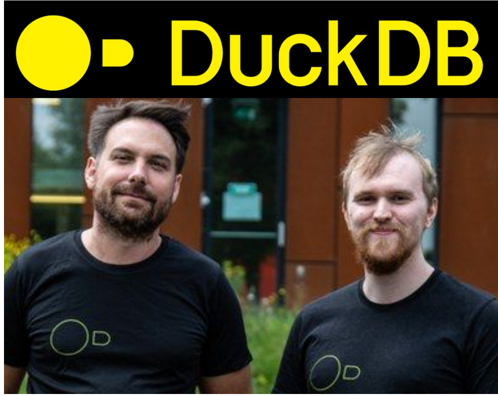
DuckDB: in-process analytics

- Created by Hannes Mühleisen and Mark Raasveldt
- Idea: **analytical SQL system as a linkable library**
- From research on data systems support for data science

https://duckdb.org/ https://shell.duckdb.org

CWI
Centrum Wiskunde & Informatica

DuckDB: in-process analytics



- Created by Hannes Mühleisen and Mark Raasveldt
- Idea: **analytical SQL system as a linkable library**
- From research on **data systems support for data science:**
    - why don’t data scientists use database systems?
    - can transactions handle data science bulk-updates?
    - can we ensure reliability on cheap consumer hardware?
    - how to execute queries on compressed vector data?
    - scalable algorithms that respect changing/limited host resources?
    - …

Pedro
Holanda

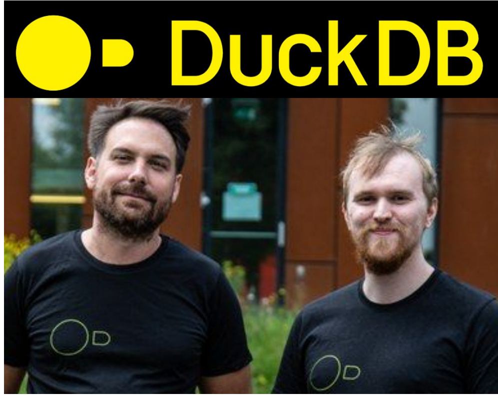
Laurens
Kuiper

+ Ilaria Battiston

https://duckdb.org/ https://shell.duckdb.org

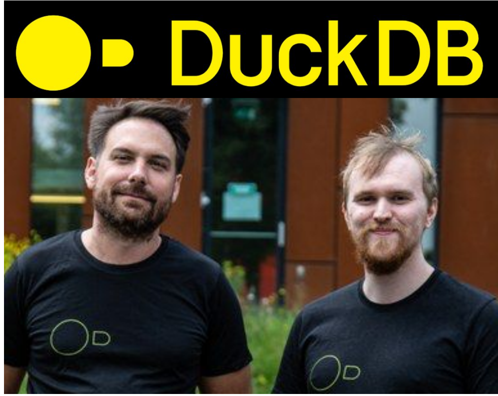
CWI
Centrum Wiskunde & Informatica

DuckDB: in-process analytics



- Created by Hannes Mühleisen and Mark Raasveldt
- Idea: **analytical SQL system as a linkable library**
- We foresee many applications beyond data science notebooks:
  - cloud analytics based on DuckDB
  - mobile apps that keep (most) data decentralized
  - edge analytics (e.g., intelligent streetlights, self-driving cars)
  - client-side (in-browser) query processing
  - computational storage (SQL in the disk drive)
  - secure enclave DBs
  - **your research?**

https://duckdb.org/ https://shell.duckdb.org



DuckDB taking flight

Hannes & Mark put very serious effort in this

- Target non-academics also, e.g., data scientists
  - R, python user groups. Now “DuckCon” also.
  - Blog at duckdb.org (has more page views than cwi.nl ;-)
  - Getting to Hacker News front page, e.g. by tweeting
- Results
  - 200K downloads last month (PyPi + CRAN + ..)
  - 5400+ github stars
  - 40 venture capitalists calling
  - DuckDB Labs start-up, plans for scale-ups

Also: academic adoption for research prototypes (e.g. GrainDB, U Waterloo)

https://duckdb.org/ https://shell.duckdb.org

Centrum Wiskunde & Informatica

Outline

- **Intro**
  - Data systems research @ CWI DA
- **Past**
  - MonetDB & VectorWise + impact
- **Present**
  - DuckDB ("in-process analytics")
- **Future**
  - Learned Data Formats
  - SQL:2023 (Property Graph Queries)
  - Responsible Decentralized Data Architectures

Learned Data Formats

- **Problem**: data scientists/cloud-users store data inefficiently
  - Correlated columns, denormalized tables
  - Wrong datatypes (dates, numbers as string)
  - No key constraints specified
- Idea: **learn** better ways of storing the data under the hood
  - Detect constraints and exploit them in queries
  - Store normalized, decorrelated data, in the best datatype
  - new low-level encodings optimized for SIMD/GPU/TPU (**FastLanes**)

Diego Tomé

Azim Afroozeh

CWI
Centrum Wiskunde & Informatica

https://ldbcouncil.org
LDBC
The graph & RDF benchmark reference

SQL/PGQ: Property Graph Query

- ISO started working with LDBC on graph query languages
- SQL:2023 will introduce PGQ:
  - Path-finding and pattern-matching

```
CREATE PROPERTY GRAPH pg
  VERTEX TABLES (person PROPERTIES (personId, firstName))
  EDGE TABLES (knows SOURCE person DESTINATION person PROPERTIES(creationDate))
)
```

Gábor Szárnyas

SQL/PGQ: Property Graph Query

- ISO started working with LDBC on graph query languages
- SQL:2023 will introduce PGQ:
    - Path-finding and pattern-matching

Gábor Szárnyas

```
CREATE PROPERTY GRAPH pg (
  VERTEX TABLES (person PROPERTIES (personId, firstName))
  EDGE TABLES (knows SOURCE person DESTINATION person PROPERTIES(creationDate))
)
```

```
SELECT gt.src, gt.dst
FROM GRAPH_TABLE (pg, MATCH (p1 IS person)-[k IS knows]*->(p2 IS person)
                      WHERE k1.creationDate = 2022/06/01
                      COLUMNS (p1.name AS src, p2.name AS dst)) gt
```

SQL/PGQ: Property Graph Query

CWI is working on a SQL/PGQ module for DuckDB

Research Challenges:

- vectorized pattern matching? (worst-case-optimal, factorized)
- vectorized many–to-many path-finding algorithms
- SQL/PGQ query optimization / statistics
- In-process in-database GNN training on tabular PGs

Responsible Distributed Data Architectures

- VIDI grant 2022 by Hannes Mühleisen
- Problem:
  - centralizing data in the cloud is dominant architecture
  - creates privacy nightmares
- Goal:
  - provide an easy-to-use decentralized alternative
  - application data stays in a user-owned on-device database
  - design infra for managing large fleets of DuckDB databases
  - design infra for aggregated data collection to enable global queries
  - enable apps like distributed differential privacy, federated learning

CWI
Centrum Wiskunde & Informatica

CWI Database Architectures (DA)

Dutch Seminar on Data Systems Design (every other Friday 16:00-17:30)
subscribe: https://dsdsd.da.cwi.nl with UvA, TU/e, TuD

Dutch News on DA & DuckDB:

- bit.ly/fd-boncz (Financieel Dagblad)
- bit.ly/tweakers-duckdb (Tweakers)

@cwi_da bit.ly/cwi-da
@duckdb duckdb.org
@hfmuehleisen duckdblabs.com

Thanks to all! **Consider joining us!**