Automatic Capture of Minimal, Portable, and Executable Bug Repros Using AMPERe

Lyublena Antova (Greenplum) Konstantinos Krikellas (Google) Florian Waas (Greenplum)

DBTest May 21, 2012





Query Optimization

- Key to performance
 - Possibly multiple orders of magnitude improvement
- Key to extensibility
 - Need optimizer support for new features/elements
- Key to customer fidelity
 - Nothing makes customers happier than a working system with generates optimal results





Investigate Customer Bugs: Example



Investigating Optimizer Bugs: Challenges

- Query optimizers are complex
 - and therefore bug-prone
- Repros for optimizer bugs hard to obtain
 - schema definition, statistics, offending query, logs, configuration parameters, …
- Collection is done *after* the fact
 - may be too late in concurrent environments where data changes constantly
- Collection and investigation require significant manual effort
- Long turnaround times for fixing optimizer bugs





In an ideal world...

- Bug repros are automatically generated when a bug is hit
 - no need for manual collection
- Bug repros are executable
 - live repro vs. eyeballing static stack traces
- Bug repros are portable
 - problem can be debugged in-house without access to customer cluster
- Bug repros are minimal
 - no need to dump the whole database schema if query only touches a single relation
- Bug repros can be added to test framework after the bug is fixed
 - make sure it doesn't happen again



EMC²

AMPERe



ORCA: Greenplum's Query Optimizer

- Extensibility
- Modularity
- Multi-core/many-core enabled
- Verifiability
- AMPERe is built into ORCA



EMC²

ORCA: Architecture





EMC²







AMPERe: Capture Mechanism

- Based on DXL abstraction
 - abstraction of algebrized query, metadata, statistics
- Capture mechanism
 - pre-serialize incoming query, parameters, and accessed metadata objects and statistics
 - register objects with capture mechanism
 - at crash/exception time, harvest registered objects and serialize them to a memorymapped output buffer





AMPERe Dump: Example







AMPERe: Loading

- Load AMPERe dumps
 - load query from dump file
 - file-based metadata provider on top of the dump file







Use cases

- Customer support cases
 - crashes and run-time exceptions
 - bad plan choices
 - mining support cases
 - integration with EMC DialHome
- QA
 - AMPERe-based test framework
 - cost model sensitivity and calibration





Operational Characteristics



AMPERe file size in KB for TPC-H queries





Summary

- Investigating optimizer bugs is hard
- AMPERe
 - automatic
 - minimal
 - portable
 - executable
 - extensible
- Use cases
 - customer support cases
 - QA





Thanks!





AMPERe Capture Mechanism: Workflow





