Optimizing Performance of In-Memory Tables

Pinal Dave
SQL SERVER PERFORMANCE TUNING EXPERT

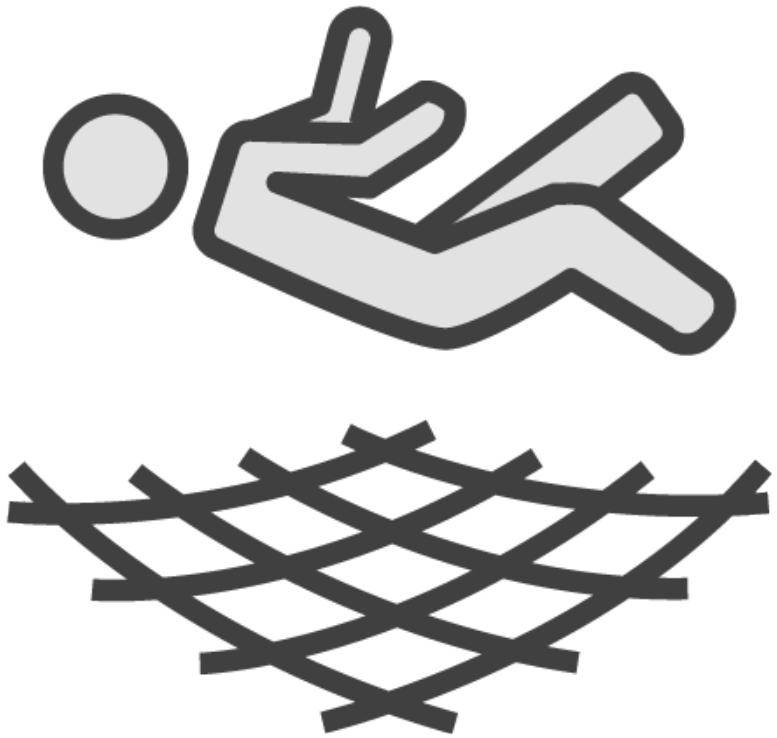
https://blog.sqlauthority.com

Agenda

- Memory Optimized Tables are Faster
- Trade-offs
- Compatibility Level
- Memory Optimized Filegroup
- Snapshot Isolation Level
- Memory-optimized Nonclustered Index
- Memory-optimized Hash Index
- Demo

How Memory Optimized Tables Perform Faster?

- Dual nature
- No locks
- Row versions
- Less logging
- Natively compiled stored procedure



Trade-offs of Memory Optimized Tables

- Estimating memory
- Partitioning large table
- Natively compiled modules

Compatibility Level

It sets certain database behaviors to be compatible with the specified version of SQL Server.

Minimum: 120 Current: 140 Maximum: Latest

Demo

Compatibility Level

Memory Optimized Filegroup

Memory Optimized Tables can only be created on Memory Optimized FileGroups.

Possible to add after database creation

Demo

Memory Optimized Filegroup

Snapshot Isolation Level

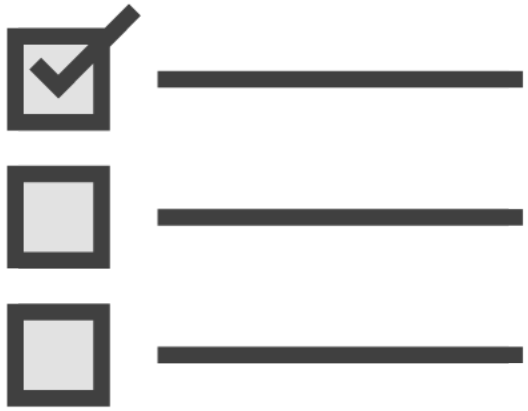
It sets certain database behaviors to be compatible with the specified version of SQL Server .

