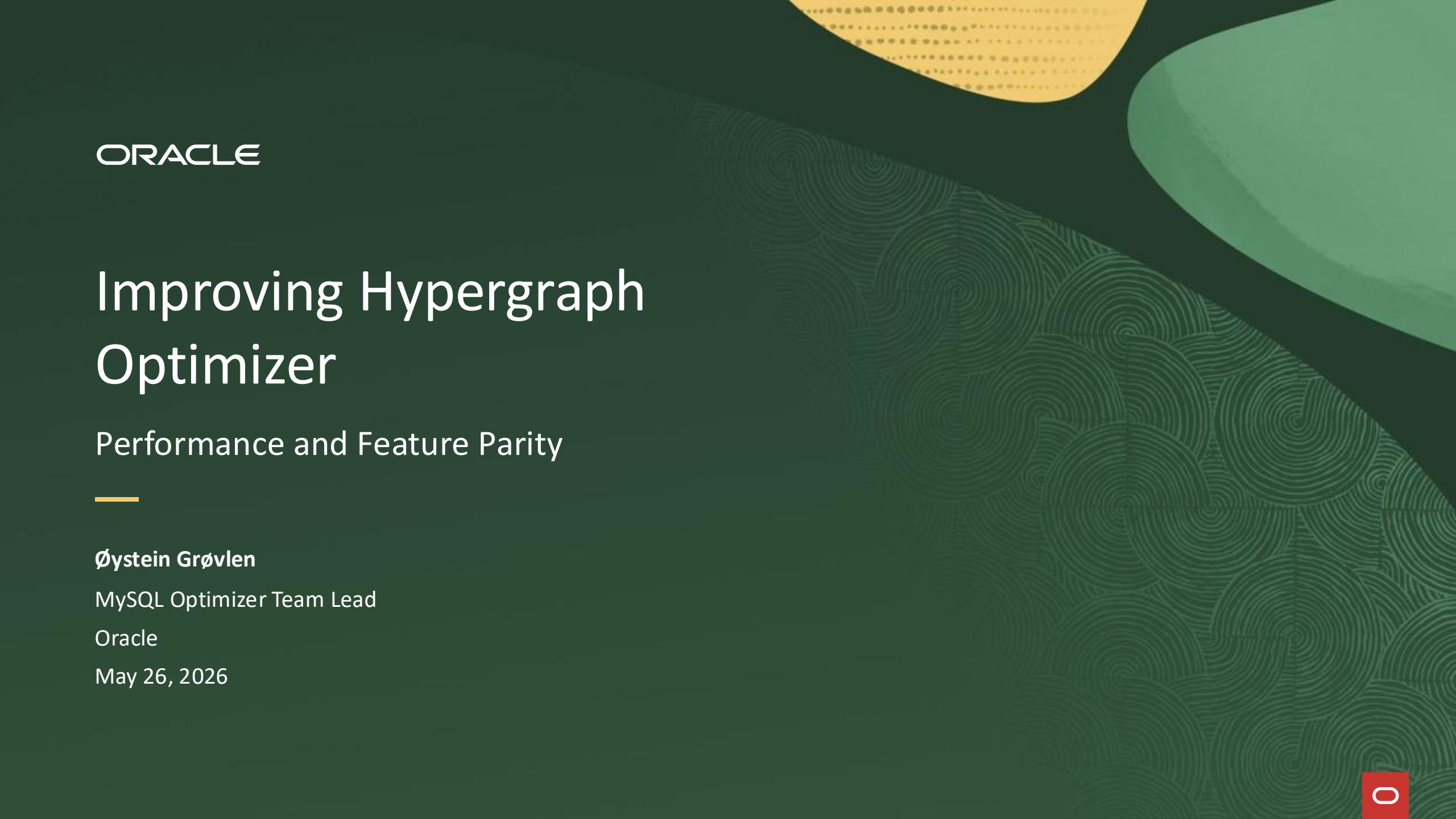




ORACLE

Improving Hypergraph Optimizer

Performance and Feature Parity

Øystein Grøvlen

MySQL Optimizer Team Lead

Oracle

May 26, 2026



Øystein Grøvlen

MySQL Optimizer Team Lead, Oracle

Safe harbor statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, timing, and pricing of any features or functionality described for Oracle's products may change and remains at the sole discretion of Oracle Corporation.

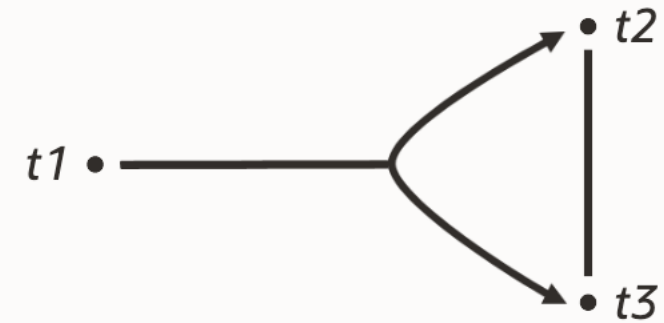
Why a new Optimizer?

