This is Going to Sound Crazy, But What If We Used Large Language Models to **Boost** Database Tuners By Leveraging Prior History? We Will Find Better Configurations More Quickly Than Retraining From Scratch!

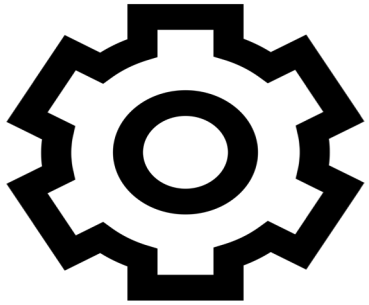
William Zhang, Wan Shen Lim, Andrew Pavlo

wz2@cs.cmu.edu

(work under submission)

Carnegie Mellon
Database Group

Database Tuning is Hard

System Knobs	Physical Design	Query Options
		
Buffer Pool	Indexes	Plan Hints
Parallelism	Per-Index Knobs	Query Knobs
…	…	…

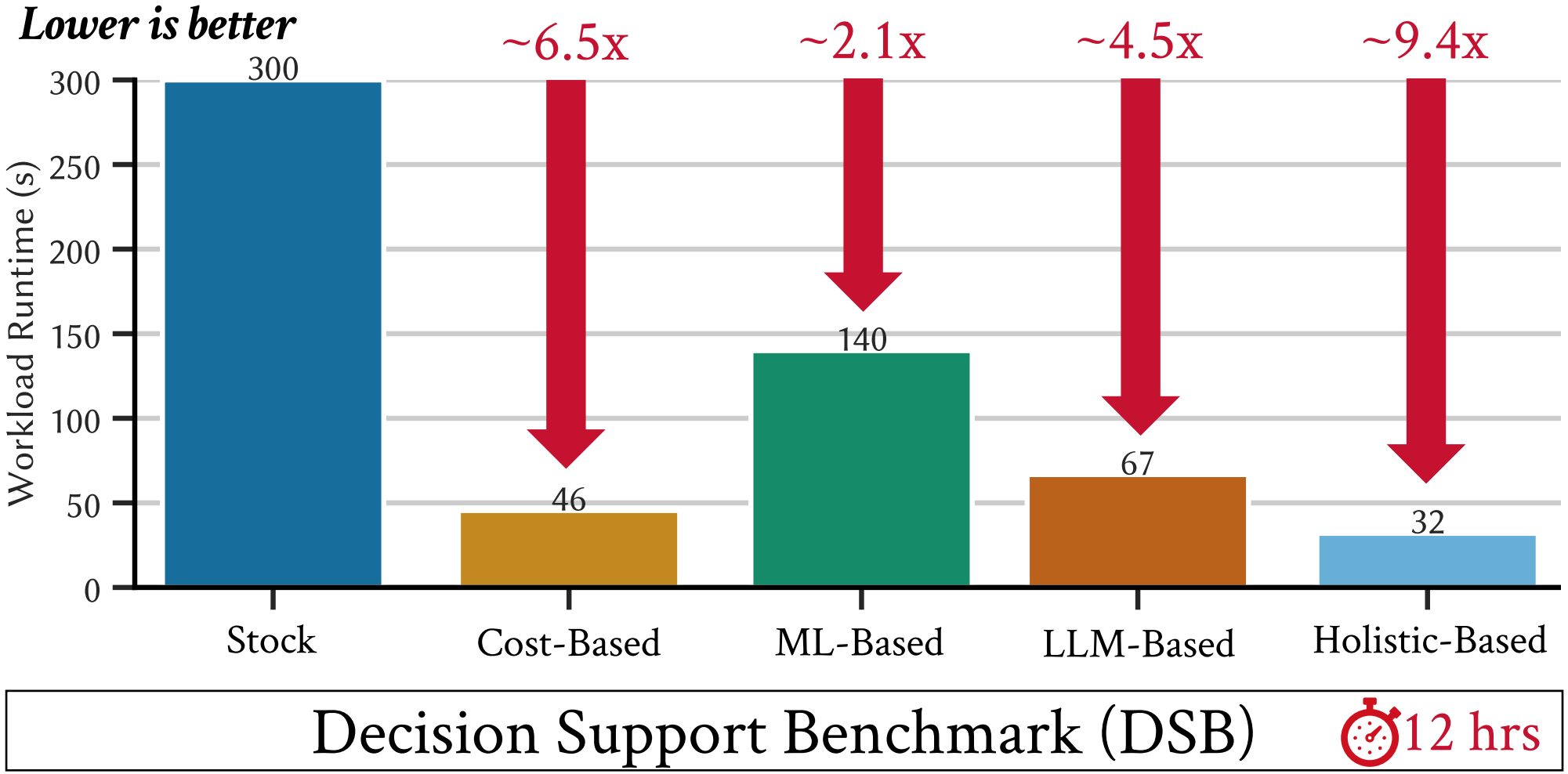
Trillions of possibilities with subtle interactions!

