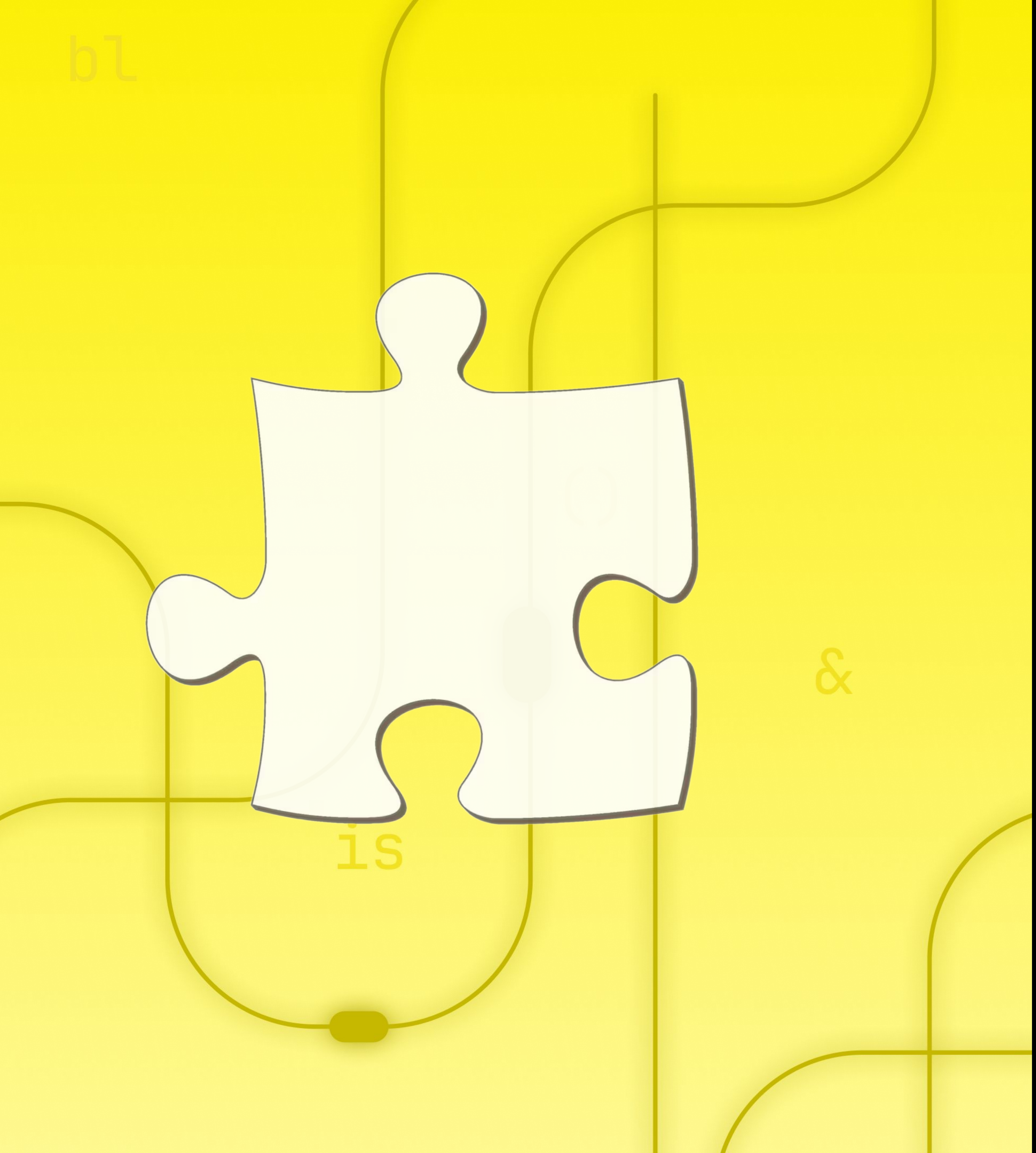


DuckDB Extensions

the past,
the present,
the future

Sam Ansmink, January 2026

Today’s menu: DuckDB Extensions

- Introduction
- The past
- The present
- The future

Introduction - who am I?

