SQL Server 2019 Improvements

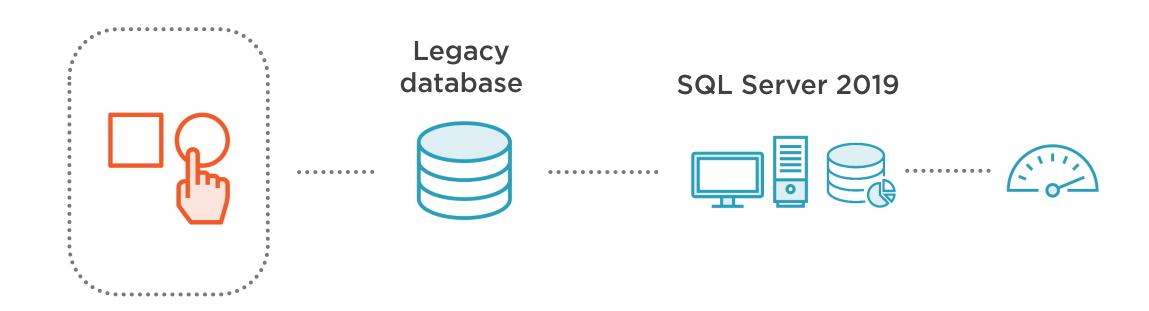


Viktor Suha CONSULTANT

@realeddiesson www.linkedin.com/in/viktor-suha-sql



Database and Server Upgrade





Supportability, security, performance, and new features

Version, Edition, Patching



SQL Server Major Versions

Supportability and servicing

SQL Server 2012 11.0 (v11) SQL Server 2016 13.0 (v13) **SQL Server 2019** 15.0 (v15)

SQL Server 2014 12.0 (v12) **SQL Server 2017** 14.0 (v14)



Database Compatibility Levels

Major Version

Supported Compatibility Levels

SQL2019

| | '

150 140 130 120 110 100

SQL2017

140 130 120 110 100

SQL2016

130 120 110 100

SQL2014

120 110 100

SQL2012

110 100 90



Compatibility Level Impact on Performance

Compatibility Level 140

Cardinality Estimator (CE) v140

Automatic tuning

Adaptive joins for batch mode

Interleaved execution

Memory grant feedback for batch mode

Compatibility Level 150

Cardinality Estimator (CE) v150

Batch mode on rowstore

Memory grant feedback for row mode

Scalar UDF inlining

Table variable deferred compilation



Why Does the Edition Matter?



