



Sergio Govoni



SQL Server 2022 Parameter Sensitive Plan Optimization



<https://bit.ly/3gxFh8e>

Sergio Govoni



Sergio Govoni



twitter.com/segovoni



github.com/segovoni

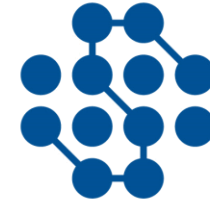


linkedin.com/in/sgovoni

Sponsor & Org



UNIVERSITÀ DEGLI STUDI DI PARMA



DATA SKILLS
UNDERSTANDING THE WORLD



Agenda

- Introduction to Plan Caching mechanisms
- Introduction to Parameter Sensitivity
- SQL Server 2022 Parameter Sensitive Plan (PSP) Optimization

Introduction to plan caching mechanisms

Plan Cache internals

- Using [sys.dm_exec_cached_plans](#) you can inspect the plan cache
- There are six different types of plan that could be cached
 - Compiled Plan
 - Compiled Plan Stub
 - Parse Tree
 - Extended Proc
 - CLR Proc
 - CLR Function
- For Compiled Plan category we have three types
 - Adhoc
 - Prepared
 - Proc

Ad-hoc Plans

- Ad-hoc plans are the default operation for SQL Server plan caching mechanism
 - In order to reuse an ad-hoc plan, you must submit the same query text
- The environment must be identical, SET options must be identical!
- Every difference will force a recompilation of the query and another plan will be generated for the same logical query
- usecounts gives you a good indication about how many times a particular plan has been used

Optimize for Ad-hoc workloads

- Server level option, available since SQL 2008
- Ad-hoc plans are not cached on the first use
 - It only creates a Compiled Plan Stub
- Better memory management
 - SQL Server will be more conservative to cache the ad-hoc plans
- One more recompilation is necessary
- [sys.dm_exec_cached_plans](#)
 - objtype “Adhoc”
 - cacheobjtype “Compiled Plan Stub”

Prepared Plans

- When objtype shows “Prepared” means that SQL Server has used parameters in the execution plan
- Auto-parameterization
 - SIMPLE (default) applied to small class of queries
 - FORCED parametrization, database option (since SQL 2005), allows SQL Server to parametrize more queries
- Explicit parameterization
 - Could be done at the application level by “Prepare” method

Prepared Plans

- When SQL Server decides to parameterize, it decides not only what values in your query is able to parameterize, it will also decide what data type the parameter should be
- A shell query (ad-doc) is created for each unique query text
 - Used to find the parameterized query
 - It doesn't contain the full plan

SIMPLE Parametrization

- SQL Server will perform simple parametrization if the plan will be recognized as “Safe”
- Simple parameterization requires simple query!
- A query isn't simple if it contains
 - JOIN
 - Sub-queries
 - SET clause that contains variables (for UPDATES)
 - UNION, INTO, DISTINCT, TOP
 - GROUP BY, HAVING or COMPUTE BY
 - ...

FORCED Parametrization

- When you set the PARAMETERIZATION database option to FORCED, SQL Server will start to parameterizing almost everything
- [sp_get_query_template](#) returns the parameterized form of a query mimic FORCED parametrization
- Exclusions
 - INSERT...EXEC
 - Statements into SPs
 - Triggers
 - UDFs
 - Cursors
 - Statements running with ANSI_PADDING or ANSI_NULLS set to OFF
 - ...

DEMO

Introduction to Parameter Sensitivity

Parameter Sensitivity

- Parameter sensitivity, also known as Parameter Sniffing, refers to a process whereby SQL Server “sniffs” the current parameter values during compilation or recompilation, and passes it along to the Query Optimizer
- Parameter values are sniffed during compilation or recompilation for the following types of batches
 - Stored procedures
 - Queries submitted via `sp_executesql`
 - Prepared queries

Parameter Sensitivity

- A parameter sensitive plan problem happens when the query optimizer generates a query execution plan that is optimal only for a specific parameter value (or set of values)
- Plans that are not optimal can then cause query performance problems
- It isn't necessary a bad thing
- It's a bad thing when it's a bad thing ☹️
- It's a good thing when it's a good thing 😊, so it depends!
- The problems come when cached plan, with compiled parameters value, is not good enough for current parameters

