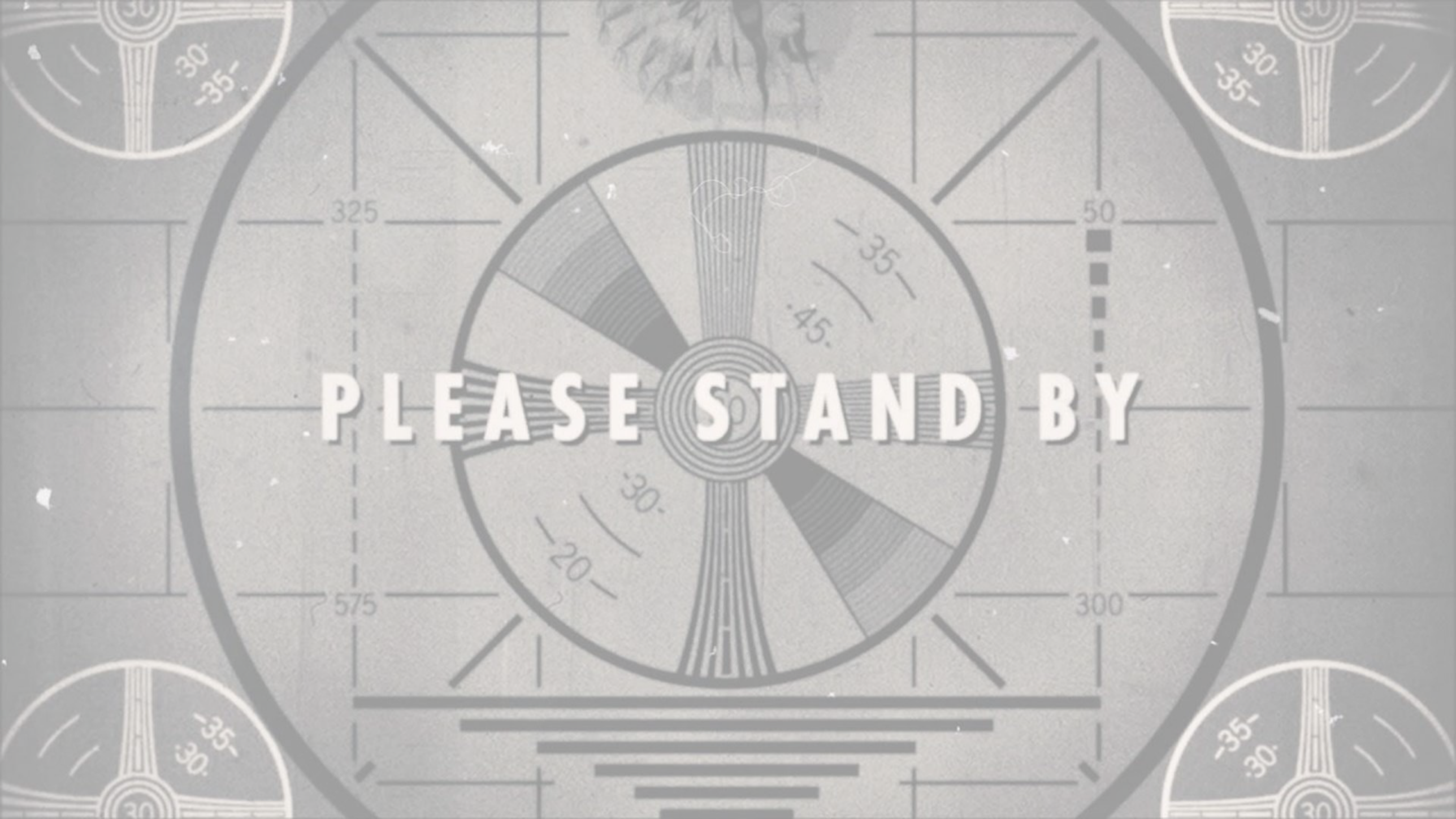

PLEASE STAND BY

Hannes Mühleisen & Mark Raasveldt — DuckCon #7

2026-06-24

State of the Duck

DuckDB


ON AIR

qa.duckcon.org

Thanks to our sponsors

GOLD SPONSOR

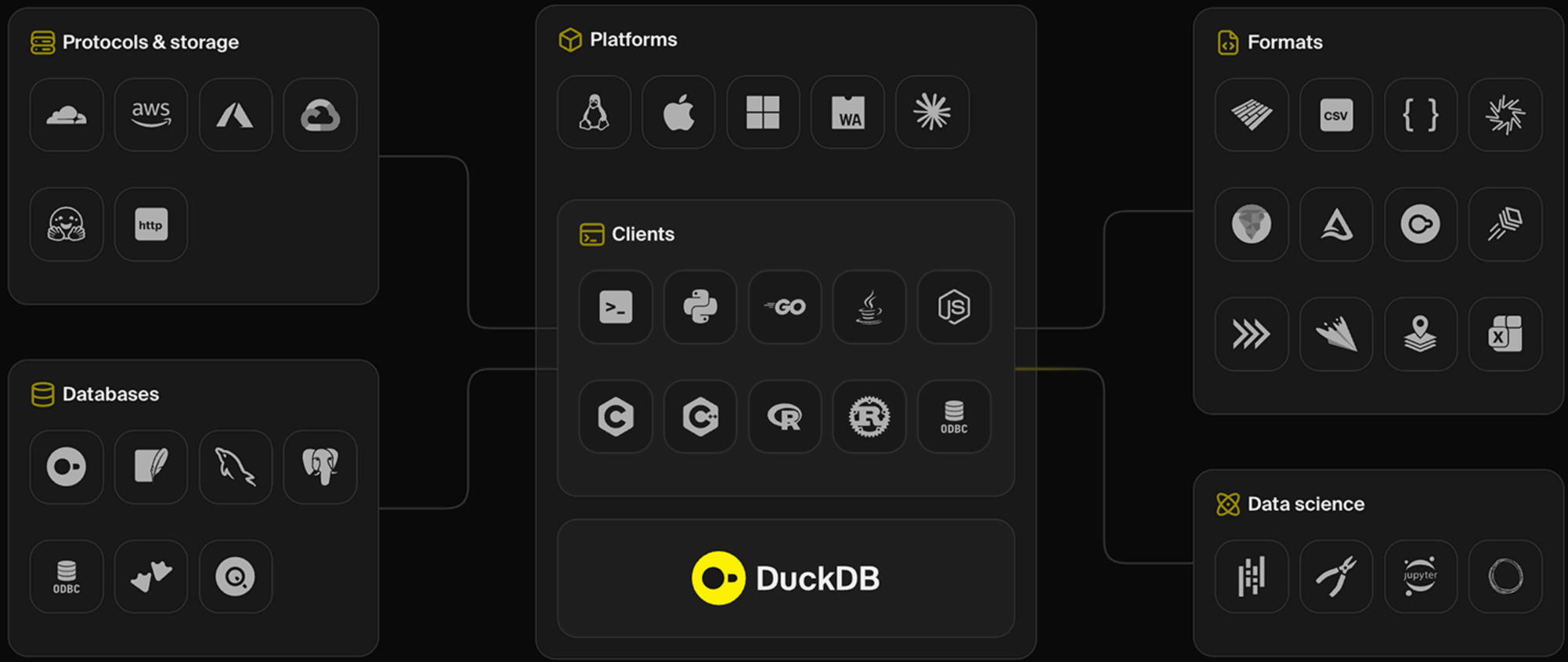
SILVER SPONSOR

Data *Fear* → Data **Confidence**

DuckDB

Built for your stack

DuckDB has native clients and integrations with the data ecosystem



DuckDB Adoption

	Downloads /month	Extensions installed /month	Unique web visitors /month
AUG 2024	4M	17M	600k
JAN 2025	10M	32M	1.8M
JUN 2026	40M	160M	7M

DuckDB Adoption

	GitHub stars	LinkedIn followers	DB-Engines ranking
AUG 2024	21k	23k	OVERALL 62 | RELATIONAL 35
JAN 2025	26k	35k	OVERALL 55 | RELATIONAL 33
JUN 2026	39k	50k	OVERALL 42 | RELATIONAL 27