Proliferation of Tuners

Type	Tuner	System Knobs	Physical Design	Query Options
Cost	PGTune (‘14)	✓		
Cost	Dexter (‘17)		✓	
Cost	DTA (‘20)		✓	
Cost	Auto-Steer (‘23)			✓
ML	UniTune (‘23)	✓	✓	✓
Holistic	Proto-X (‘24)	✓	✓	✓
LLM	λ-Tune (‘25)	✓	✓	

The Holon Approach... (Proto-X)
VLDB 2024. William Zhang et al.

Tuners are Effective



But…Environments Drift

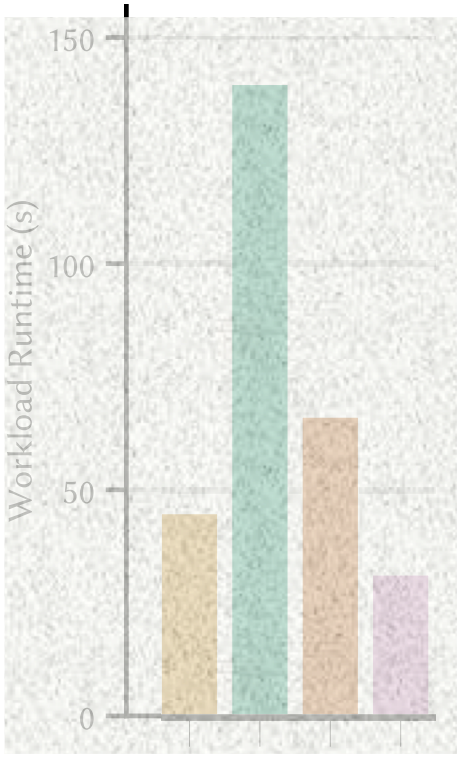
- Diverse types of environment changes
  - Parameter Drift, Query Template Drift, Volume
  - Hardware, Dataset Growth
  - Cross-Deployment

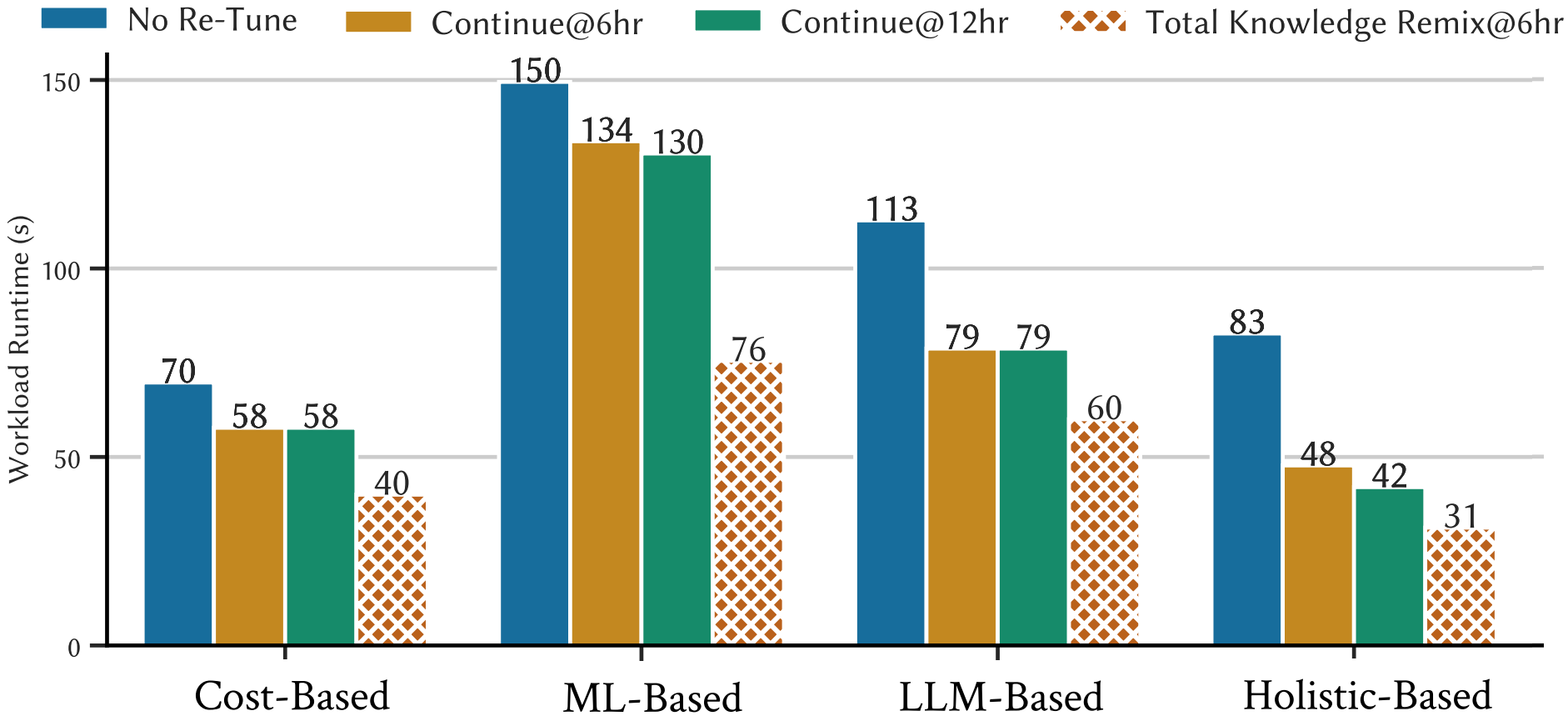
- **Issue #1**: Tuners are rigid and brittle
  - Unable to select and utilize relevant historical artifacts

