SQL Server 2022 Programmability & Performance

Gianluca Hotz

ghotz@ugiss.org | @glhotz

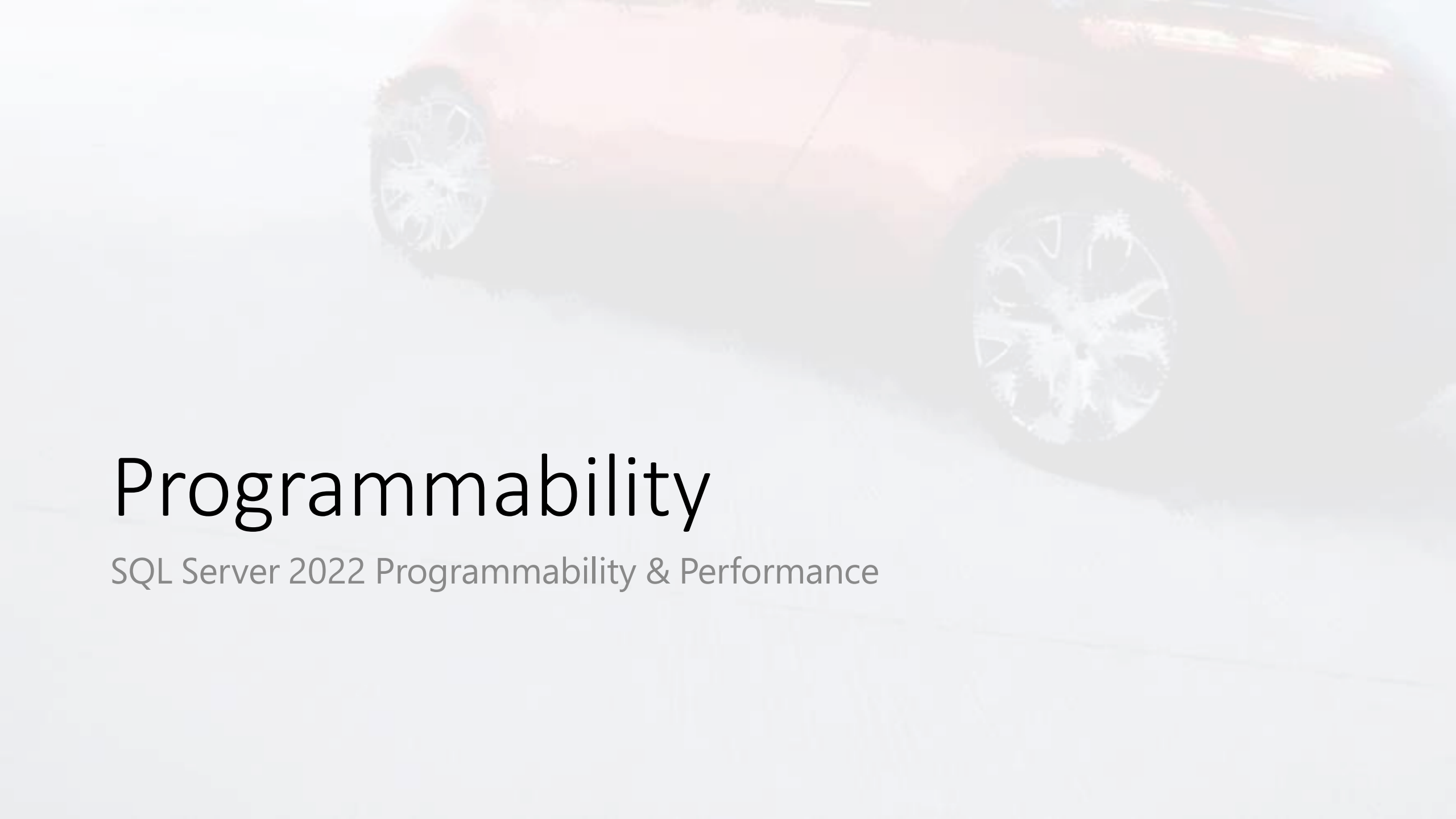
Sponsors & Organizers

Who am I?