DuckLake

DuckLake v1.0: The Lakehouse Format Built on SQL Reaches Production-Readiness

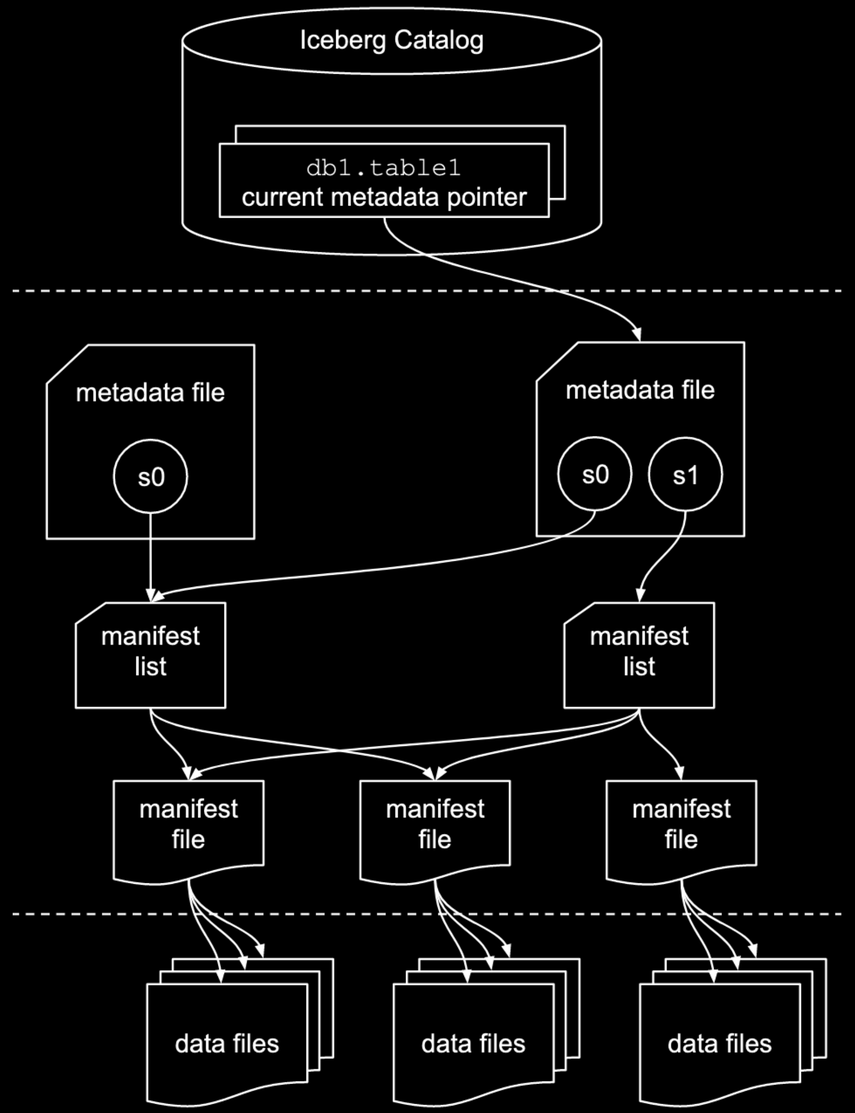
The DuckDB team
2026-04-13 · 23 min

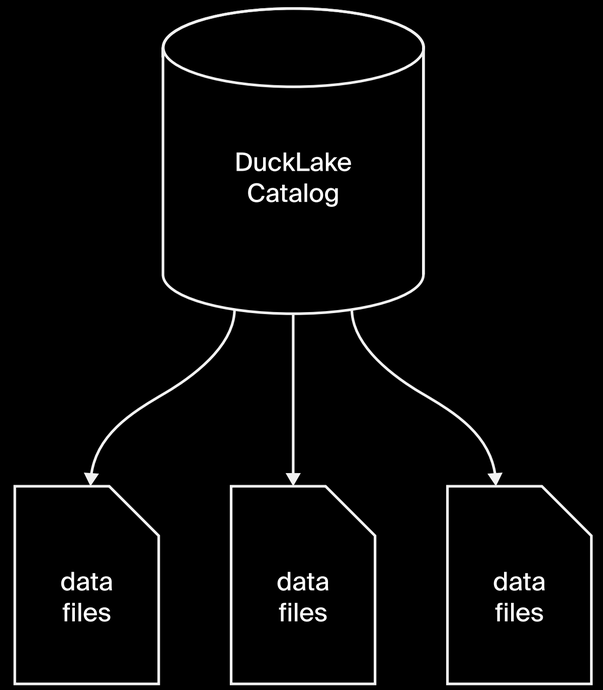
TL;DR: We are happy to release DuckLake v1.0, a production-ready lakehouse format specification built on SQL. Its reference implementation, the `ducklake` DuckDB extension, is available as of today in DuckDB v1.5.2.

In May 2025, we published the DuckLake manifesto, where we explained what motivated us to work on DuckLake. Here's a quick recap: we basically outlined how, in our view, it makes much more sense to store all metadata of a lakehouse in a **database** rather than in scattered files in object storage. This is why we created DuckLake.

> The manifesto is much more compelling, we recommend you read it!