- **Issue #2**: Workload-Level Granularity
  - Ignores query-level semantics

Limited Tuner Adaptivity

Lower is better





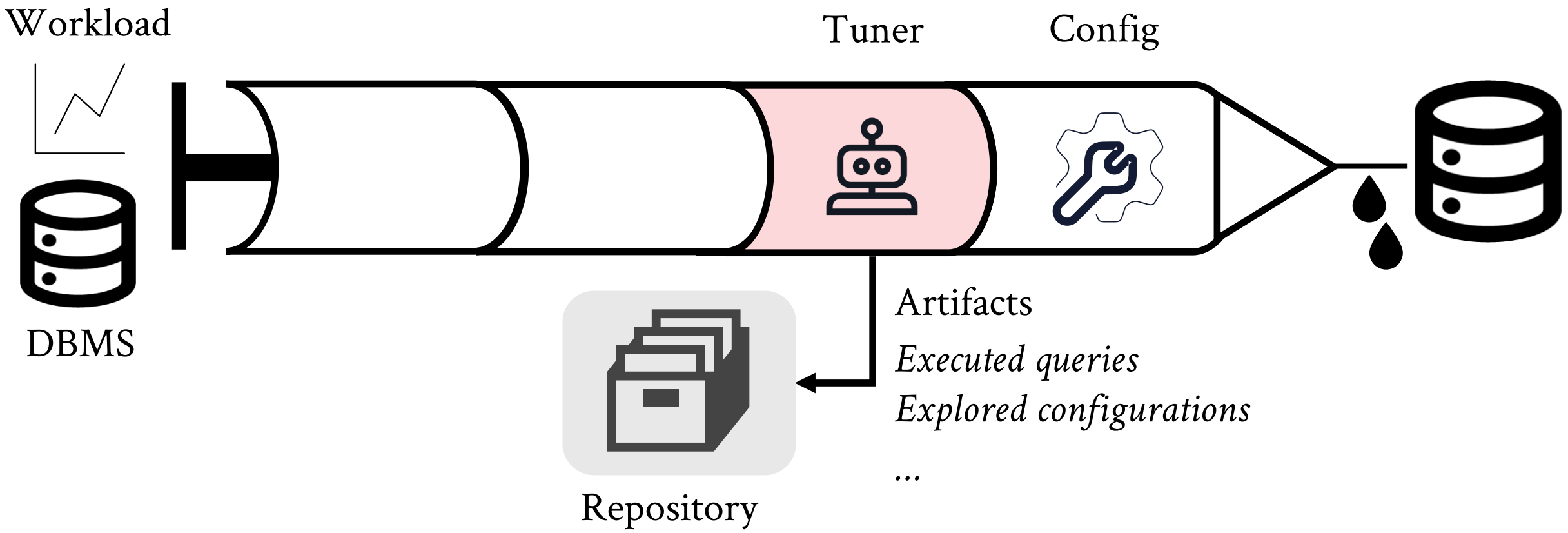
DSB

Drift