- Gianluca Hotz | @glhotz | ghotz@ugiss.org
- Independent Consultant
  - 25+ years on SQL Server (from 4.21 back in 1996)
  - Database modeling & development, sizing & administration, modernization (upgrades & migrations), performance tuning, security
- Community
  - 24 years Microsoft MVP SQL Server/Data Platform (from 1998)
  - VMware Experts SQL Server
  - Founder and president UGISS (ex «PASS Chapter»)

Agenda

- Programmability
- Performance



Programmability

SQL Server 2022 Programmability & Performance

Programmability enhancements

T-SQL	JSON	Time Series
• GREATEST • LEAST • STRING_SPLIT • TRIM functions • DATETRUNC • IS [NOT] DISTINCT FROM • WINDOW clause • LEFT_SHIFT • RIGHT_SHIFT • BIT_COUNT • GET_BIT • SET_BIT	• ISJSON • JSON_PATH_EXISTS • JSON_OBJECT • JSON_ARRAY	• DATE_BUCKET • GENERATE_SERIES • FIRST_VALUE • LAST_VALUE

XML Compression

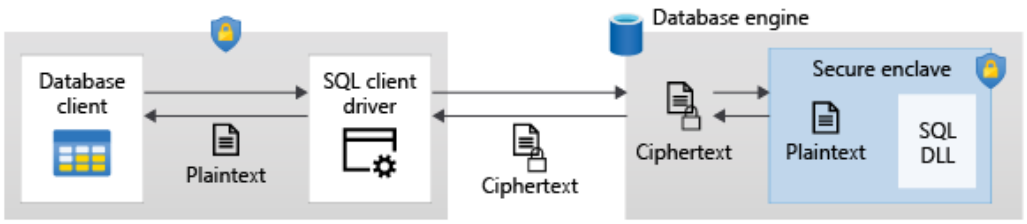
- Finally!!! ☺
- New **CREATE/ALTER TABLE** or **CREATE/ALTER INDEX** option
  - **`XML_COMPRESSION = ON|OFF [ON PARTITIONS(…)]`**
- More info
  - https://learn.microsoft.com/sql/t-sql/statements/create-index-transact-sql#xml_compression

Multi-write replication

- Last Writer Wins (LWW) in Peer-to-peer transactional replication
  - Introduced in SQL Server 2019 CU13
  - Previously manual resolution in case of conflict and replication paused
- Enterprise Edition only
- More info
  - https://learn.microsoft.com/sql/relational-databases/replication/transactional/peer-to-peer-conflict-detection-in-peer-to-peer-replication#automatically-handle-conflicts-with-last-write-wins
  - https://techcommunity.microsoft.com/t5/sql-server-blog/replication-enhancements-in-the-sql-server-2019-cu13-release/ba-p/2814727

Always Encrypted with Secure Enclaves

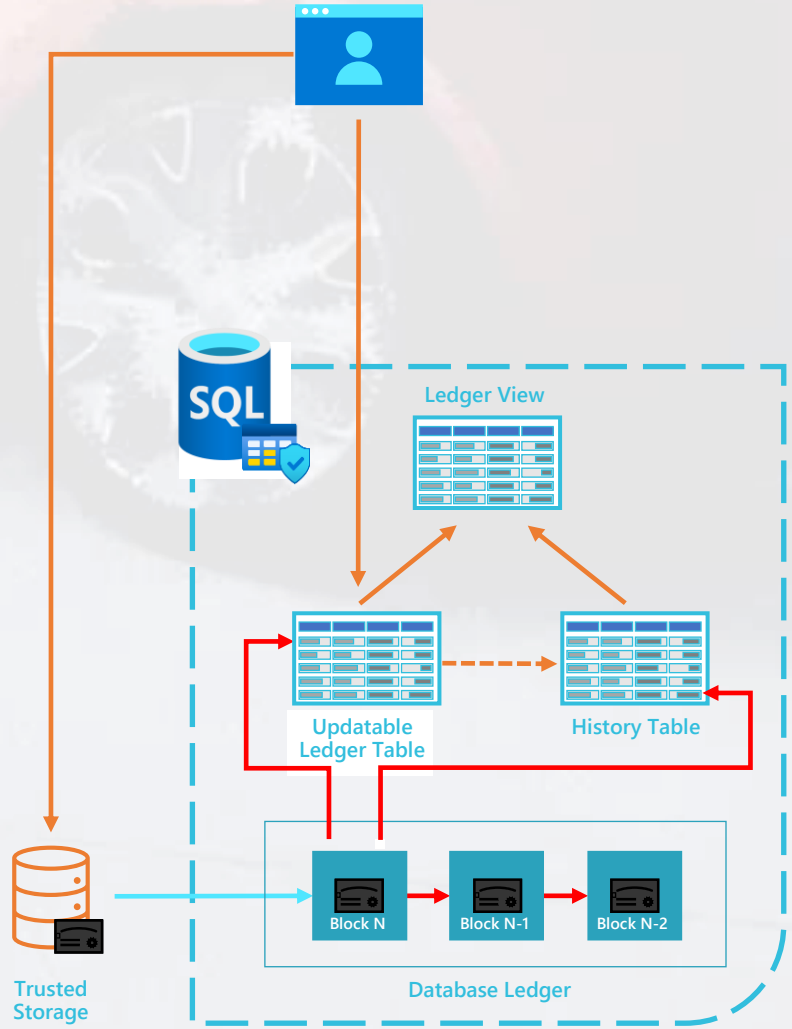
- Enhancements inside enclave
  - Multiple threads and key caching
  - More operations supported
- More info
  - https://learn.microsoft.com/sql/relational-databases/security/encryption/always-encrypted-enclaves