Today, we are happy to announce **DuckLake v1.0**, almost a year after we released our first sketch of the specification. This is a production-ready release with guaranteed backward-compatibility. DuckLake v1.0 ships a stable specification, a feature-rich and fast reference implementation (the DuckDB `ducklake` extension), as well as a roadmap for future


Iceberg Catalog
db1.table1
current metadata pointer
metadata file
s0
metadata file
s0
s1
manifest list
manifest list
manifest file
manifest file
manifest file
data files
data files
data files


DuckLake Catalog
data files
data files
data files

ESSAY

DuckDB and DuckLake: Why We Bet the Company on the Duck Stack

MIKE RITCHIE · FEBRUARY 6, 2026 · 12 MIN READ

Downloads / Month

2.1M

2.6M

3.0M

Data: 23 June 2026

Quack

SERVER

```
CALL quack_serve(
    'quack:localhost',
    token='asdf'
);

CREATE TABLE fuu AS
    FROM (VALUES(42));

FROM fuu;
FROM fuu;
```

CLIENT

```
CREATE SECRET (
    TYPE quack,
    TOKEN 'asdf'
);

ATTACH 'quack:localhost' AS remote;


FROM remote.fuu;
FROM remote.query($$ FROM fuu $$);
```

SERVER

```
CALL quack_serve(
    'quack:localhost',
    token='asdf'
);

CREATE TABLE fuu AS
    FROM (VALUES(42));

FROM fuu;
```

CLIENT

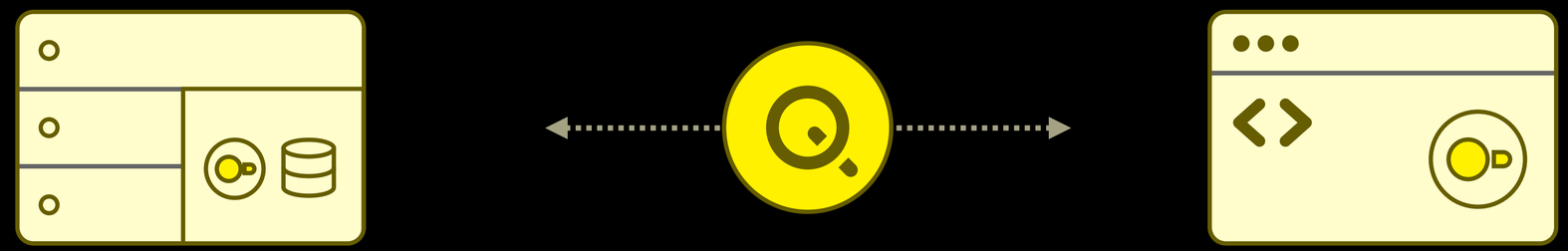
```
CREATE SECRET (
    TYPE quack,
    TOKEN 'asdf'
);

ATTACH 'quack:localhost' AS remote;

CONNECT remote;
FROM fuu;
DISCONNECT;
```