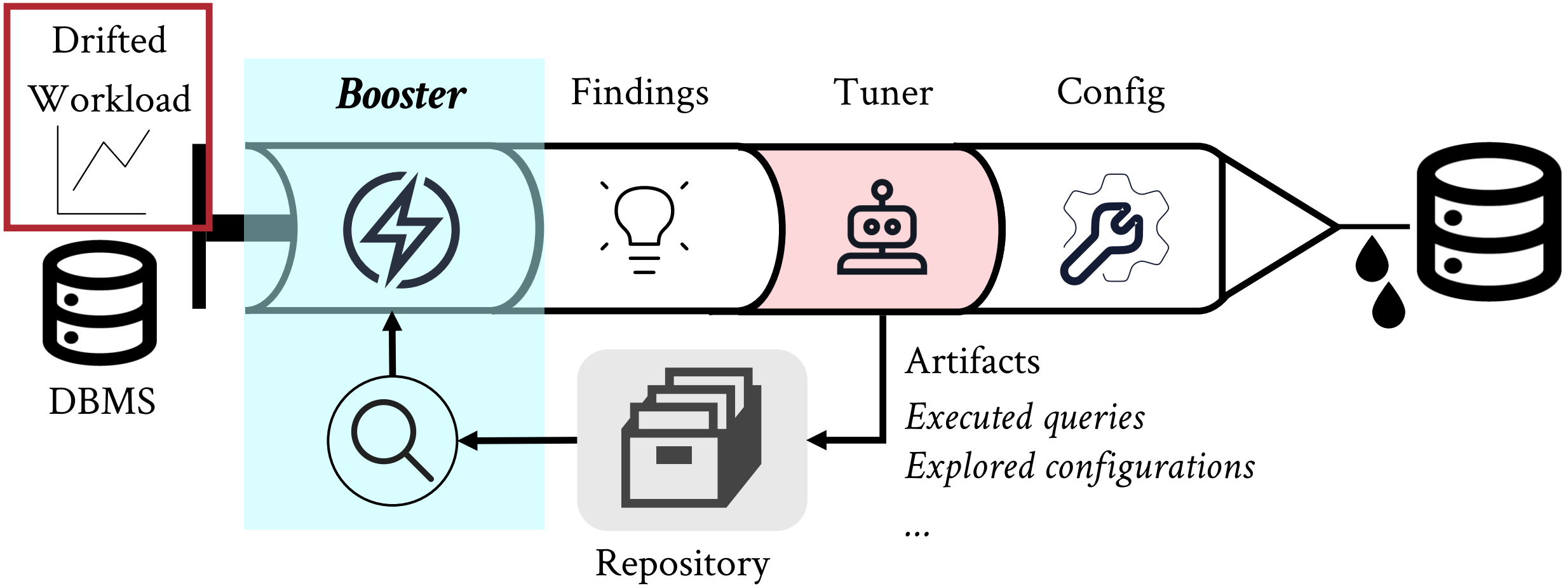
50% Parameter Drifted DSB

Booster

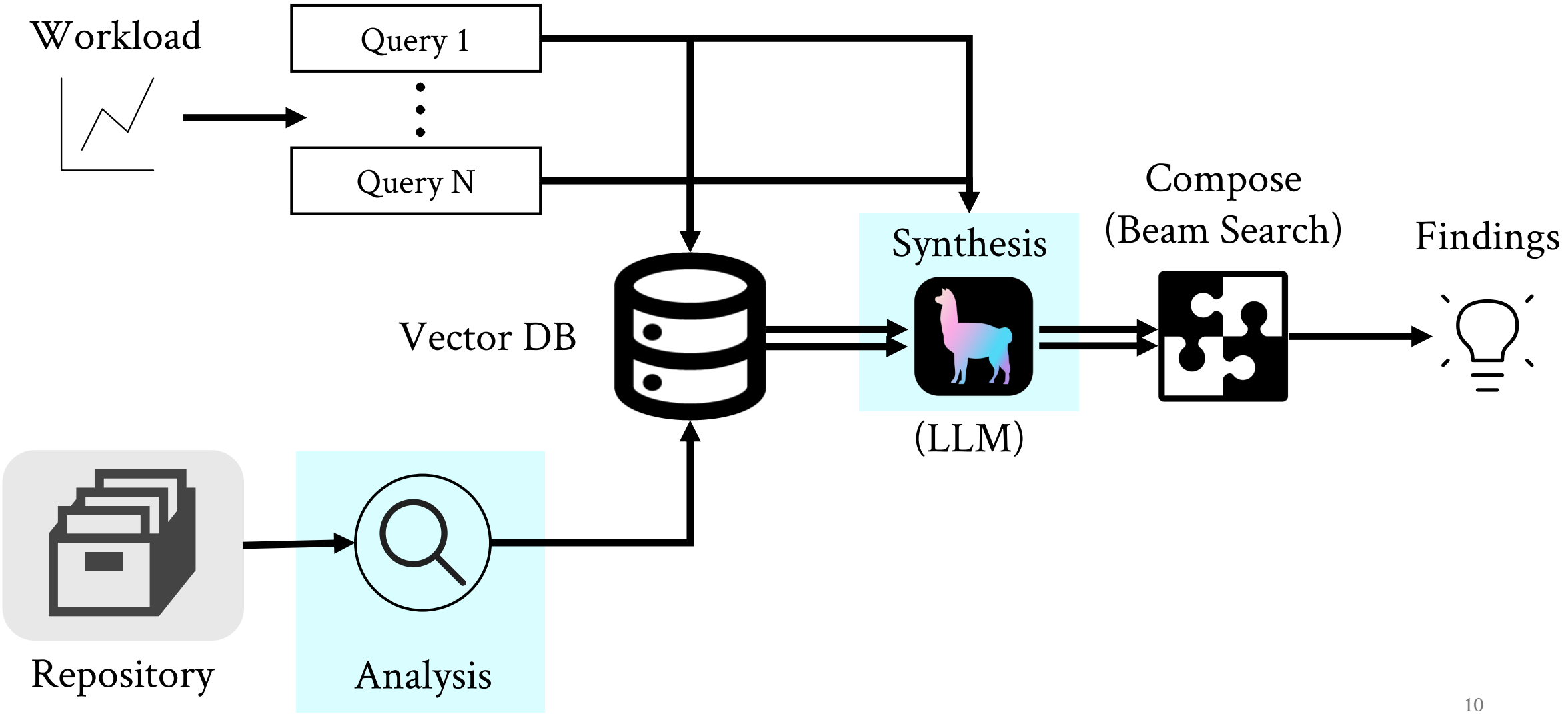
Booster Overview



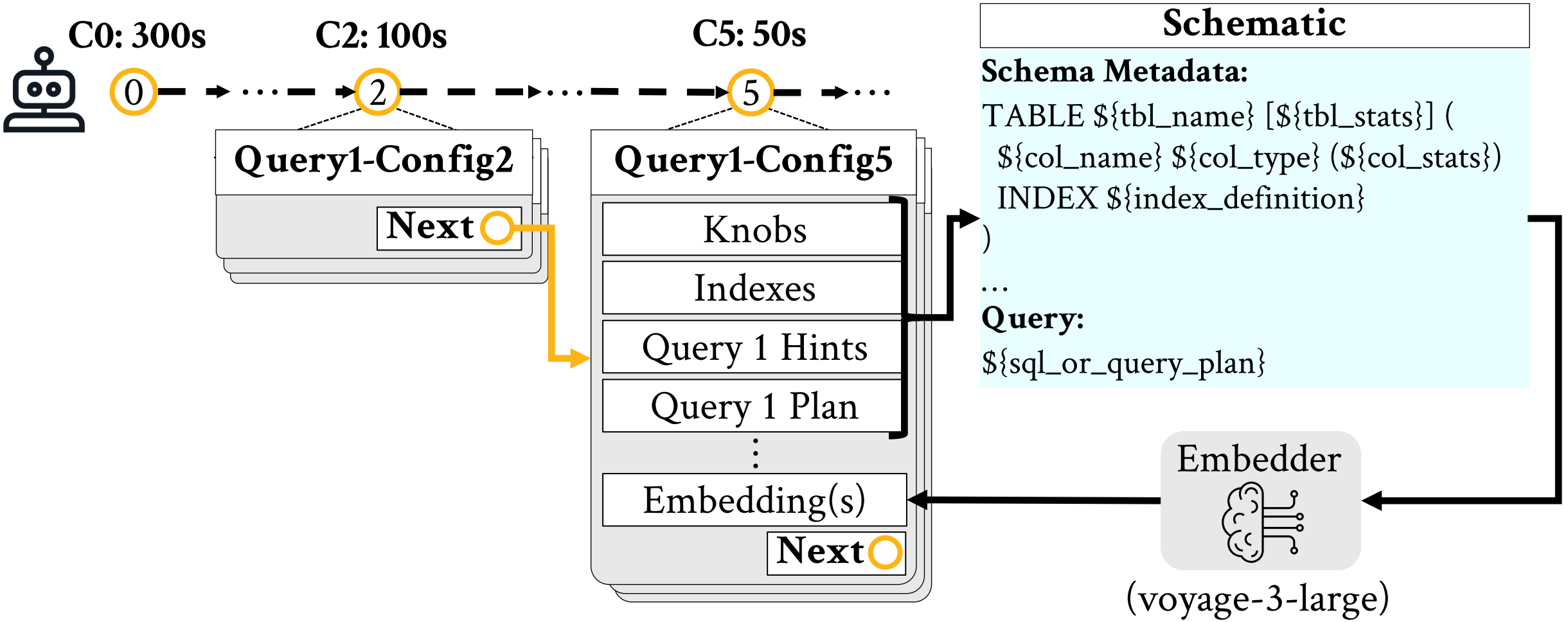