- Sam Ansmink
- Working at DuckDB Labs (>4 years)
- Master thesis on DuckDB @ CWI
- Worked on various extensions
  - httpfs, aws, parquet, azure, iceberg, delta
- Worked on extension framework
  - extension templates, extension APIs, CI/CD pipelines

Introduction - who are you?

Introduction - who are you?

- Who has used DuckDB Extensions?

Introduction - who are you?

- Who has used DuckDB Extensions?
- Who has written DuckDB Extensions?

Introduction - DuckDB Extensions

So what ***are*** DuckDB extensions?

Introduction - DuckDB Extensions

So what ***are*** DuckDB extensions?

- Add/alter **functionality** to core feature set

Introduction - DuckDB Extensions

So what ***are*** DuckDB extensions?

- Add/alter **functionality** to core feature set
- Table Functions, Types, Filesystems, Catalogs, ...

Introduction - DuckDB Extensions

So what ***are*** DuckDB extensions?

- Add/alter **functionality** to core feature set
- Table Functions, Types, Filesystems, Catalogs, ...
- Examples:
  - json → reading/writing json files

Introduction - DuckDB Extensions

So what ***are*** DuckDB extensions?

- Add/alter **functionality** to core feature set
- Table Functions, Types, Filesystems, Catalogs, ...
- Examples:
  - json → reading/writing json files
  - postgres → integration with postgres

Introduction - DuckDB Extensions

So what ***are*** DuckDB extensions?

- Add/alter **functionality** to core feature set
- Table Functions, Types, Filesystems, Catalogs, ...
- Examples:
  - json → reading/writing json files
  - postgres → integration with postgres
  - gsheets → integration with google sheets

Introduction - DuckDB Extensions

So what ***are*** DuckDB extensions?

- Add/alter **functionality** to core feature set
- Table Functions, Types, Filesystems, Catalogs, ...
- Examples:
  - json → reading/writing json files
  - postgres → integration with postgres
  - gsheets → integration with google sheets
  - psyduck → various pokemon related functionality

Introduction - DuckDB Extensions

So why have DuckDB extensions at all?

Introduction - DuckDB Extensions

So why have DuckDB extensions at all?

- Binary size
- External dependencies
- Functional incompatibility
- Maintainer

Introduction - DuckDB Extensions

Now how do I ***use*** DuckDB extensions?

Introduction - DuckDB Extensions

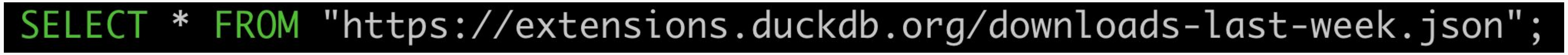
Now how do I ***use*** DuckDB extensions?

- You probably already are!

Introduction - DuckDB Extensions

Now how do I ***use*** DuckDB extensions?

- You probably already are!
- Extensions can be auto-installed and loaded:



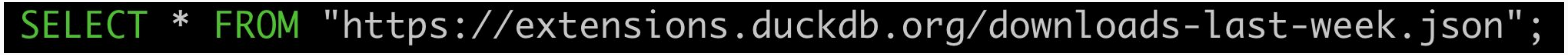
```
SELECT * FROM "https://extensions.duckdb.org/downloads-last-week.json";
```

  - Installs + loads httpfs + json extensions

Introduction - DuckDB Extensions

Now how do I ***use*** DuckDB extensions?

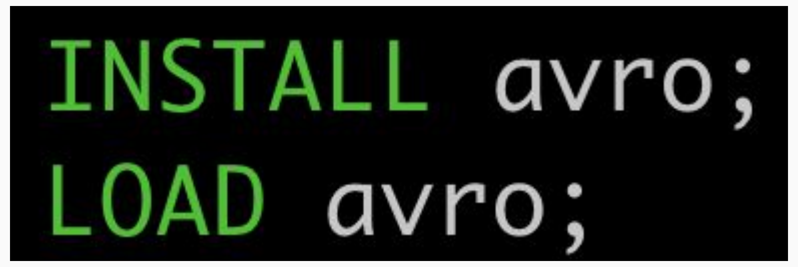
- You probably already are!
- Extensions can be auto-installed and loaded:



```
SELECT * FROM "https://extensions.duckdb.org/downloads-last-week.json";
```

  - Installs + loads httpfs + json

- Or, install & load manually:



```
INSTALL avro;
LOAD avro;
```

Introduction - DuckDB Extensions

Where do I get em?