SERVER

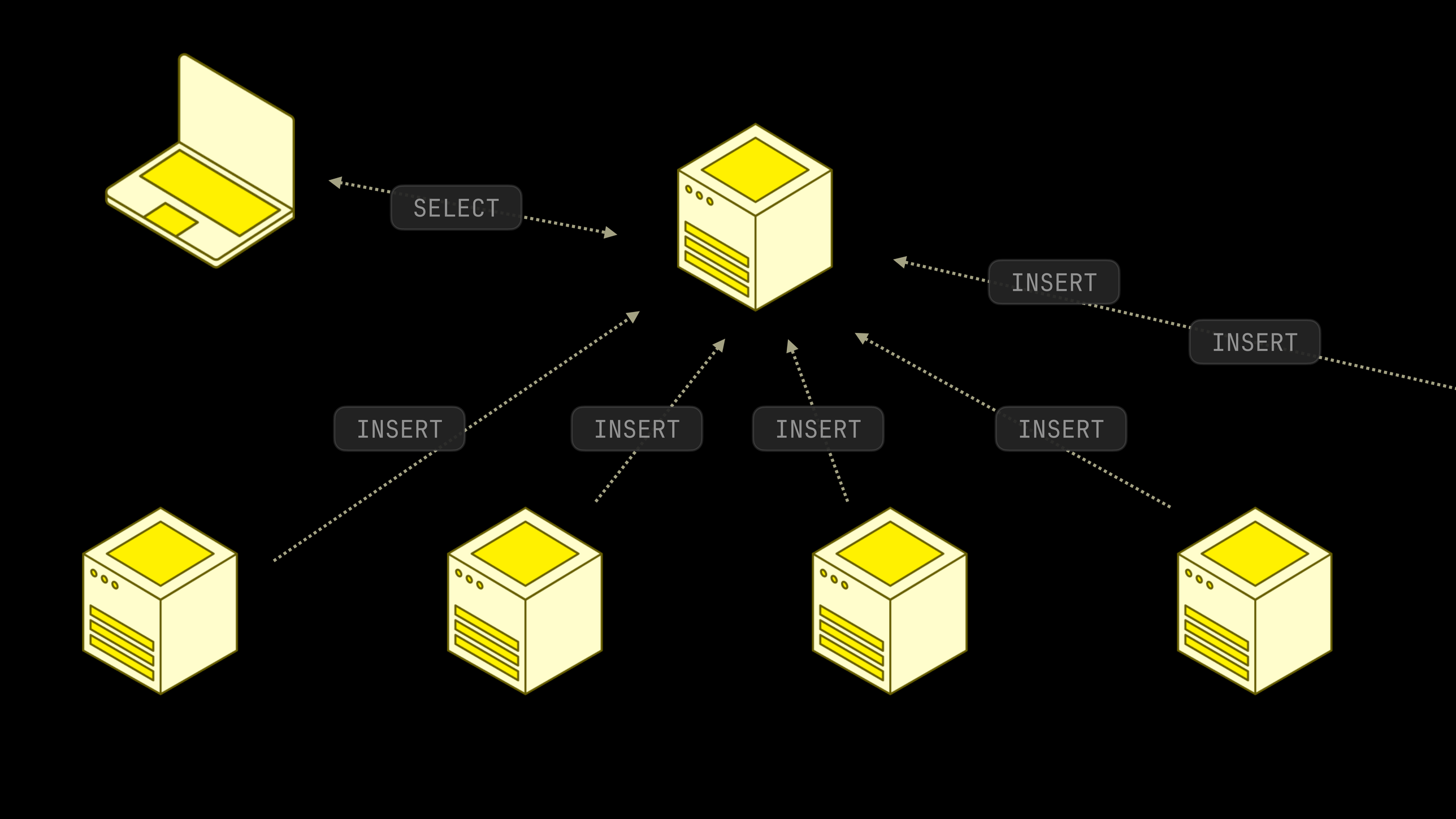
CLIENT



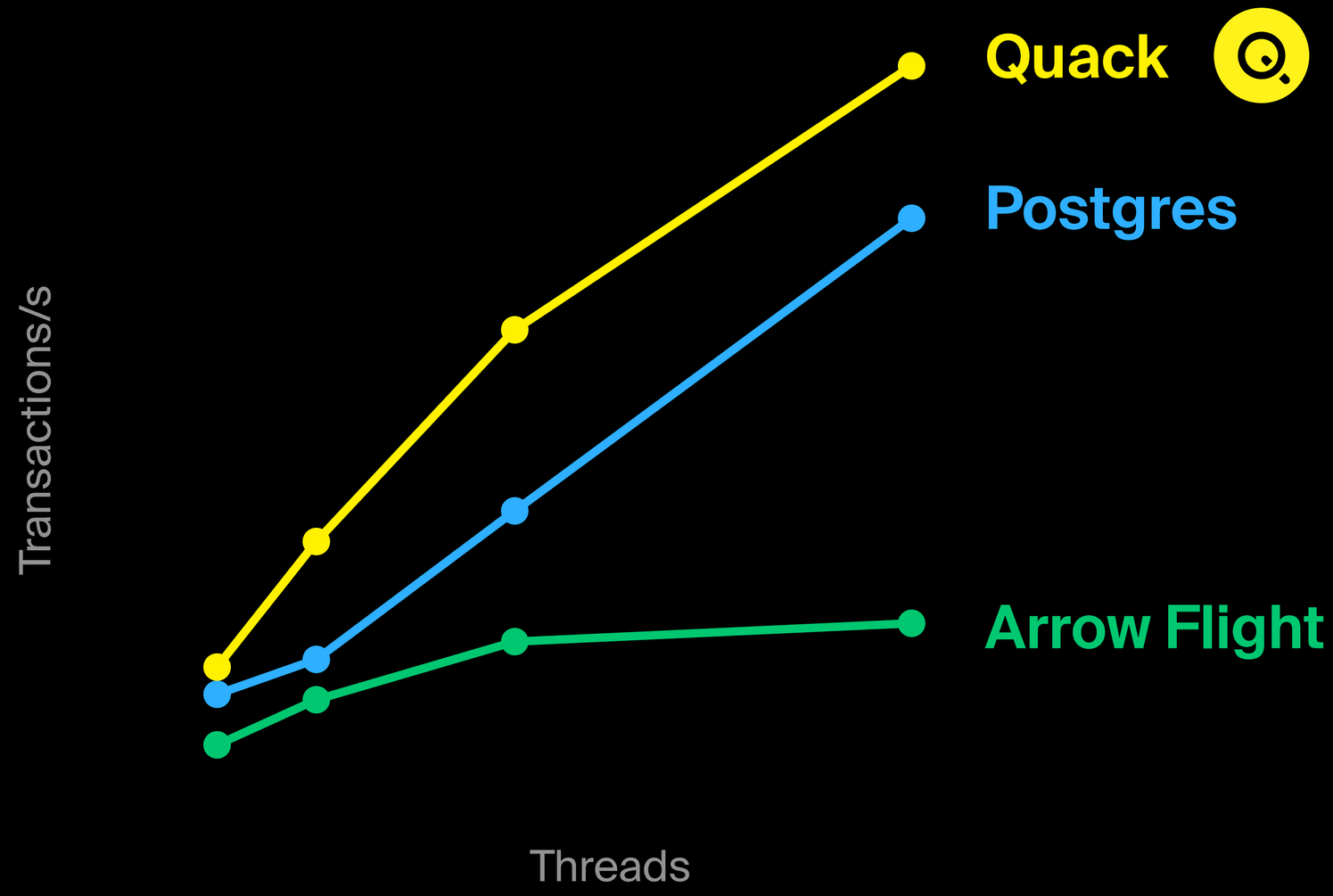
Rebranding

Rebranding

DuckDB 2.0 (Cinnamon Teal)

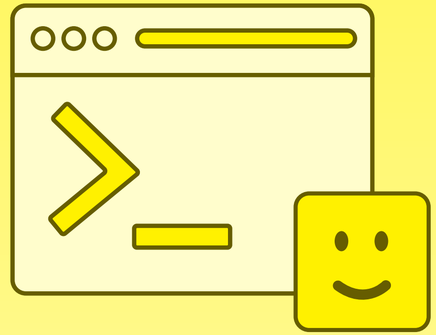

SELECT
INSERT
INSERT
INSERT
INSERT
INSERT
INSERT

Quack Performance



01

SQL Features



VARIANT Type

```
CREATE TABLE variant_table (j VARIANT);
