Anarchy in the Database

Survey and Evaluation of Database Management System Extensibility

Abigale Kim
University of Wisconsin–Madison
abigale@cs.wisc.edu

Collaborators
Andy Pavlo (pavlo@cs.cmu.edu)
Dave Andersen (dga@cs.cmu.edu)
Marco Slot (marco.slot@crunchydata.com)

What is extensibility?

- Extensibility: DBMS lets custom software extend its capabilities
  - UDFs, OptimizerExtension
- Extension: an instance of this software

My Research

- Taxonomy: categorized different methods for support extensibility in database systems (extensibility types, interfaces, tooling)
- Survey: surveyed 6 DBMSs against our taxonomy
- Analysis: wrote an analysis tool to collect information about extensions and run compatibility testing
  - Smoke test: download two extensions, run their unit tests with both installed
- Website: https://db.cs.cmu.edu/pgexts-vldb2025/

How does DuckDB stack up against the other DBMSs?

The Competition

Survey: Types of Extensibility

	PostgreSQL	MySQL	MariaDB	SQLite	Redis	DuckDB
User-Defined Functions	**Yes** (408)	**Yes** (2)	**Yes** (1)	**Yes** (79)	**Yes** (57)	**Yes** (41)
User-defined Types	**Yes** (139)	No	**Yes** (13)	No	No	**Yes** (4)
Utility Commands	**Yes** (43)	No	No	No	No	No
Parser Modifications	No	**Yes** (2)	**Yes** (1)	No	No	**Yes** (4)
Query Processing	**Yes** (46)	**Yes** (7)	**Yes** (5)	No	No	**Yes** (4)
Storage Managers	**Yes** (44)	**Yes** (13)	**Yes** (18)	**Yes** (43)	No	**Yes** (9)
Index Access Methods	**Yes** (67)	No	No	No	No	**Yes** (3)
Client Authentication	**Yes** (17)	**Yes** (3)	**Yes** (10)	No	No	No
Version Examined	v16	v8	v11	v3	v7	v1
Number of Extensions	441	29	68	98	57	44+

Survey: Interfaces & Tooling

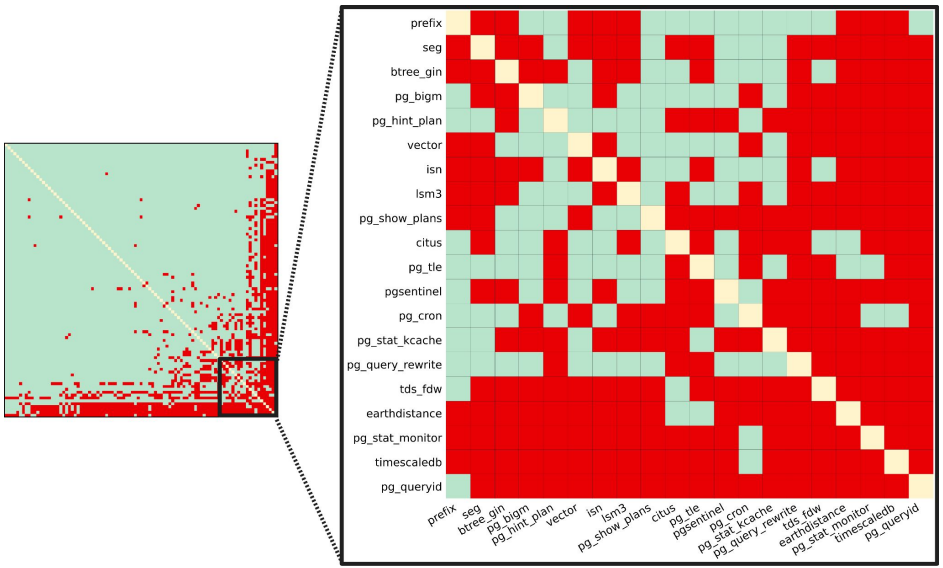
	PostgreSQL	MySQL	MariaDB	SQLite	Redis	DuckDB
Adding Components	Yes	Yes	Yes	Yes	Yes	Yes
Overriding Components	Yes	Yes	Yes	Yes	No	Yes
State Modification	All state	All state	All state	All state	Etxn. + Ephmrl.	All state
Isolation/Security	None	Low	Low	Medium	High	Low
Background Workers	Yes	Yes	Yes	No	No	No
Memory Allocation	Yes	Yes	Yes	Yes	Yes	Yes
Configuration Options	Yes	Yes	Yes	No	Yes	Yes
Source Code	Yes	Yes	Yes	Yes	No	Yes
Programming Languages	C, C++, Rust	C++	C++	C, Rust	C, Lua	C++
Installation Interface	SQL, configs	SQL	SQL	SQL	SQL, configs	SQL
Build & Test Tooling	Both	Testing	Testing	Both	None	Both
Package Manager	Yes (community)	No	Yes (OS)	Yes (community)	No	Yes

DuckDB Survey Highlights

- Vectorized UDFs: UDFs process batch of values
- You can add rules to the optimizer!
- Scanner extension implementation
- Super good build & test tooling
- Registry!!

DuckDB vs. PostgreSQL Analysis

PostgreSQL:



DuckDB: 0% compatibility errors!

1. Only 4/45 extensions have state sharing interfaces
2. DuckDB invokes extensions instead of extensions invoking themselves
3. Memory errors in C vs. C++

Fun Monstrosities in PostgreSQL

```
#if PG_VERSION_NUM>=150000
static shmem_request_hook_type  PreviousShmemRequestHook = NULL;
#endif
```

```
2023-07-25 05:04:55.945 UTC [687073] STATEMENT:  truncate table t;
2023-07-25 05:04:56.945 UTC [687074] ERROR:  deadlock detected
```

```
-- Create first test user
CREATE USER user1 password 'password';
ERROR:  password must contain both letters and nonletters
ALTER ROLE user1 SET pgaudit.log = 'ddl, ROLE';
```

```
2023-07-26 11:57:04.925 UTC [2087903] LOG:  database system is shut down
```

Takeaways

- DuckDB extensibility is up and on the rise
- DuckDB's got incredible tooling and comprehensive extensibility support
- PostgreSQL is insanely flexible (a great thing!) but unsafe (not a great thing!)
- Read my paper & go to my website (shameless plug): https://db.cs.cmu.edu/pgexts-vldb2025/