Introduction - DuckDB Extensions

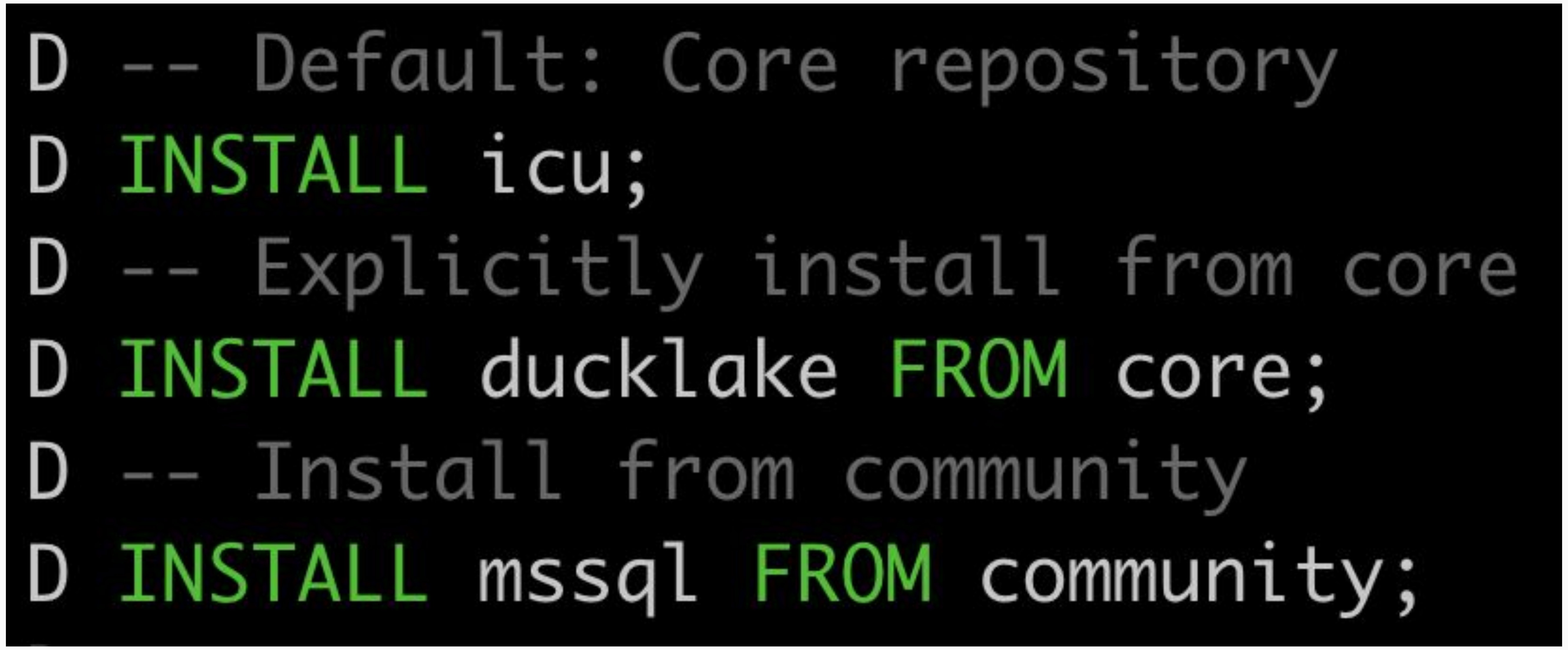
Where do I get em?

- Extension Repositories
    - Core (default)
    - Community

Introduction - DuckDB Extensions

Where do I get em?