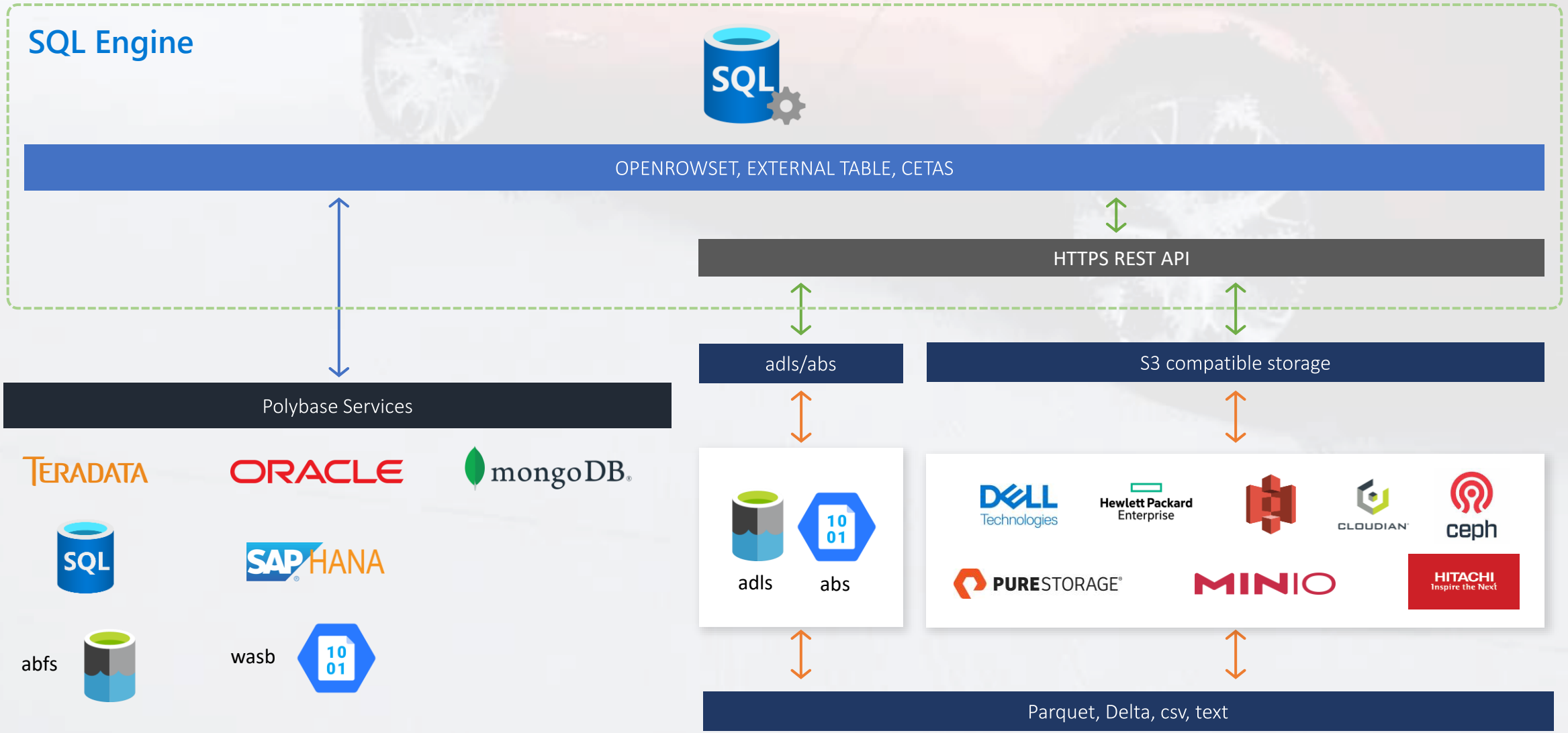
Operation/ Support	Azure SQL Database	SQL Server 2022	SQL Server 2019
Comparison operators	Yes	Yes	Yes
BETWEEN	Yes	Yes	Yes
IN	Yes	Yes	Yes
LIKE	Yes	Yes	Yes
DISTINCT	Yes	Yes	Yes
Joins	Yes	Yes	Only nested loops
ORDER BY	Yes	Yes	No
GROUP BY	Yes	Yes	No

