

DuckDB and Its Ecosystem

The DuckDB story now extends beyond the original database. Here is how the projects and organizations around it developed, and how they fit together today.

● 2018

Hello World

Mark Raasveldt and Hannes Mühleisen began developing DuckDB in 2018 while working as database researchers at Centrum Wiskunde & Informatica (CWI) in Amsterdam. Their work grew from database systems research, later documented in [papers on DuckDB's design and architecture](#), including its embedded analytical model.

DuckDB was open source from day one. In 2019, Mark and Hannes announced [DuckDB v0.1](#) at the SIGMOD conference in Amsterdam. DuckDB could run as a single binary or inside Python, R, or another host process, giving users an analytical SQL engine without requiring them to deploy and maintain a separate database server.

This in-process architecture removed a familiar source of friction. Developers could query files and data where they were already working. Data scientists could bring SQL into a notebook without first deploying database infrastructure. Product teams could embed analytical processing directly in their applications.

As more people began using DuckDB, the project needed a permanent engineering home and a structure that could support long-term development.

● 2021

The Ecosystem Begins to Take Shape

On July 14, 2021, Mark and Hannes announced [DuckDB Labs as a CWI spin-off](#). The company brought the core engineering team together and created a commercial route for organizations that needed support or development work.

The independent DuckDB Foundation was incorporated later that year. It holds most of DuckDB's intellectual property and safeguards the project's continuity under the permissive MIT license.

The company and Foundation were designed for different responsibilities. DuckDB Labs became the engineering and commercial home of the core team. The Foundation provided the legal structure around intellectual property, trademarks, and long-term open-source continuity. Day-to-day development remained with the engineers and contributors working on the project.

2022

MotherDuck Adds a Managed Cloud Path

DuckDB's local and embedded model covered a wide range of analytical work. Some teams also wanted centralized storage, sharing, collaboration, access control, and managed operations. MotherDuck was formed to build that cloud experience on DuckDB. In November 2022, DuckDB Labs announced [a long-term partnership with MotherDuck](#).

MotherDuck contracts with DuckLabs for engineering work, supports the DuckDB Foundation, and contributes technology to the wider project. DuckLabs gives MotherDuck direct access to the people developing and maintaining the database.

The partnership works through clearly defined roles. MotherDuck operates the managed cloud service, the Foundation holds DuckDB's intellectual property, and DuckLabs leads core engineering and commercial support around the open-source projects. The companies collaborate where their technical work overlaps.

2024

DuckDB Reaches v1.0

On June 3, 2024, the team [released DuckDB v1.0.0](#).

The release emphasized stability rather than a long list of new features. DuckDB had gained backward compatibility for its storage format, extensive testing across the engine, and a more cautious approach to changes in the SQL dialect and C API. Version 1.0 expressed the team's confidence that developers could build applications on DuckDB with clearer expectations about compatibility over time.

It was also a marker of adoption. Six years after the first code was written, DuckDB had moved from a research project into production systems, data workflows, and embedded products across a growing community.

Later that year, MotherDuck made significant contributions to the [new DuckDB Node.js client](#), another example of technical work shared across the ecosystem.

2025

The Work Expands beyond the Original Database

In March 2025, DuckLabs and MotherDuck introduced the DuckDB Local UI, a browser-based interface delivered through the DuckDB ui extension. It runs queries against a local DuckDB instance and provides an explicit path to connect to MotherDuck when users want cloud capabilities. The project showed how the two companies could combine local and managed experiences in one workflow.

Two months later, the DuckLabs team introduced DuckLake, an open lakehouse format that keeps catalog metadata in a SQL database while storing data in the open Parquet format.

DuckLake applied the team's database engineering experience to a different layer of the data stack. The work now covered the analytical engine and the organization of shared lakehouse data. The name DuckDB Labs was beginning to describe only part of what the company maintained.

● 2026, Q2 **Broader Horizons**

DuckLake reached version 1.0 on April 13, 2026, establishing a production-ready specification and reference implementation.

The Quack technical preview followed in May. Quack is the DuckDB client-server protocol, designed for workloads that need remote connections or multiple clients. It is available as a beta release while the team develops it toward stable status as DuckDB v2.0.

The change from DuckDB Labs to DuckLabs came later that month. By then, the company employed more than 30 people working across DuckDB, DuckLake, Quack, extensions, language bindings, and related systems engineering.

● 2026, Q3 **DuckLabs Is Joining AWS**

On August 26, 2026, DuckLabs announced that it is joining Amazon Web Services. The DuckLabs team stays in Amsterdam and remain focused on the development of the core projects – DuckDB, DuckLake and Quack.

These projects will remain the intellectual property of the DuckDB Foundation and open source under the MIT license. The scope of the community support is extended to accept larger volumes of issues and pull requests, and will cover more components.