- Extension Repositories
  - Core (default)
  - Community



```
D -- Default: Core repository
D INSTALL icu;
D -- Explicitly install from core
D INSTALL ducklake FROM core;
D -- Install from community
D INSTALL mssql FROM community;
```

The Past

2019 2020 2021 2022 2023 2024 2025 2026

The Past

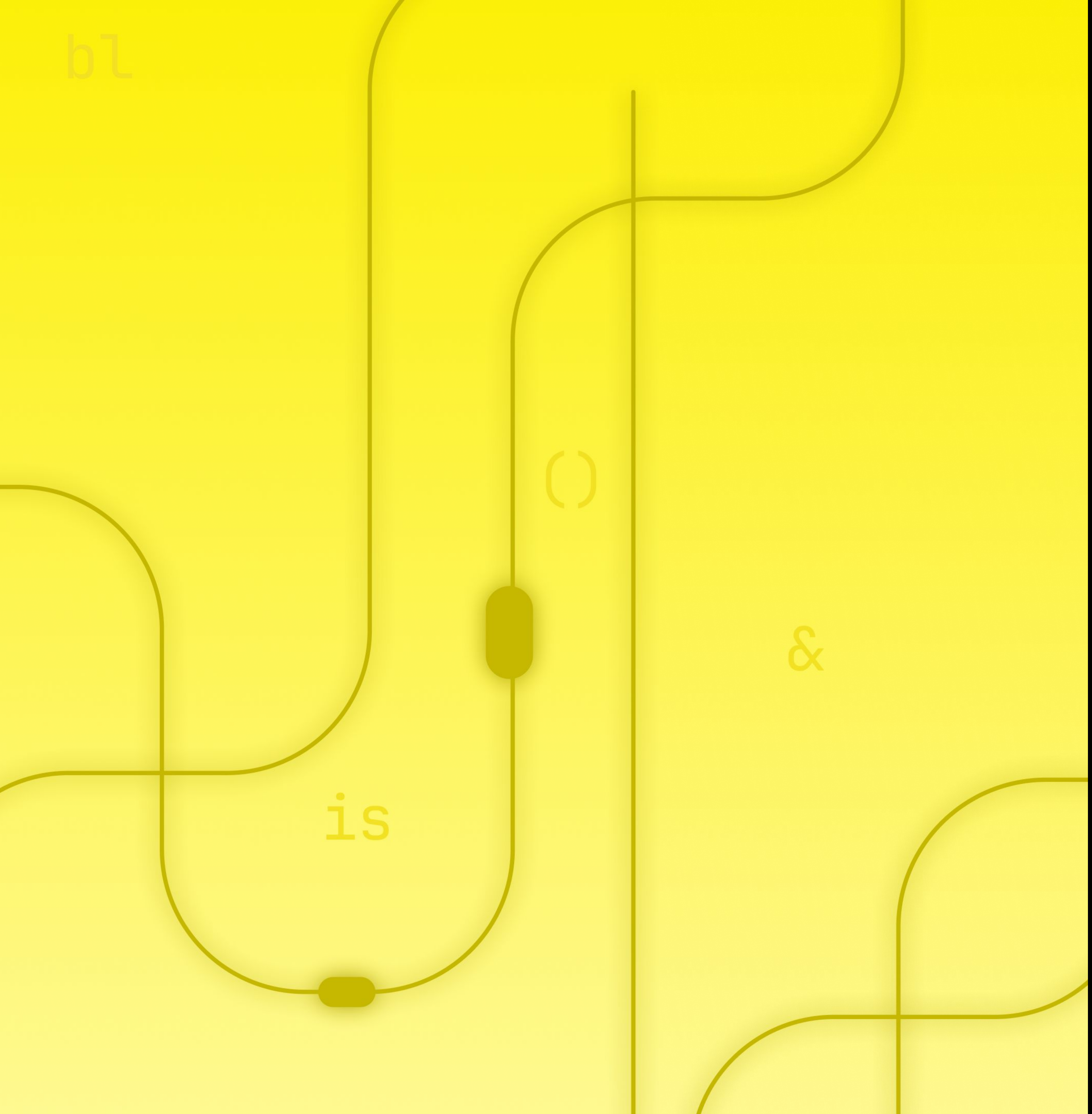
2019
2020
2021
2022
2023
2024
2025
2026
First DuckDB commit
DuckDB v0.1.0
DuckDB v1.0.0
DuckDB v1.4.4