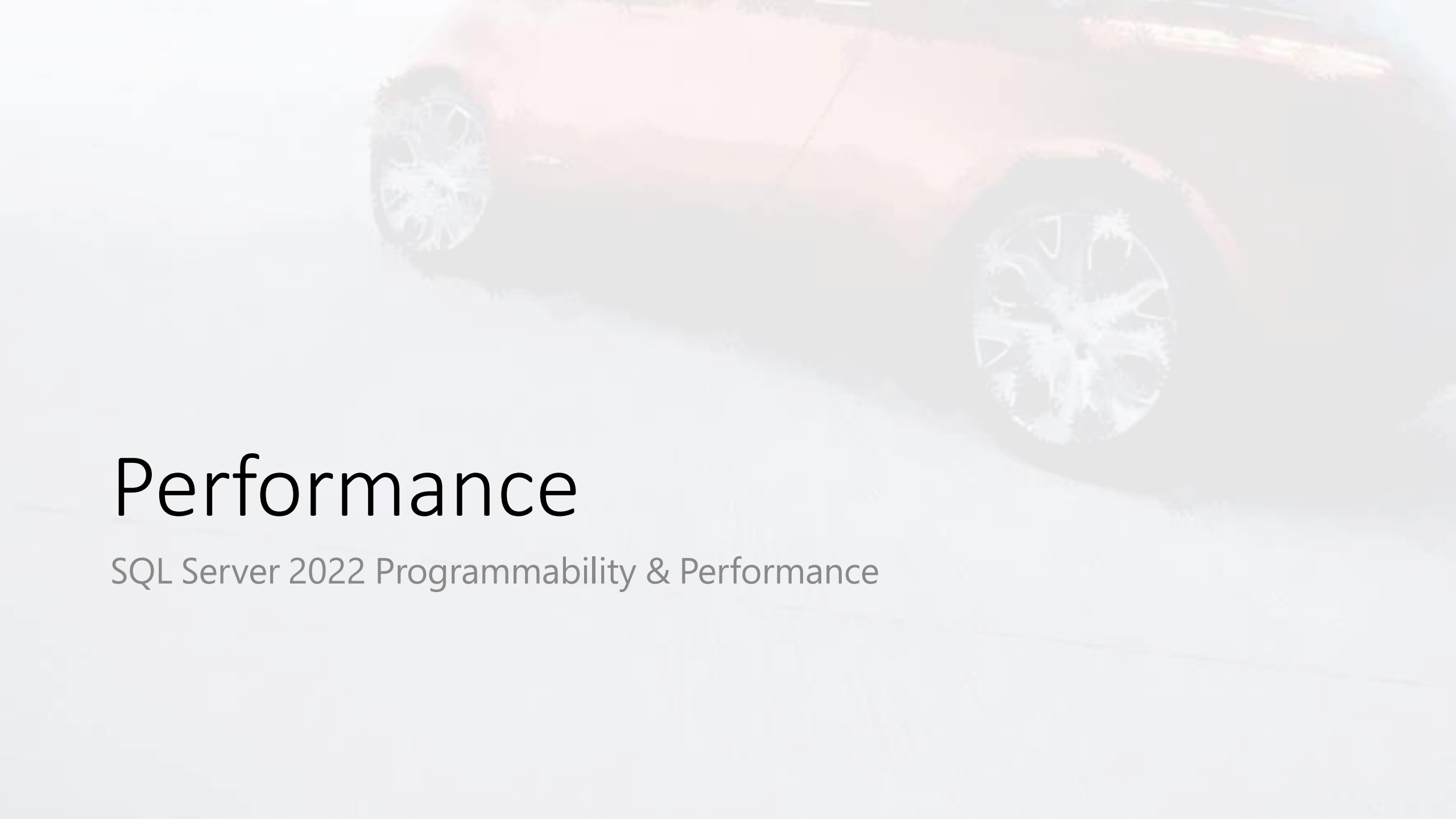
Ledger Tables

- **Updatable** allow insert/update/delete
- History of updated/deleted rows preserved in history table and easy-to-query Ledger View
- Integrity of updatable/history tables maintained through cryptographic links of the Database Ledger
- System can periodically upload digital receipts to a customer-configured trusted storage service
- Customer can use digital receipts to verify the integrity of the data
- **Append-Only** allow only insert
  - no need for a history table