Booster Overview



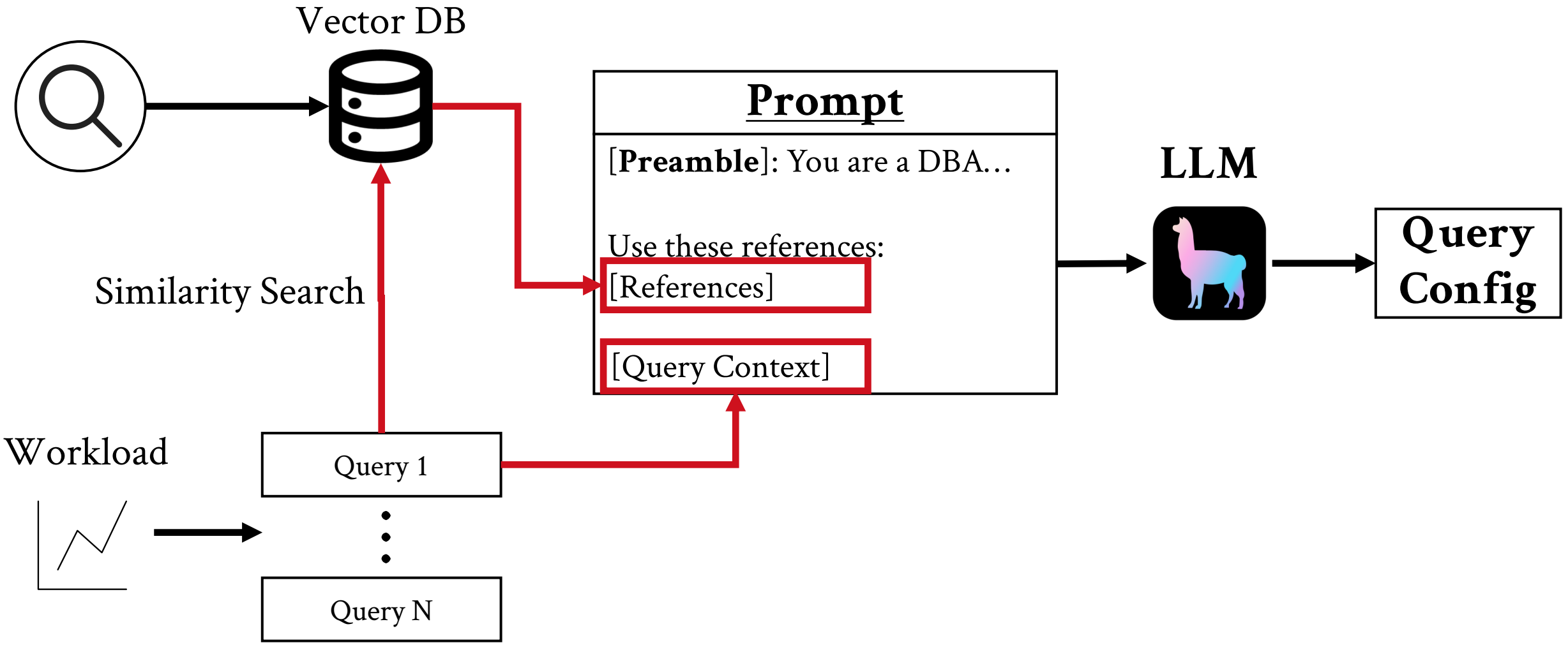
Booster Flow



Analysis: Trajectory & Schematics



Synthesis: Query-Level



Experiments