The Past

2019
2020
2021
2022
2023
2024
2025
2026
First DuckDB commit
DuckDB v0.1.0
Extension Mechanism
Core Repository
C++ Extension Template
C Extension Template
DuckDB v1.0.0
DuckDB v1.4.4
Community Repository
Rust Extension Template

The Past

2019
2020
2021
2022
2023
2024
2025
2026
First DuckDB commit
DuckDB v0.1.0
Extension Mechanism
Core Repository
C++ Extension Template
C Extension Template
DuckDB v1.0.0
DuckDB v1.4.4
ICU Extension
Parquet Extension
HTTPFS Extension
SQLite Extension
JSON Extension
Iceberg Extension
Spatial Extension
Community Repository
Rust Extension Template
DuckLake Extension



The Present

The Present - DuckDB Extensions

The Present - DuckDB Extensions

DuckDB Extensions are ubiquitous

The Present - DuckDB Extensions

DuckDB Extensions are ubiquitous

- Core
  - 32 Extensions
  - ~27.5M downloads a week
- Community
  - 145 Extensions
  - ~500K downloads week

The Present - DuckDB Extensions

- DuckDB Extensions are ubiquitous
  - Core
    - 32 Extensions
    - ~27.5M downloads a week
  - Community
    - 145 Extensions
    - ~500K downloads week
- For reference: DuckDB Python Client
  - ~6M downloads a week

The Present - Building DuckDB Extensions

How are extensions built extensions today?

The Present - Building DuckDB Extensions

How are extensions built extensions today?

- extension-template
  - C++
  - Unstable API

The Present - Building DuckDB Extensions

How are extensions built extensions today?

- extension-template (recommended)
  - C++
  - Unstable API

The Present - Building DuckDB Extensions

How are extensions built extensions today?

- extension-template (recommended)
  - C++
  - Unstable API
- extension-template-rs (experimental)
  - Rust
  - Unstable API

The Present - Building DuckDB Extensions

How are extensions built extensions today?

- extension-template (recommended)
  - C++
  - Unstable API
- extension-template-rs (experimental)
  - Rust
  - Unstable API
- extension-template-c (experimental)
  - C / C++
  - Stable API

The Present - Maintaining DuckDB Extensions

How are extensions maintained today?

The Present - Maintaining DuckDB Extensions

How are extensions maintained today?