Feature limitation



Scale and resource utilization limits



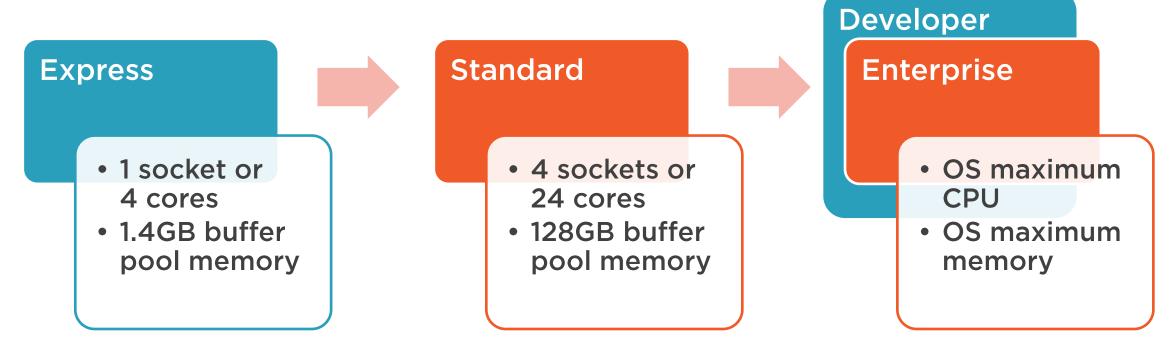
Licensing cost



SQL Server Edition and Feature Matrix

Editions and supported features of SQL Server 2019

https://bit.ly/3j1zmFE



Intelligent database and performance



SQL Server 2019 Servicing



No Service Packs from SQL2017 onwards

- Modern Servicing Model

CU 12 for SQL Server 2019

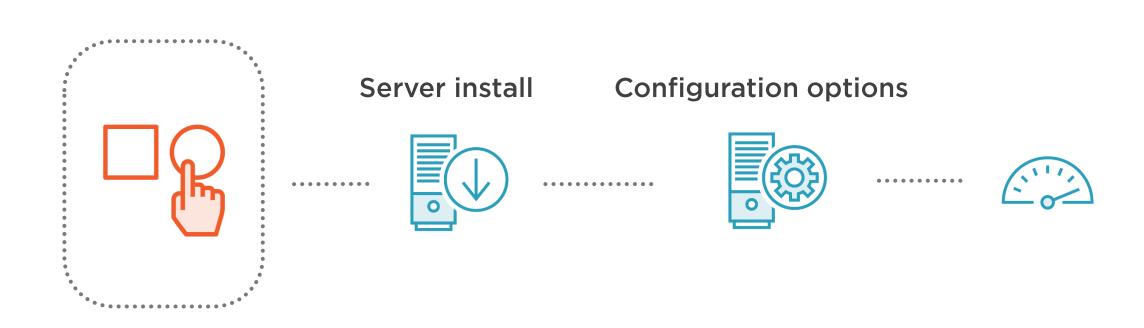
- August 2021



Setup



Setup Improvements





Recommended server configuration options

New Setup Options and Features

Memory

Recommended min and max server memory (MB) values

MAXDOP

Recommended max degree of parallelism value

Licensing

Server/CAL licensing warning tied to the number of CPU cores



Tempdb



Tempdb Configuration and Major Versions

Performance

SQL Server 2012 All manual with trace flags SQL Server 2016
Built-in
improvements

SQL Server 2019 In-memory metadata

SQL Server 2014
All manual with trace flags

SQL Server 2017
Built-in
improvements



Previous Tempdb Setup Improvements

Multiple data files
SQL Server 2016

Large data file size
SQL Server 2017



New Tempdb Server Configuration Option



Tempdb metadata memory-optimized

- Disabled by default (on-premises)

When to turn it on?

- Tempdb-heavy workloads with tempdb metadata contention
- PAGELATCH waits in tempdb



Configuring In-memory Tempdb Metadata

```
ALTER SERVER CONFIGURATION SET MEMORY_OPTIMIZED TEMPDB_METADATA = ON;
```

```
SELECT SERVERPROPERTY('IsTempdbMetadataMemoryOptimized');
```



Query Store



Custom Capture Policy for Query Store



QUERY_CAPTURE_MODE
QUERY_CAPTURE_POLICY



Using Custom Capture Policy for Query Store

```
ALTER DATABASE [WideWorldImporters] SET QUERY_STORE = ON
    OPERATION_MODE = READ_WRITE,
    QUERY_CAPTURE_MODE = CUSTOM,
    QUERY_CAPTURE_POLICY = (
        STALE_CAPTURE_POLICY_THRESHOLD = 24 HOURS,
        EXECUTION\_COUNT = 30,
        TOTAL_COMPILE_CPU_TIME_MS = 1000,
        TOTAL_EXECUTION_CPU_TIME_MS = 100 )
```

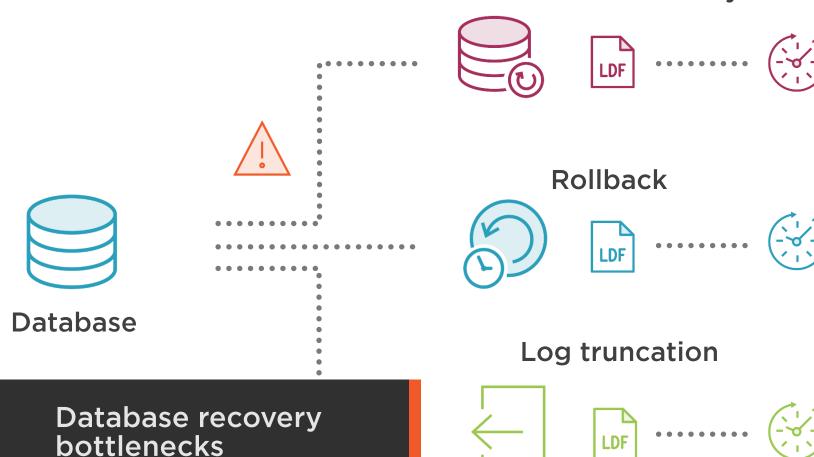


Accelerated Database Recovery (ADR)



Standard Database Recovery

Database recovery



Accelerated Database Recovery Setting



ACCELERATED_DATABASE_RECOVERY

Disabled by default (on-premises)



Configuring Accelerated Database Recovery

```
ALTER DATABASE [WideWorldImporters]
SET ACCELERATED_DATABASE_RECOVERY = ON;
```

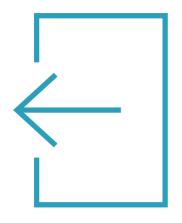
```
SELECT is_accelerated_database_recovery_on
FROM sys.databases
WHERE [name] = 'WideWorldImporters';
```