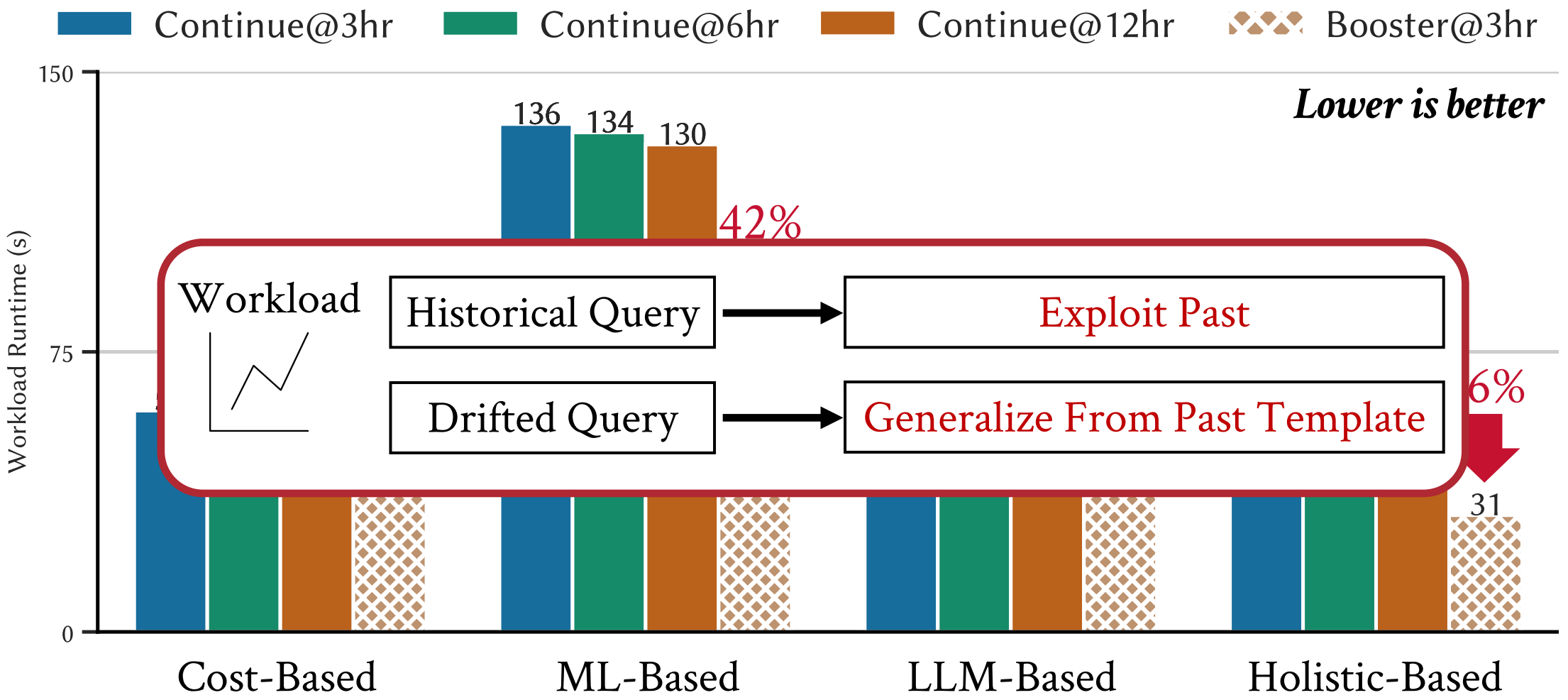
Experiment Setup

- Hardware:
  - Ubuntu 22.04.
  - 2x20 core Intel Xeon Gold 5218R, Samsung SM883 SSD. ~192GB RAM
- PostgreSQL v15
  - HypoPG for hypothetical indexes
  - pg_hint_plan for query tuning
  - Additional extensions/patches to facilitate observability
- Workload: Microsoft's Decision Support Benchmark (DSB)