Data virtualization in SQL Server 2022





Performance

SQL Server 2022 Programmability & Performance

Buffer Pool parallel scans

- BP operations use hash tables (BUF structures to find pages)
- Some operations still needs to scan all BUF structures
  - Problem on systems with large amount of memory e.g. 1TB+
  - SQL Server warning in ERRORLOG
  - https://docs.microsoft.com/troubleshoot/sql/performance/buffer-pool-scan-runs-slowly-large-memory-machines
- SQL Server 2022 introduces buffer pool parallel scan
  - Enabled by default (available in Standard and Enterprise editions)
  - Standard Edition limited to 2 threads on 64GB
- More info
  - https://cloudblogs.microsoft.com/sqlserver/2022/07/07/improve-scalability-with-buffer-pool-parallel-scan-in-sql-server-2022
  - https://youtu.be/4GvU106Xiag

“Hands-free” tempdb

Pre SQL Server 2019

- 1 file = PFS, GAM, SGAM contention
- Add multiple files
- Trace flags 1117 and 1118
- SQL 2016 setup auto adds multiple files
- Trace flags not required by SQL Server 2016

SQL Server 2019

- PFS concurrency
- Autogrow and uniform default for tempdb
- Now system table pages become hotspot
- Tempdb metadata optimization ON
- SGAM and GAM contention remain

SQL Server 2022

- SGAM and GAM concurrency
- Latch contention gone

“Purvi’s list”

- Reduced buffer pool I/O promotions
  - Tuned read-ahead to avoid single page promotions to 8 pages I/Os
- Enhanced spinlock algorithms
  - No details, internal adjustments make spinlocks more efficient…
- Improved Virtual Log File (VLF) algorithms
  - If growth > 1/8 current size, if < 64MB creates 1 VLF instead of 4
  - https://learn.microsoft.comsql/relational-databases/sql-server-transaction-log-architecture-and-management-guide#virtual-log-files-vlfs
- **Instant file initialization (IFI) for transaction log file growth events!!!**
  - **Only growth events up to 64MB!**

Columnstore enhancements

- Ordered Clustered Columnstore Index!
  - https://docs.microsoft.com/azure/synapse-analytics/sql-data-warehouse/performance-tuning-ordered-cci
- Columnstore string enhancements
  - Deep data (e.g. char, binary, guid) min/max maintained when rebuilding
  - Fast string-equal operation
  - LIKE pushdown (RG elimination only for prefix searches i.e. str% not %str)
- Segment elimination
  - Extends to string, binary, guid data types and datetimeoffset w/ scale > 2
  - Was only numeric, date, time data types and datetimeoffset w/ scale <= 2

Batch mode enhancements

- Processor Advanced Vector Extension (AVX) 512
- Some operations faster for Columnstore and Rowstore batch mode
- Recommended for the following processors
  - Intel Ice Lake and later
  - AMD EYPC Genoa and later
- Currently enabled by trace flag 15097
  - https://learn.microsoft.com/sql/t-sql/database-console-commands/dbcc-traceon-trace-flags-transact-sql#tf15097

Hybrid Buffer Pool

- SQL Server 2019
  - Clean pages directly referenced on PMEM devices without copy
  - Dirty pages still kept in DRAM
- SQL Server 2022
  - Direct write, reduces number of memcpy operations
  - Currently enabled by trace flag 809
    - https://learn.microsoft.com/sql/t-sql/database-console-commands/dbcc-traceon-trace-flags-transact-sql#tf809
- More info
  - https://learn.microsoft.com/sql/database-engine/configure-windows/hybrid-buffer-pool#hybrid-buffer-pool-with-direct-write

Query Store enhancements

- On by default (migrated databases retain original configuration)
- Query Store Hints (shape plans with no code changes)
- Query Store support for AG secondary replicas
- Now used also by Intelligent Query Processing