For cross-container transactions use MEMORY_OPTIMIZED_ELEVATE_TO_SNAPSHOT

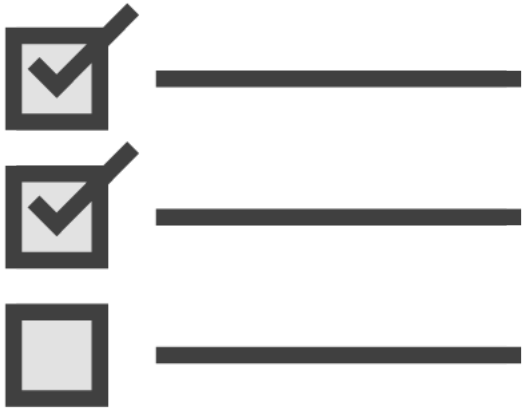
Demo

Snapshot Isolation Level

Indexes on Memory Optimized Tables



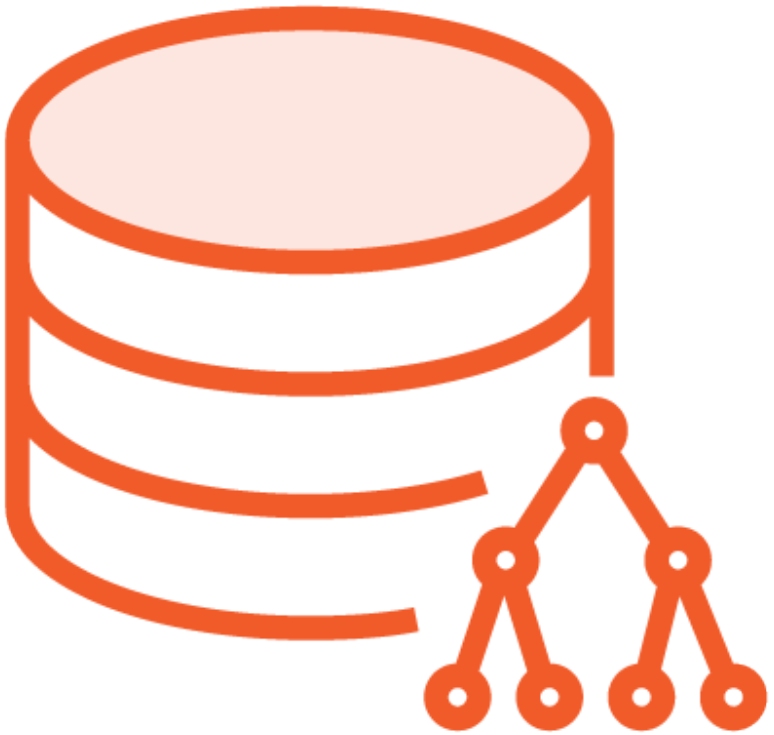
Memory-optimized Nonclustered Index



Hash Index

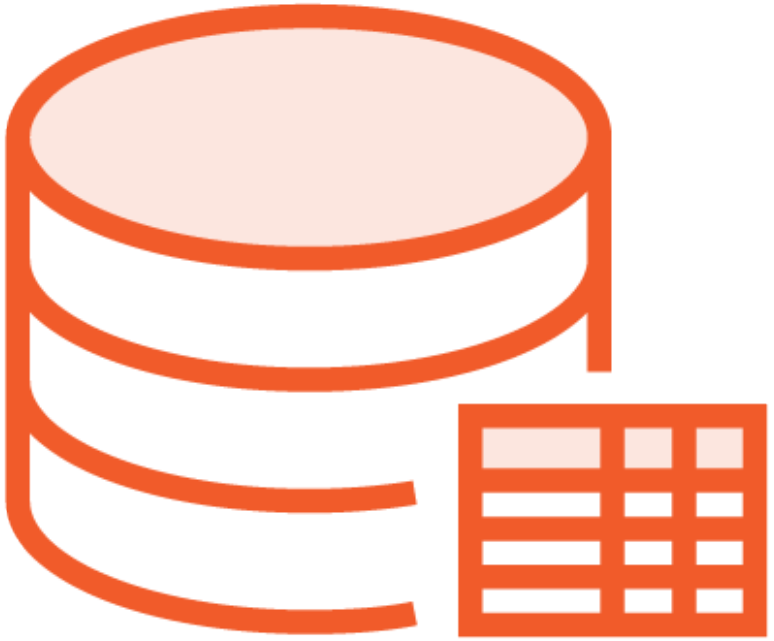
3 Important Differences

- No FillFactor – data rows are stored in memory and not on pages
- Index modification stays in memory
- Rebuilt when database comes online



Memory-optimized Nonclustered Index

- Durability – schema and data
- Primary key nonclustered
- Queries with Order By clause
- Queries with inequality and value ranges



Memory-optimized Hash Index

- Array of pointers
- Element of array is called hash bucket
- Each bucket of 8 bytes
- Max buckets per index 1,073,741,824
- Ideal Bucket Count 2 x (distinct rows)
- Queries with the exact match

Demo

Indexes with Memory Optimized Tables

Summary

Memory Optimized Tables are Faster

Advantages and Tradeoffs

Index Types and Query Patterns