SQL Server 2022

SQL Server 2022

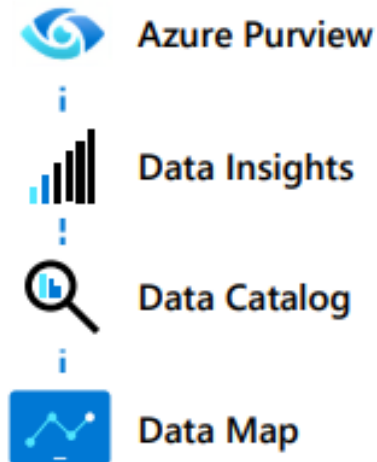
Business continuity through Azure



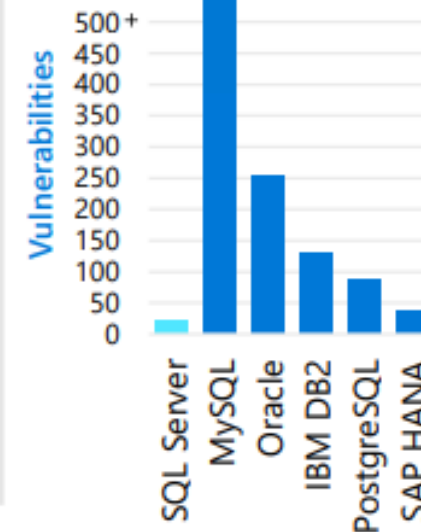
Seamless analytics over on-prem operational data