Query Store hints

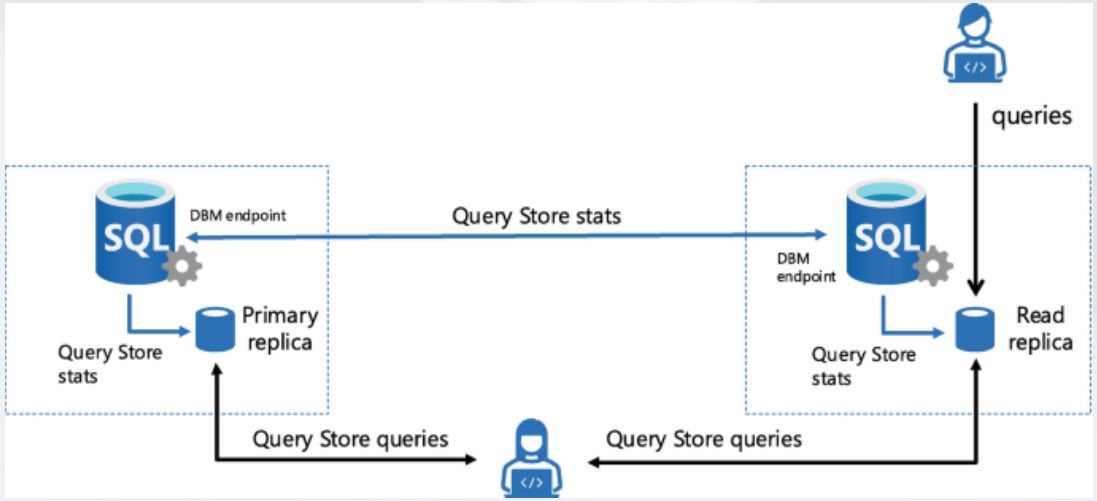
- Process much simpler than *Plan Guides*
  - Find query id in Query Store
  - Use **`sys.sp_query_store_set_hints`** to apply hint
  - Check for hint failures in **`sys.query_store_query_hints`**
- Hints survive plan cache eviction and restart!
- More info
  - https://learn.microsoft.com/sql/relational-databases/performance/query-store-hints