INSERT INTO variant_table VALUES (JSON '{"hello":"world", "best_number": 42}');
FROM variant_table;
```

j variant
{'hello': world, 'best_number': 42}

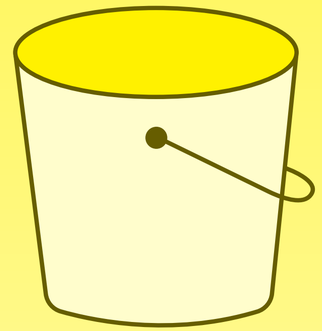
Trigger

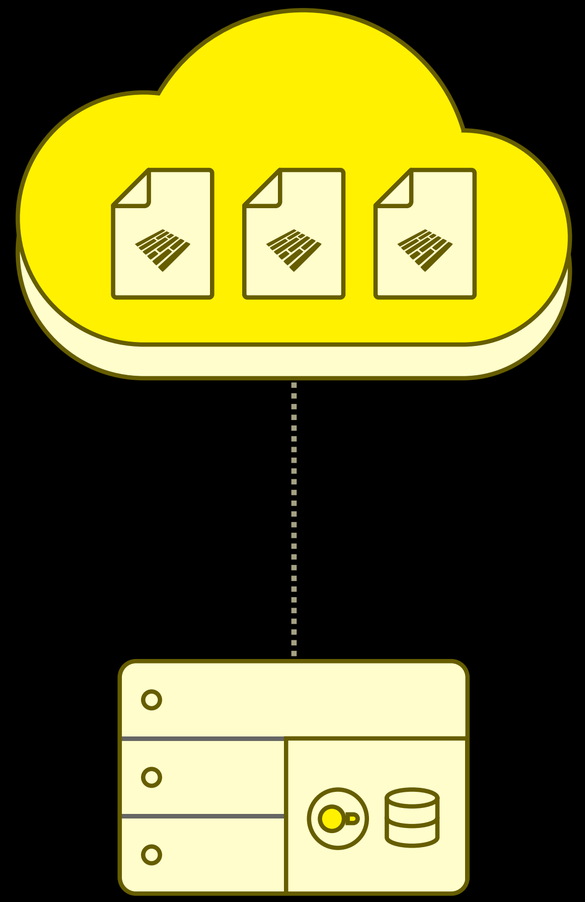
```
CREATE TABLE target (id INTEGER);
CREATE TABLE audit  (id INTEGER);

CREATE TRIGGER t
    AFTER INSERT ON target
    FOR EACH ROW INSERT INTO audit VALUES (NEW.id);
```