Visibility over your entire data estate



Most secure over the last 10 years¹



Industry-leading performance and availability

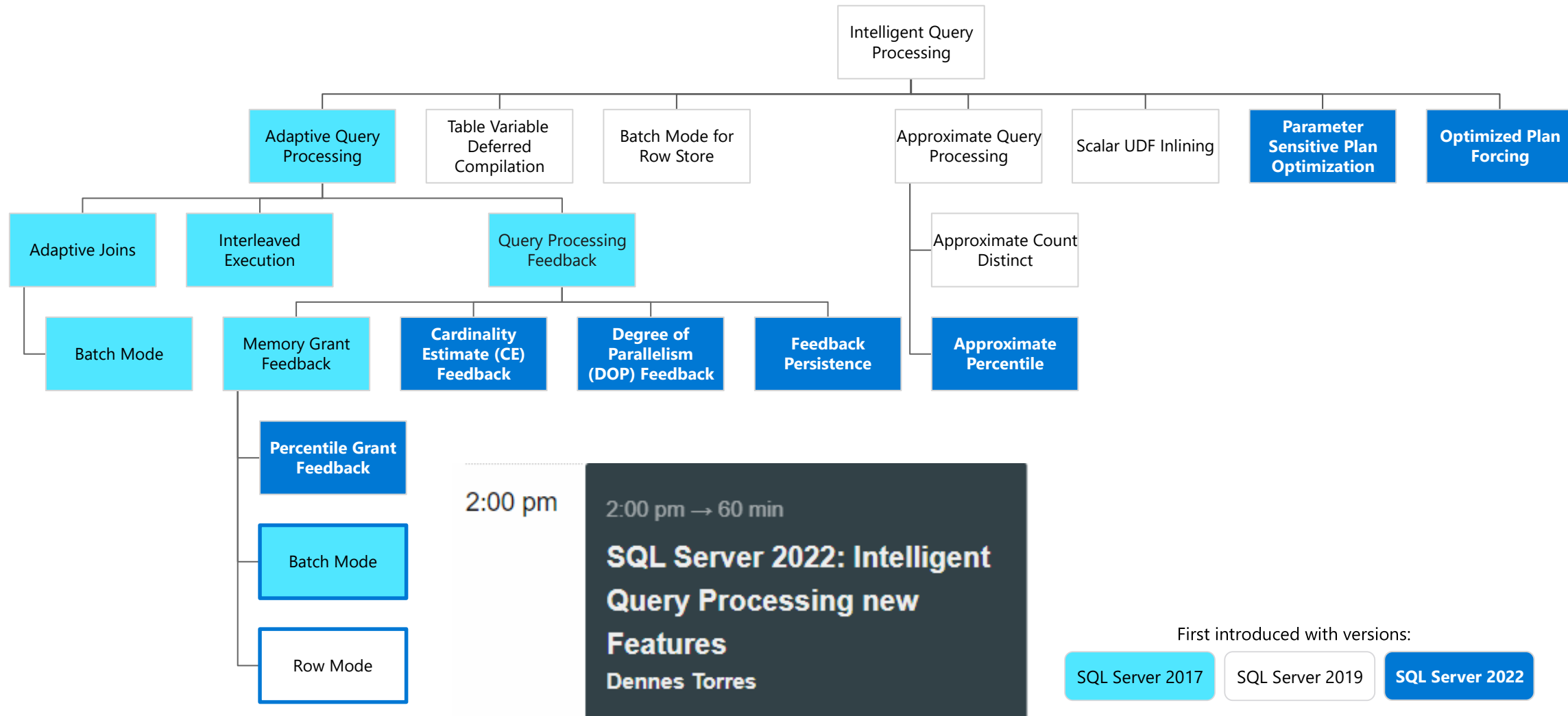


#1 OLTP performance²
#1 DW performance²
on 1TB³, 3TB⁴, 10TB⁵,
30TB⁶ and 100TB⁷

All TPC Claims as of 10/06/2021.

¹ <http://www.tpc.org/4087>; ² <http://www.tpc.org/3374>; ³ <http://www.tpc.org/3380>; ⁴ <http://www.tpc.org/3362>; ⁵ <http://www.tpc.org/3364>; ⁶ National Institute of Standards and Technology Comprehensive Vulnerability Database

