PERCONA

Open-source support for JS stored programs in Percona Server

Dmitry Lenev, FOSDEM 2025

Background

- Stored Programs in JavaScript in Oracle MySQL
  - Tech preview - December 2023
  - Generally available from MySQL 9.0 (July 2024)
  - Available in 9.x Innovation Releases

- Enterprise or Cloud - only feature
  - Not available in a free Community Edition
  - No source code publicly available

- Free and open-source alternative from Percona
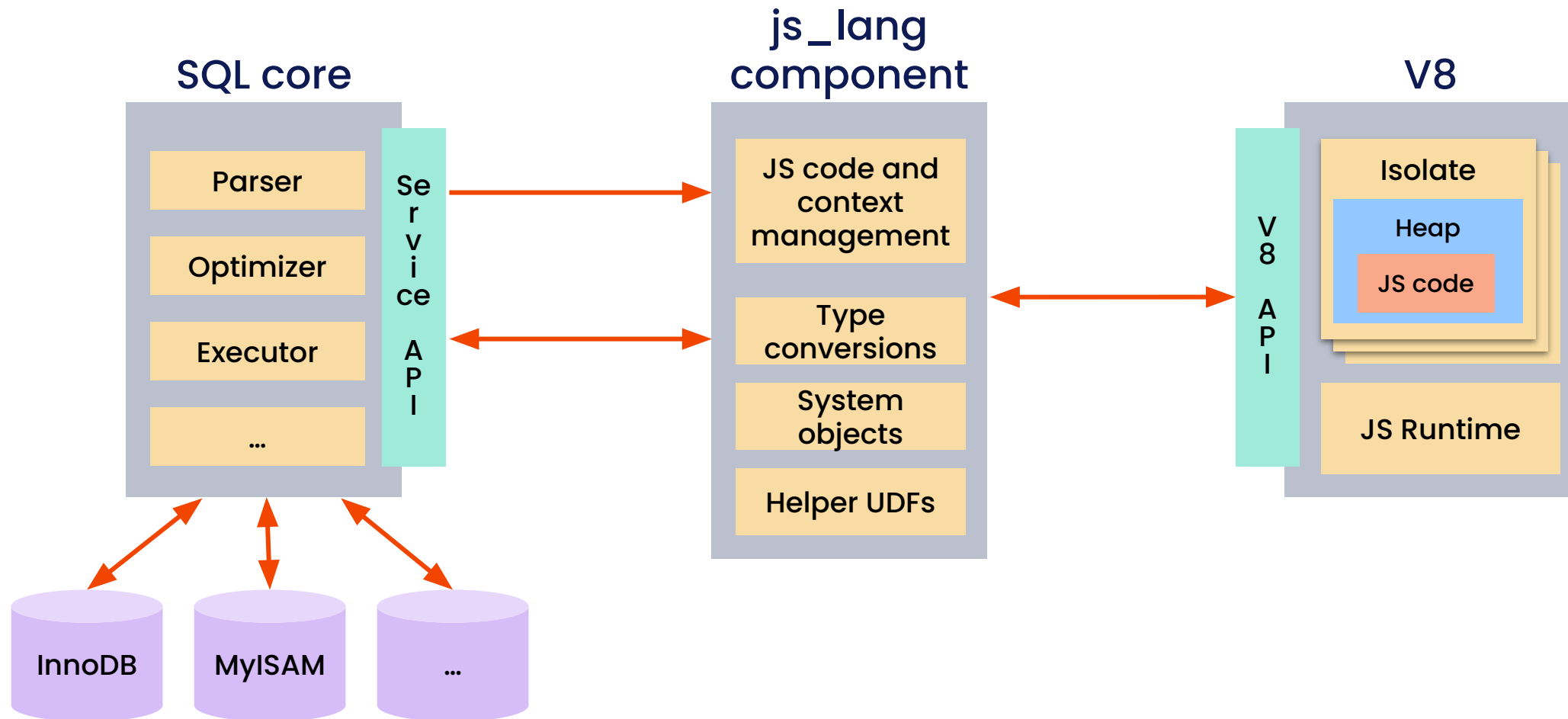 - Work-in-progress for Percona Server for MySQL

Why Stored Programs in JS?