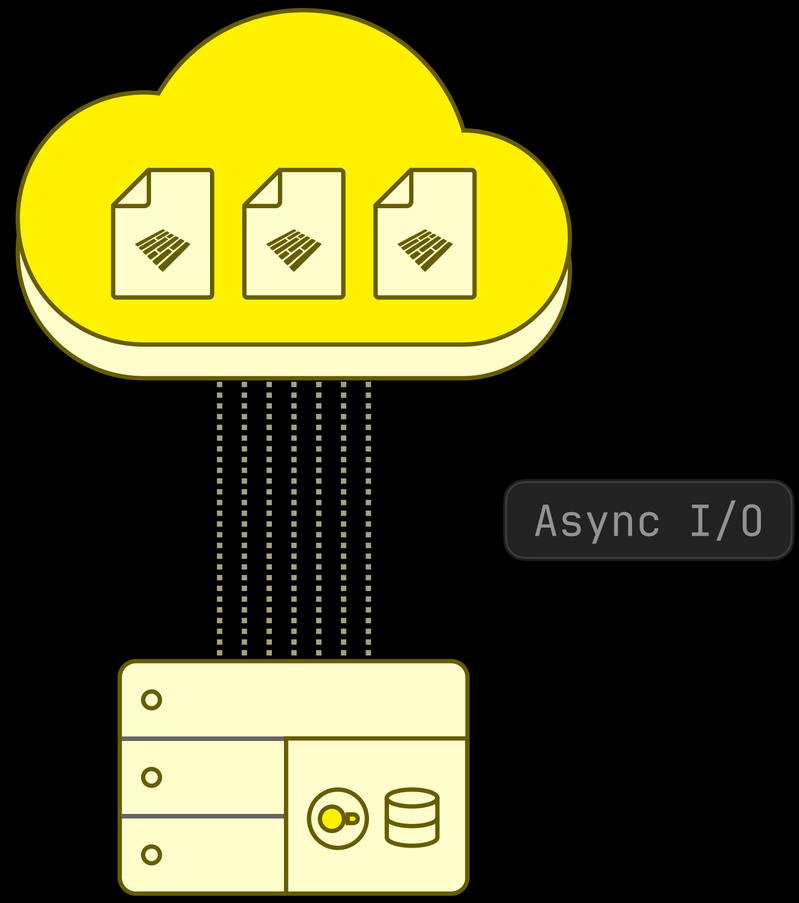
02

Object Stores

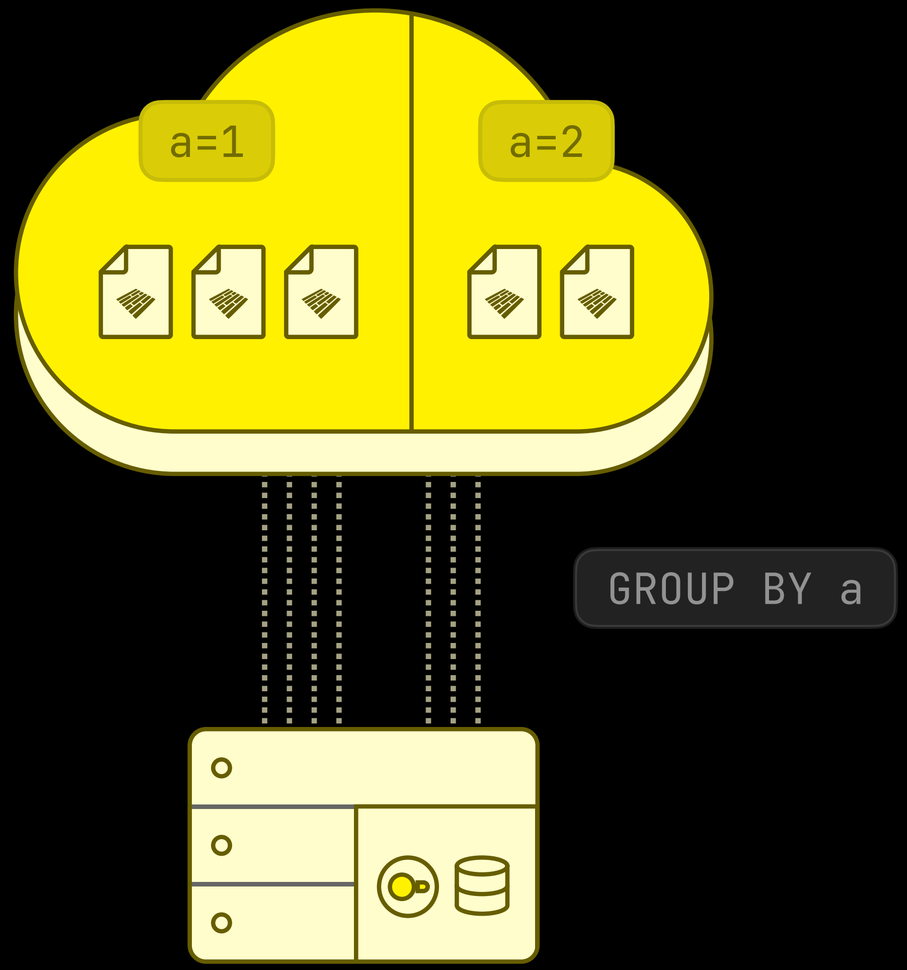




Async I/O

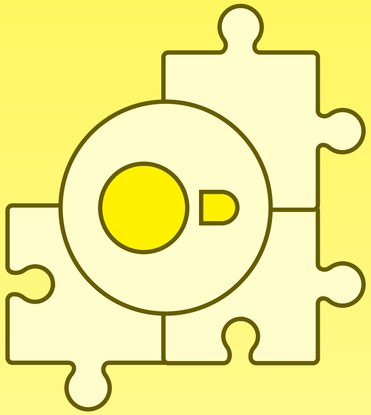


Partitioning



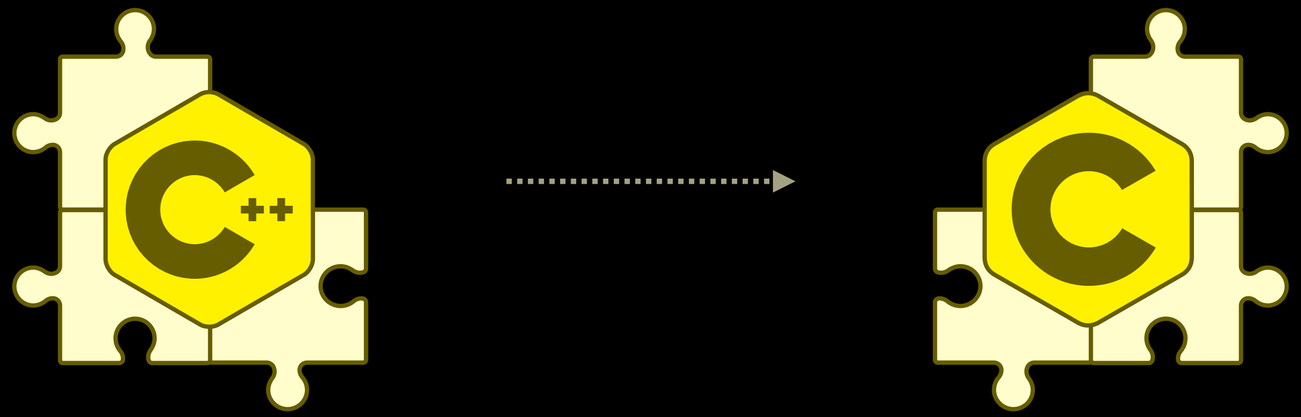
03

Under the Hood

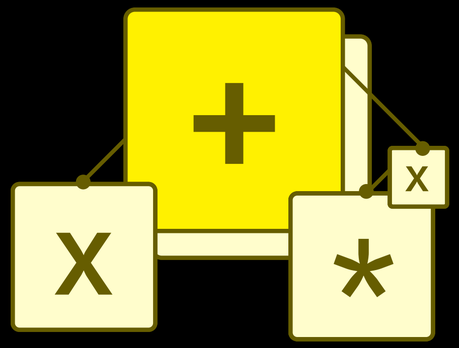


C API v2

C API v2 for Extensions



New Extensible Parser Default



qa.duckcon.org

Main track, session 1

15:00	**State of the Duck**	Hannes Mühleisen & Mark Raasveldt	DuckLabs
15:30	**Effortlessly modernize your legacy: SQLFrame migrates PySpark to DuckDB without code changes**	Nicolas Renkamp	Merck KGaA
15:45	**Phages and DuckDB: The mighty duo against the giants**	Herminio Vazquez-Cano Virginie Grosboillot	ETH Zürich U. of Ljubljana
16:00	**ggsql: A grammar of graphics for SQL**	Teun van den Brand	Posit PBC

DuckCon #7

Main track, session 2

16:35	**How we built a SQL layer over our user listening history for agentic access**	Kian Mehrabani	Spotify
16:50	**Grep your lakehouse: Search-first retrieval for DuckDB-powered agents**	Sylvain Utard	Altertable.ai
17:05	**When the sea lion learns to quack: DuckDB as a MariaDB storage engine**	Roman Nozdrin	MariaDB Corporation

DuckCon #7

Lightning talks

17:20	**Native app development using DuckDB**	Milla Henriksson	Toyota Gazoo Racing World Rally Team
17:26	**Making Iceberg walk and talk like Postgres**	Marco Slot	Snowflake
17:32	**DuckLake on Hetzner: Your own data lakehouse for under €15 a month**	Floyd Berndsen	
17:38	**DuckDB powering cancer genomics research**	Thomas Visser	Hartwig Medical Foundation
17:44	**DuckDB as an analytical runtime: Building local-first analytics apps with SQLRooms**	Ilya Boyandin	Foursquare / GeoVisually GmbH
17:50	**A laptop, a duck, and twenty years of wind: Auditing UK energy policy without a cluster**	Barry Smart	endjin