Accelerated Database Recovery Advantages









Faster database recovery

Faster transaction rollback

Aggressive log truncation

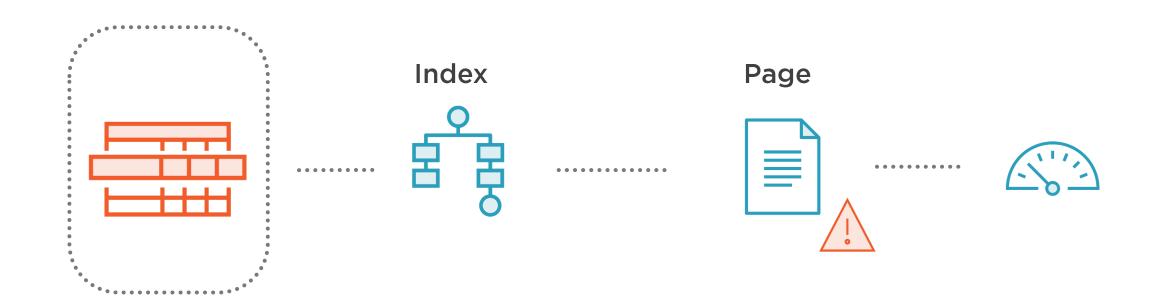
Improved database availability



Sequential Key Insert Optimization



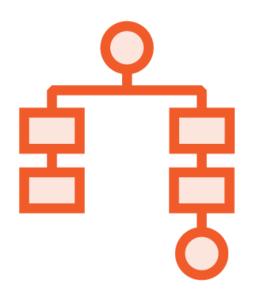
Sequential Key Insert Bottleneck





PAGELATCH_EX contention

Optimizing for Sequential Key Inserts



OPTIMIZE_FOR_SEQUENTIAL_KEY

- Index option



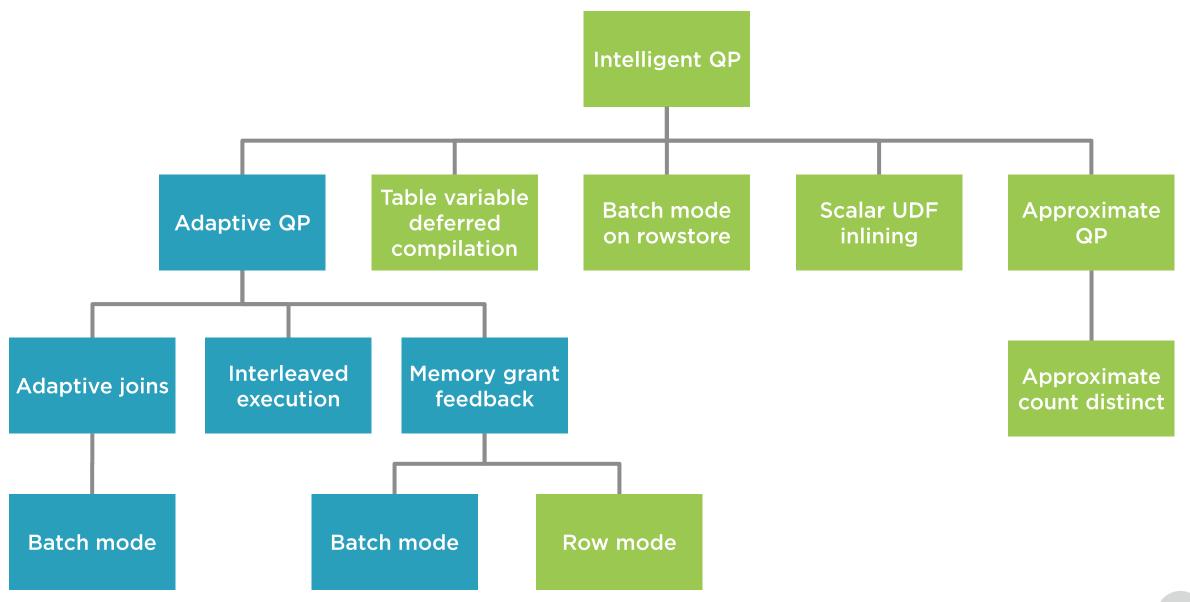
Using the New Index Option

```
CREATE TABLE [dbo].[Cities](
    [CityID] [int] NOT NULL,
    [CityName] [nvarchar](50) NOT NULL,
    CONSTRAINT [PK_Cities]
        PRIMARY KEY CLUSTERED ([CityID] ASC)
    WITH (OPTIMIZE_FOR_SEQUENTIAL_KEY = ON)
);
```



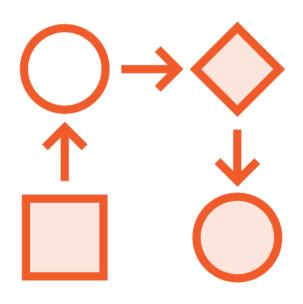
Intelligent Query Processing







Intelligent Query Processing



Batch mode on rowstore

Memory grant feedback for row mode

Scalar UDF inlining

Table variable deferred compilation

APPROX_COUNT_DISTINCT





More information

Troubleshooting SQL Server Performance Problems

Viktor Suha



References



Microsoft KB, <u>SQL Server 2019 Build Versions</u>



Microsoft KB, <u>SQL Server Modern Servicing Model</u>



Microsoft Docs, <u>SQL Server 2019 Editions and Features</u>



Microsoft Docs, <u>SQL Server 2019 Intelligent Query Processing</u>



Module Summary



Version, edition, and patching
Setup features
Tempdb in-memory metadata
Query Store custom capture policy
Accelerated Database Recovery (ADR)

Sequential key insert optimization
Intelligent query processing



Course Summary



How to approach performance problems and scalability

Why SQL Server patching, maintenance and configuration are important

How SQL Server works under the cover

Why server health-checks are crucial

How to configure memory, parallelism settings and tempdb

How to size SQL Server in Azure, with laaS and PaaS offerings

What troubleshooting methods and tools are available in SQL Server