- Modern language with many features

- Language that developers already know
  - Was used by 62% of developers in 2024 according to SO

- Great possibilities for code reuse
  - The 2nd most popular language on GitHub in 2024
  - Millions packages in NPM registry

- Ecosystem of tools for developers

- Performance
  - 100x better than SQL program in our simple benchmark

PERCONA

Architecture of JS support in Percona Server

- Implemented as a component
  - Don't use – don't enable
  - Provides external program execution service to SQL core
  - Uses services from SQL core
    - Parameters handling
    - SQL execution
    - …

- Uses Google V8 for running JS code
  - Used by Chrome, Node.JS, PLv8,…

PERCONA

Architecture of JS support in Percona Server

SQL core

- Parser
- Optimizer
- Executor
- ...

Service API

js_lang component

- JS code and context management
- Type conversions
- System objects
- Helper UDFs

V8 API

V8

- Isolate
  - Heap
    - JS code
- JS Runtime

InnoDB   MyISAM   ...

PERCONA

Feature overview

- Support FUNCTIONs and PROCEDUREs (but not EVENTs or TRIGGERs)
- IN, OUT and INOUT parameters mapped to JS variables
- Almost all SQL types as parameters/return values
  - All character sets (converted to/from UTF8mb4)
  - BLOBs are converted to/from ArrayBuffer objects

- Same level of JS language support as provided by V8
  - Standard operators, data-types, objects (e.g. Math), functions
  - No browser-specific things like DOM

- "Trusted" language
  - No direct network or file access (like in Node.JS)

PERCONA

Example

```
mysql> CREATE FUNCTION fact(n INT) RETURNS INT LANGUAGE JS AS $$
    $>   let result = 1;
    $>   while (n > 1) {
    $>     result *= n;
    $>      n--;
    $>   }
    $>   return result;
    $> $$
    -> ;

mysql> SELECT fact(5);
+---------+
| fact(5) |
+---------+
|     120 |
+---------+
```

Current project status

- Public tech preview

- Separate branch of Percona Server 8.4 (LTS)

- Code available on GitHub
  - https://github.com/percona/percona-server/tree/js-lang

- No binaries yet
  - But they are on our short term plan (end of Q1 2025)
  - Possible to build yourself
  - Build instructions in component's README.md

PERCONA

What is available and what is planned?

Available already:

- Stored program creation and execution
- Type conversions for parameters and return values
- Advanced error reporting
- Support for KILLability and timeouts

Near future (works in internal prototype):

- Console support
- Memory limits
- SQL execution

PERCONA

Differences: Percona vs Upstream

- Availability/license
  - Free and Open-source vs Enterprise or Cloud-only

- Versions
  - Branch of 8.4 vs 9.x versions.

- Full compatibility is not a goal, we aim for feature parity
  - Syntax (language, privilege name, option names)
  - Parameter conversion rules
  - Non-UTF8 charset support for parameters
  - Auxiliary UDFs
  - JS context per user/connection vs per call
  - Catching up!

- Status
  - Public preview vs GA

Challenges

Related to Upstream:

- Missing/limited Services in SQL core
- Bugs in service implementation
- We have to extend SQL core and fix bugs

Related to V8:

- Primary user project is Chrome
  - Fast-changing API
  - Things get broken often
  - Orientation towards multi-process model

How you can help?

We want your feedback!

- Especially if you are JS developer!

- Please try it out and:
  - Report bugs
  - Request usability improvements
  - Ask for missing features
  - Suggest documentation improvements

- It is easier to change things now at public preview stage

- PRs on GitHub are also welcome!

Thank You!

PERCONA

Databases run better with Percona