- Primary Core Extensions
- Secondary Core Extensions

The Present - Maintaining DuckDB Extensions

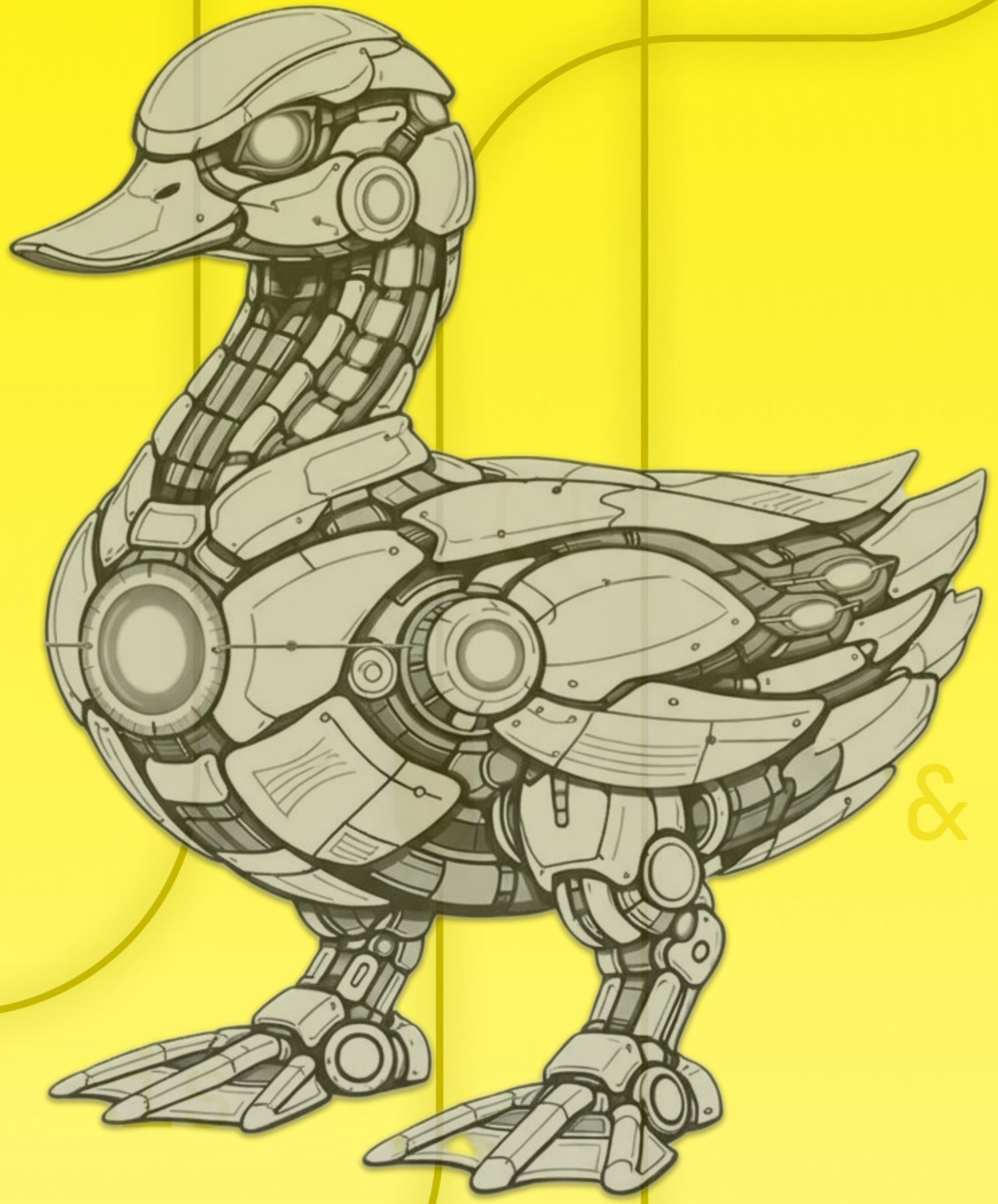
How are extensions maintained today?

- Primary Core Extensions
- Secondary Core Extensions
- Third Party Core Extensions

The Present - Maintaining DuckDB Extensions

How are extensions maintained today?

- Primary Core Extensions
- Secondary Core Extensions
- Third Party Core Extensions
- Community Extensions



The Future

The Future

The Future - Challenges

The Future - Challenges

Main challenge: Unstable C++ API

The Future - Challenges

Main challenge: Unstable C++ API

- Heavy on CI/CD
- Heavy on extension maintenance
- Hard to document well

The Future - Challenges

Main challenge: Unstable C++ API

- Heavy on CI/CD
- Heavy on extension maintenance
- Hard to document well

The Future - Challenges

Main challenge: Unstable C++ API

- Heavy on CI/CD
- Heavy on extension maintenance
- Hard to document well

We have a solution!

The Future - Challenges

Main challenge: Unstable C++ API

- Heavy on CI/CD
- Heavy on extension maintenance
- Hard to document well

We have a solution!

- The DuckDB C extension API
    - Stability
    - Good Interoperability (e.g. Rust)

The Future - Roadmap

Main goals

The Future - Roadmap

Main goals

- Expand stable C extension API

The Future - Roadmap

Main goals

- Expand stable C extension API
- Stabilize both
  - extension-template-rs
  - extension-template-c
- Migrate core extensions to stable APIs

Takeaways

- DuckDB has an extensive extension ecosystem
- The C++ extension API
    - Current standard
    - Reaching deep into DuckDB internals
- The C extension API
    - Experimental
    - Power the next-gen of extensions