Intelligent Query Processing

A brief 20 minute preview

Where are we going from here?

Intelligent Query Processing

- Overall emphasis on making the query engine smarter and better performing
- “Just upgrade and it goes faster” in most cases.
- Builds on current work:
    - Batch Mode Memory Grant Feedback
    - Automatic Plan Regression Correction
    - Interleaved Execution
  - One example of future work:
    - Batch Mode on Rowstore

Intelligent QP

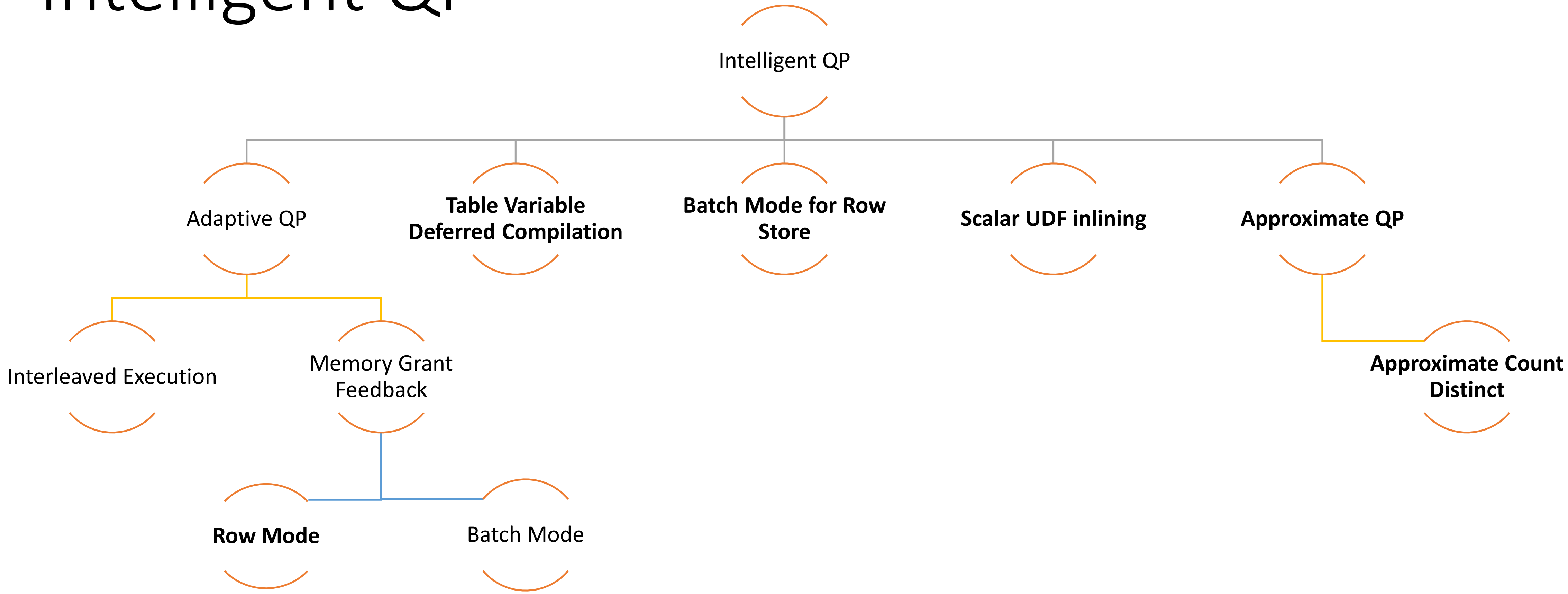

Table Variable Deferred Compilation

Adjusted cardinality estimates for table variables

- In SQL Server 2017, Interleaved Execution for Multi-Statement Table Valued functions corrected the fixed guess issue for SELECT statements

- Table variables also use a fixed CE guess of “1”
- With *Table Variable Deferred Compilation*, we’ll use the actual cardinality instead of the guess to improve query performance for larger result sets

Inlining Scalar UDFs

- Challenge
  - Scalar UDFs can cause optimization issues
  - No visibility to the internals of the UDF
- Solution
  - Inline the content of a scalar UDF pre optimization
  - Gives the optimizer visibility into the full context of the UDF
  - Results in much better overall plans
  - Maintains the modularity benefits of the UDF

Memory Grant Feedback

For row mode

- SQL Server 2017 introduced Batch Mode Memory Grant feedback

- Next step is to extend this to row-mode
  - Updating the cached plan for:
    - Row-mode spills to disk
    - Row-mode excessive memory grant waste

- Consecutive executions benefit from the adjusted grant size

Batch Mode for Row Store

HTAP Without Column Stores

- How to do HTAP where columnstores are not appropriate
    - Very high churn
    - High rate of singleton access
  - Batch Mode Processing
    - Used in columnstore queries
    - Does aggregations and calculations in batches of thousands of rows instead of one at a time.
    - Much more efficient for processing analytic queries.

Batch Mode for Row Store

- New Feature in development
- Currently does aggregations in batches.
- Will also retrieve rows in batches for more efficient reads.
- Work in the Optimizer to introduce heuristics to avoid lengthy compiles for trivial plans

Demo: Batch Mode in Rowstore

Next steps

- We’ll announce when the public preview is available for testing these new Intelligent QP features in the next version of SQL Server (via CTP) and Azure SQL Database
- We’ll provide more implementation details upon availability
- If you have *real-world workloads* that you think will benefit from any of the new features and are willing to help with early-build testing, please reach out to joe.sack@Microsoft.com and Kevin.Farlee@microsoft.com so we can notify you of future testing opportunities.