Issues with the classic query optimizer

- Partially based on heuristics instead of cost
- Inefficient join ordering algorithm
 - Tries all combinations
 - No caching of costs
- Left-deep plans only
- No tracking of interesting orders
 - Explicit sort vs. reading an index in sorted order?
 - Especially important for ORDER BY ... LIMIT queries
- Prefers indexed nested-loop join over hash join

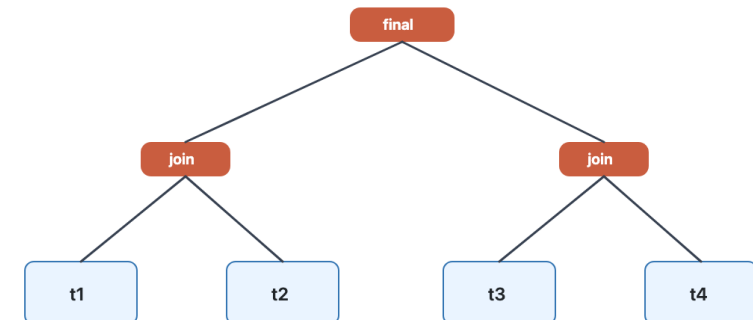
Hypergraph Optimizer

- Fully cost-based
 - Costs are calibrated from microbenchmarks
- Efficient
 - Bottom-up planner using dynamic programming
 - Only considers relevant join orders
- Tracks interesting orders
 - May do sorting at any stage
- Supports bushy join plans
- Cost-based choice between nested-loop join and hash join
- Based on research:
 - DPhyp: [“Dynamic Programming Strikes Back” by Guido Moerkotte and Thomas Neumann](#), SIGMOD 2008
 - Used in several DB system (e.g., DuckDB)



Hypergraph representation of *t1* LEFT JOIN (*t2* JOIN *t3*)

Bushy Join

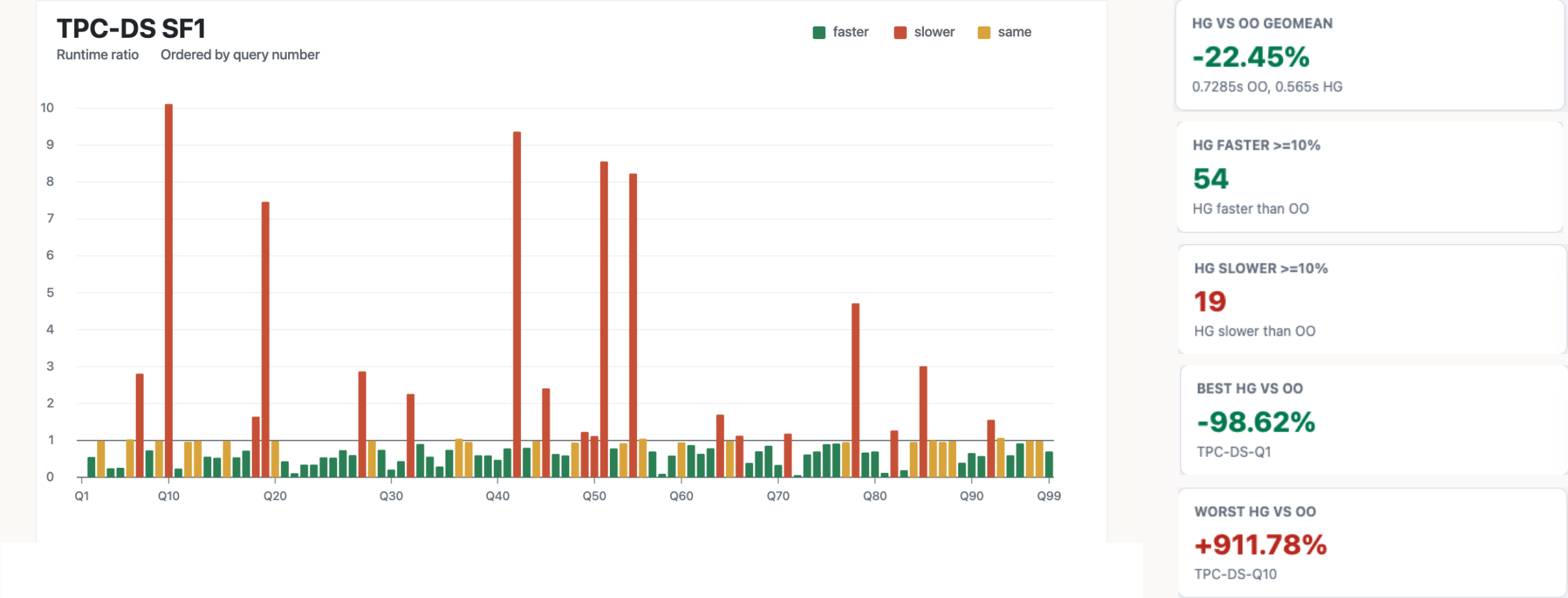


Hypergraph Optimizer in MySQL

- Used by MySQL Heatwave since 8.0.31
 - Massively parallel column store for OLAP
 - Logical optimization using Hypergraph Optimizer
 - Cost model provided by Heatwave storage engine
 - Physical optimization performed by Heatwave
 - No indexes, only scans
 - Hash join is only join method
- Available in cloud service since 9.0
- Available in MySQL Community Edition since 9.7.0
 - **SET optimizer_switch='hypergraph_optimizer=on';**
- Still evolving

Performance Comparison

TPC-DS Queries, Scale Factor 1



Future Work

Bug#120542

Model deficiencies

- Improved join cardinality estimation (WL#17047)
- Improved aggregate estimation
- Improved selectivity estimates for base tables
- Better integration with InnoDB index statistics
 - Especially, multi-column indexes
- Heuristic graph simplification (WL#17051)

Missing functionality

- Missing optimization
 - COUNT(*) of entire table
 - Min/max for index scans
 - Dynamic range access
- Cost model improvements
 - User configurable costs (WL#16111)
 - Sorting large data (WL#16551)
 - Index merge (WL#16657)
 - Set operations (UNION, INTERSECT, ...) (WL#16549)
 - Window functions (WL#16550)
 - ...
- Optimizer hints for hash join

How can you contribute?

- Test it on your applications/queries
 - Tell us about wins and regressions
- File bug reports
- Contribute bug fixes
- Extend functionality
- Develop tools
 - Visualization of planning process and query plans
- Documentation



Thank you

ORACLE