Query Store hints support

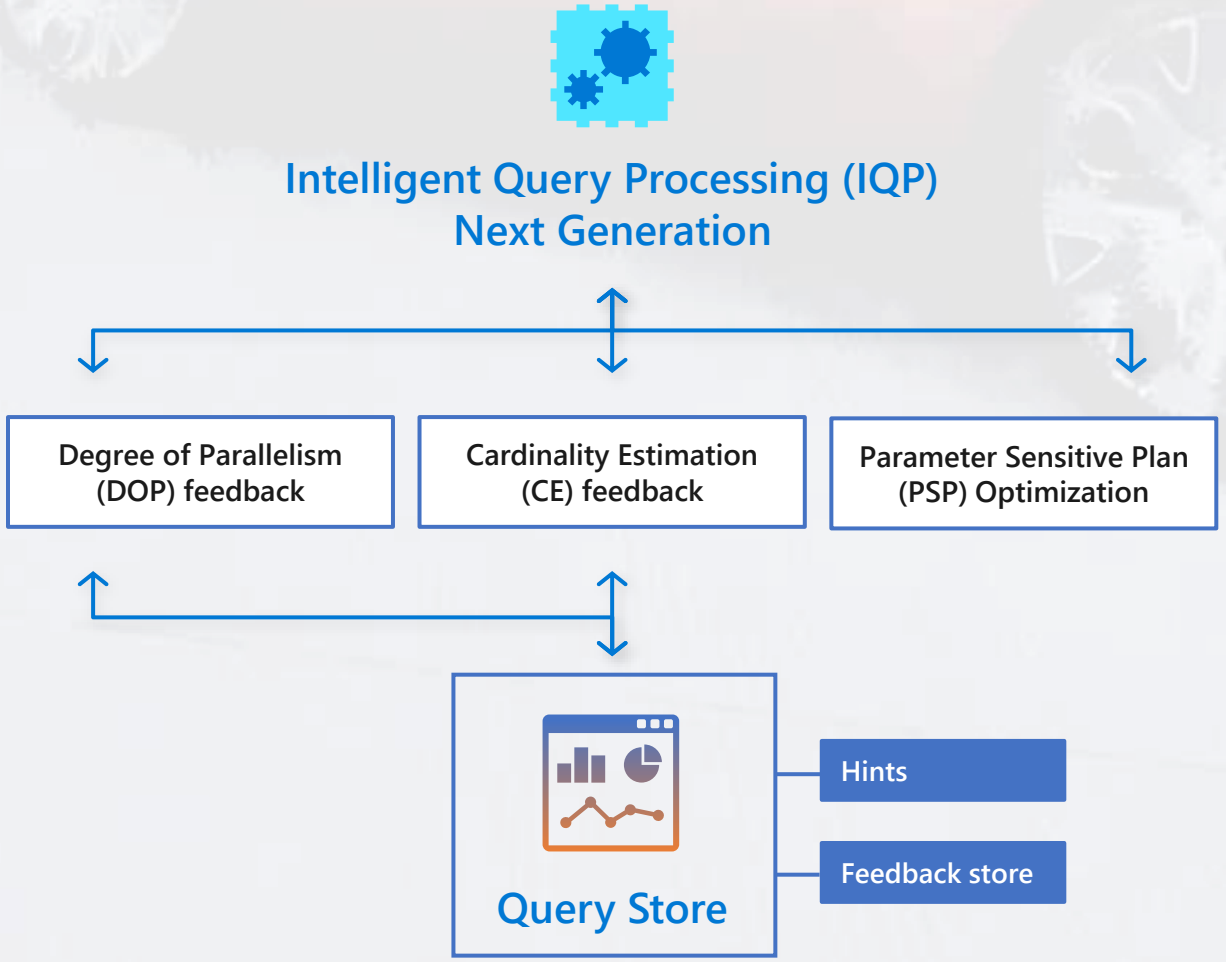
- **Supported hints**
  - { HASH | ORDER } GROUP
  - { CONCAT | HASH | MERGE } UNION
  - { LOOP | MERGE | HASH } JOIN
  - EXPAND VIEWS
  - FAST number_rows
  - FORCE ORDER
  - IGNORE_NONCLUSTERED_COLUMNSTORE_INDEX
  - KEEP PLAN
  - KEEPFIXED PLAN
  - MAX_GRANT_PERCENT = percent
  - MIN_GRANT_PERCENT = percent
  - **MAXDOP** number_of_processors
  - NO_PERFORMANCE_SPOOL
  - OPTIMIZE FOR UNKNOWN
  - PARAMETERIZATION { SIMPLE | FORCED }
  - **RECOMPILE**
  - ROBUST PLAN
  - **USE HINT** ('<hint_name>' [, ...n])
- **Unsupported hints**
  - OPTIMIZE FOR(@var = val)
  - MAXRECURSION
  - USE PLAN
    - consider QP original plan forcing
  - DISABLE_DEFERRED_COMPILATION_TV
  - DISABLE_TSQL_SCALAR_UDF_INLINING
  - Table hints
    - E.g. FORCESEEK, READUNCOMMITTED, INDEX
- **Feature interoperability**
  - https://learn.microsoft.com/sql/relational-databases/performance/query-store-hints#query-store-hints-and-feature-interoperability