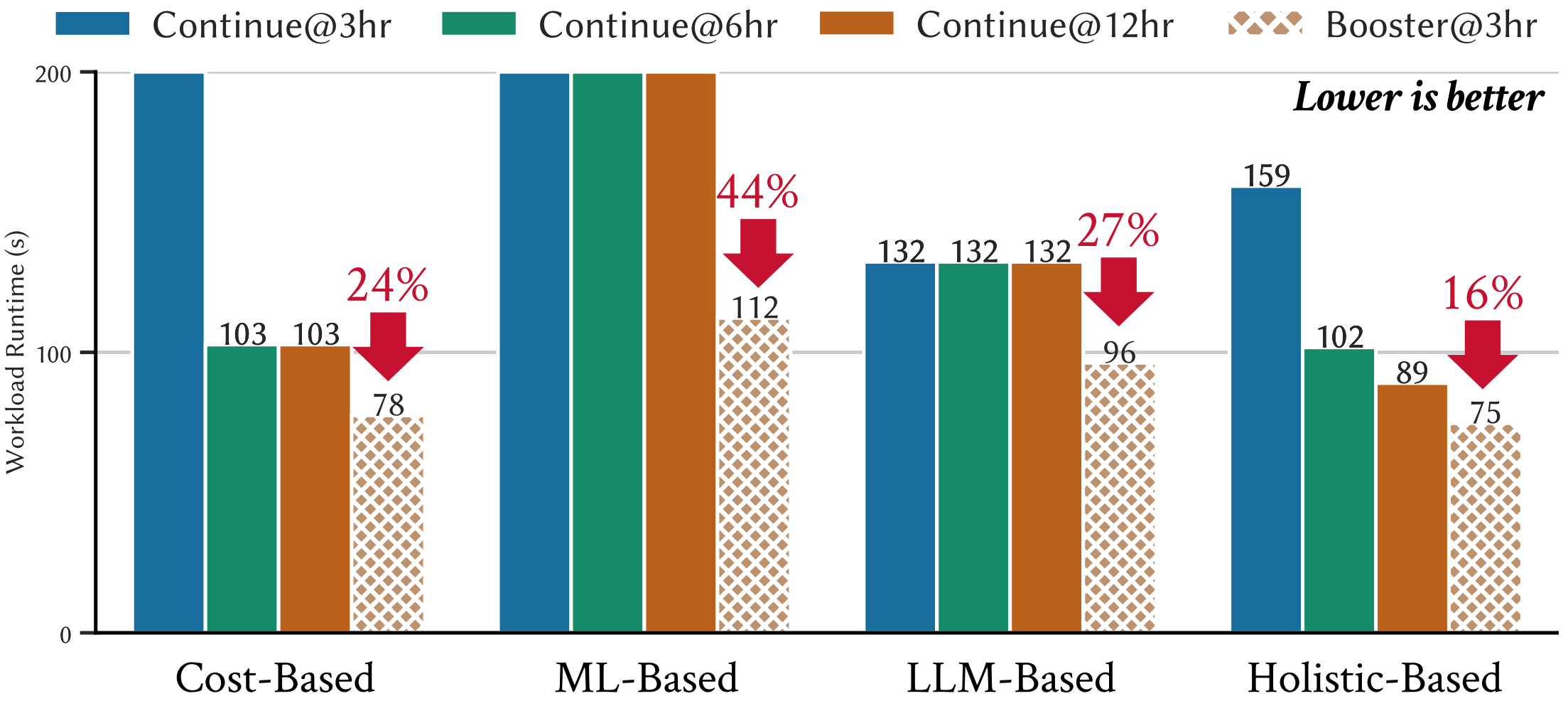
General Tuner Acceleration

- **Tuners**:
  - Cost-Based:
    - PGTune (P): heuristic knob tuner
    - DTA: cost-based index tuner [VLDB20]
    - Auto-Steer (AS): query knob tuner [VLDB23]
  - ML-Based: UniTune [SIGMOD23]
  - Holistic-Based: Proto-X [VLDB24]
  - LLM-Based: λ-Tune [SIGMOD25] + Auto-Steer
- **Initial Artifacts**: Allow each tuner to optimize same DSB for 12 hours
- ***Booster*** standardizes artifacts to match tunables supported by Proto-X

50% Parameter Drift



50% Template Drift



Takeaways and Thanks!

- Automated tuning tools are effective for optimizing a fixed deployment and workload. However, these drift and evolve over time. ***Booster*** assists tuners in finding *more performant* configurations *in less time.*

- **Booster** generalizes from *similar* historical queries by exploiting query-level semantics from historical artifacts through LLM-based synthesis. **Booster** then uses beam search to break out of local suboptima and compose promising configurations.

wz2@cs.cmu.edu