Query Store for AG secondary replicas

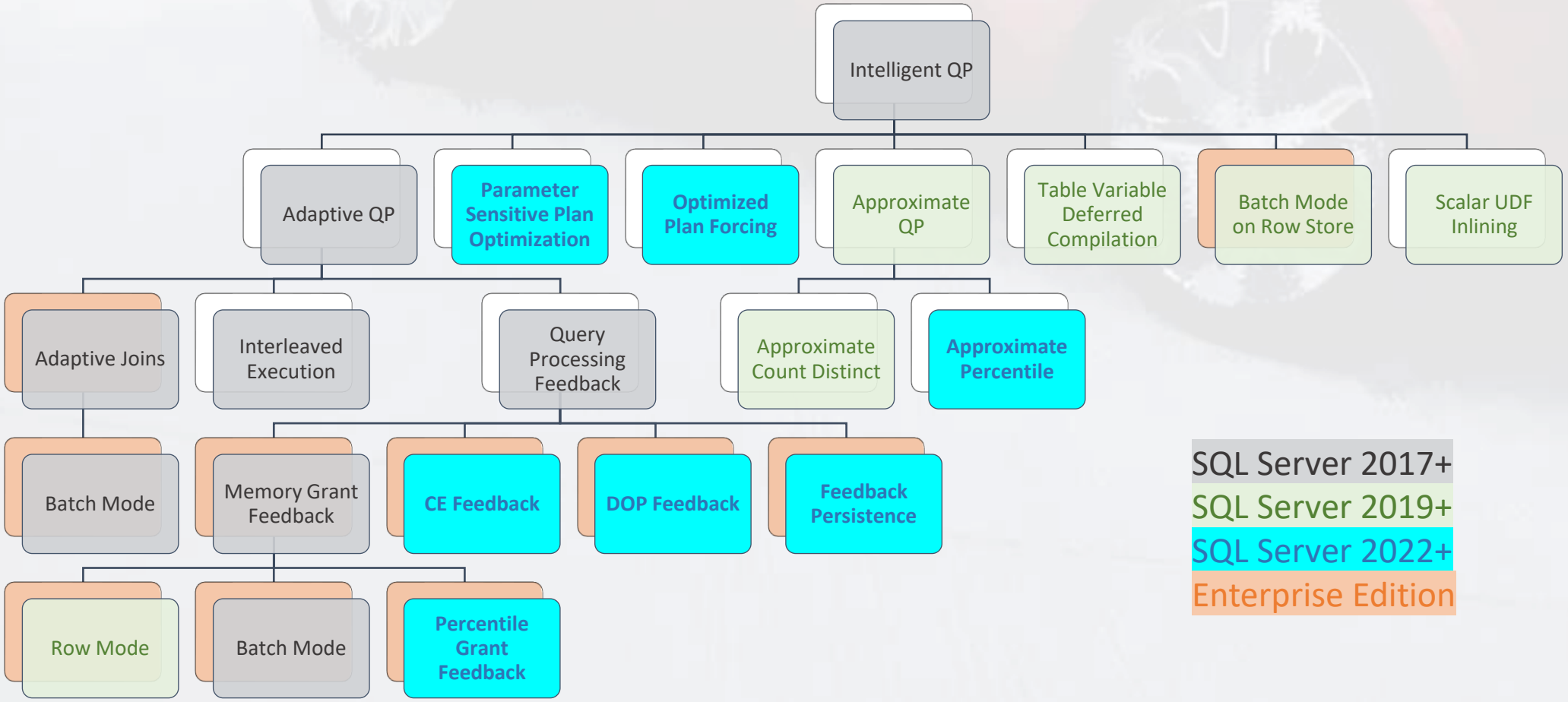
- Primary replica stores data
  - For all replicas
  - Column **`replica_group_id`** only in
    - **`sys.query_store_replicas`**
    - **`sys.query_store_runtime_stats`**
    - **`sys.query_store_wait_stats`**
- Enabled by trace flag 12606
- More info
  - https://learn.microsoft.com/sql/relational-databases/performance/query-store-for-secondary-replicas



Query Store and IQP



Intelligent Query Processing Gen 3



SQL Server 2017+

SQL Server 2019+

SQL Server 2022+

Enterprise Edition

IPQ changes not gated by dbcompat…

SQL Server 2022

- Approximate Percentile
  - PERCENTILE_CONT
  - PERCENTILE_DISC
- Optimized Plan Forcing
  - Persists compile steps
  - Reduce compilation overhead

dbcompat 140+

- Memory Grant Percentiles
  - Smooth oscillation
- Memory Grant Feedback Persistence
  - Survive cache eviction and restart

dbcompat 160

- Parameter Sensitive Plan (PSP) Optimization
- Cardinality Estimation (CE) Feedback
- Degree of Parallelism (DOP) feedback

Hardware offloaded backup compression

- Currently only Intel QuickAssist (QAT) supported
  - Install drivers
    - Diagnostic DMV `sys.dm_server_accelerator_status`
  - Enable functionality
    - `sp_configure 'hardware offload enabled'`
  - Enable specific accelerator
    - `ALTER SERVER CONFIGURATION SET HARDWARE_OFFLOAD = ON|OFF (ACCELERATOR = QAT[, MODE = SOFTWARE])`
  - Backup
    - `COMPRESSION ALGORITHM = MS_XPRESS|other_algorithm`
- Usually, good performance with concurrent intensive workload
- More info
  - https://docs.microsoft.com/sql/relational-databases/system-tables/backupset-transact-sql

Learn more

Learn more about SQL Server 2022
aka.ms/sqlserver2022

Download SQL Server 2022
aka.ms/getsqlserver2022

Get our decks
aka.ms/sqlserver2022decks

What's new for SQL Server 2022
aka.ms/sqlserver2022docs

Read the book
aka.ms/sql2022book

Try out the workshop
aka.ms/sql2022workshop

Try our demos
aka.ms/sqlserver2022demos

Watch Microsoft Mechanics
aka.ms/sqlserver2022mechanics

Don't miss us on Data Exposed
aka.ms/dataexposed

Read the SQL Server 2022 Blog Series
aka.ms/sqlserver2022blogs

Take the Microsoft Learning Path
aka.ms/learnsqlserver2022