Intelligent Query Processing (IQP) Next Generation

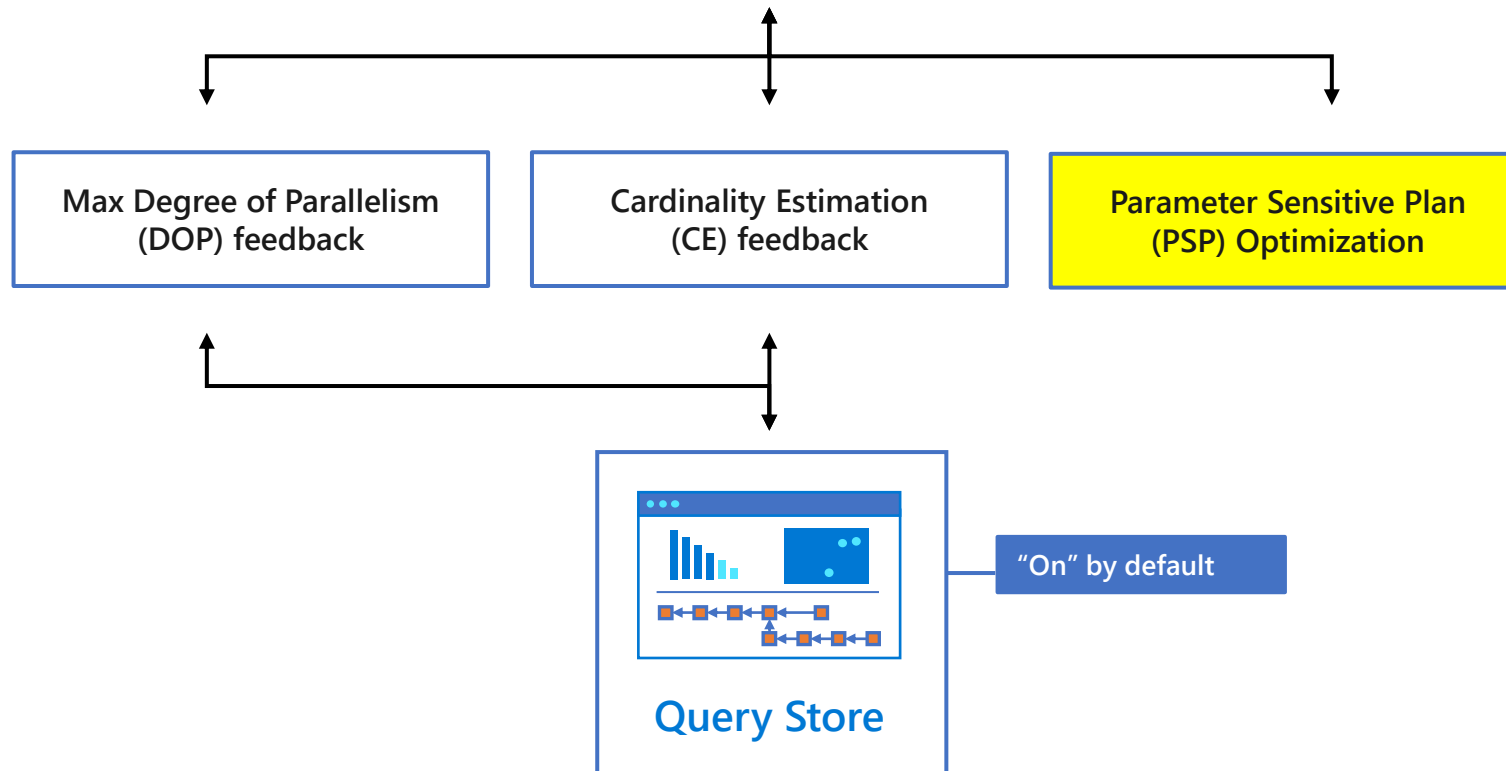


SQL Server 2022 Parameter Sensitive Plan

Query Store and Intelligent Query Processing



Intelligent Query Processing (IQP)
Next Generation



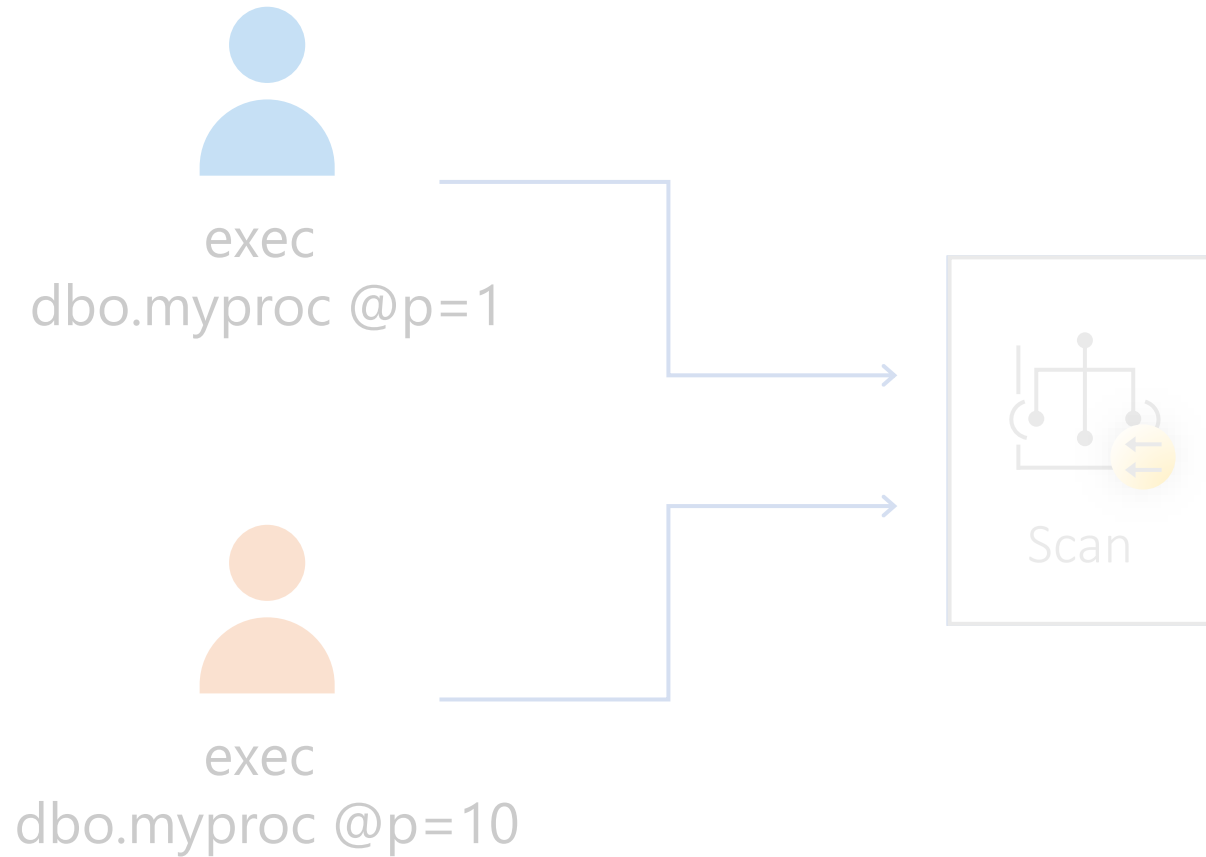
Parameter Sensitive Plan Optimization

- Parameter Sensitive Plan Optimization is one of the new features of SQL Server 2022
- It is part of the family features known as Intelligent Query Processing
- It addressing the scenario where a single cached execution plan for a parameterized query is not optimal for all possible values those parameters can take

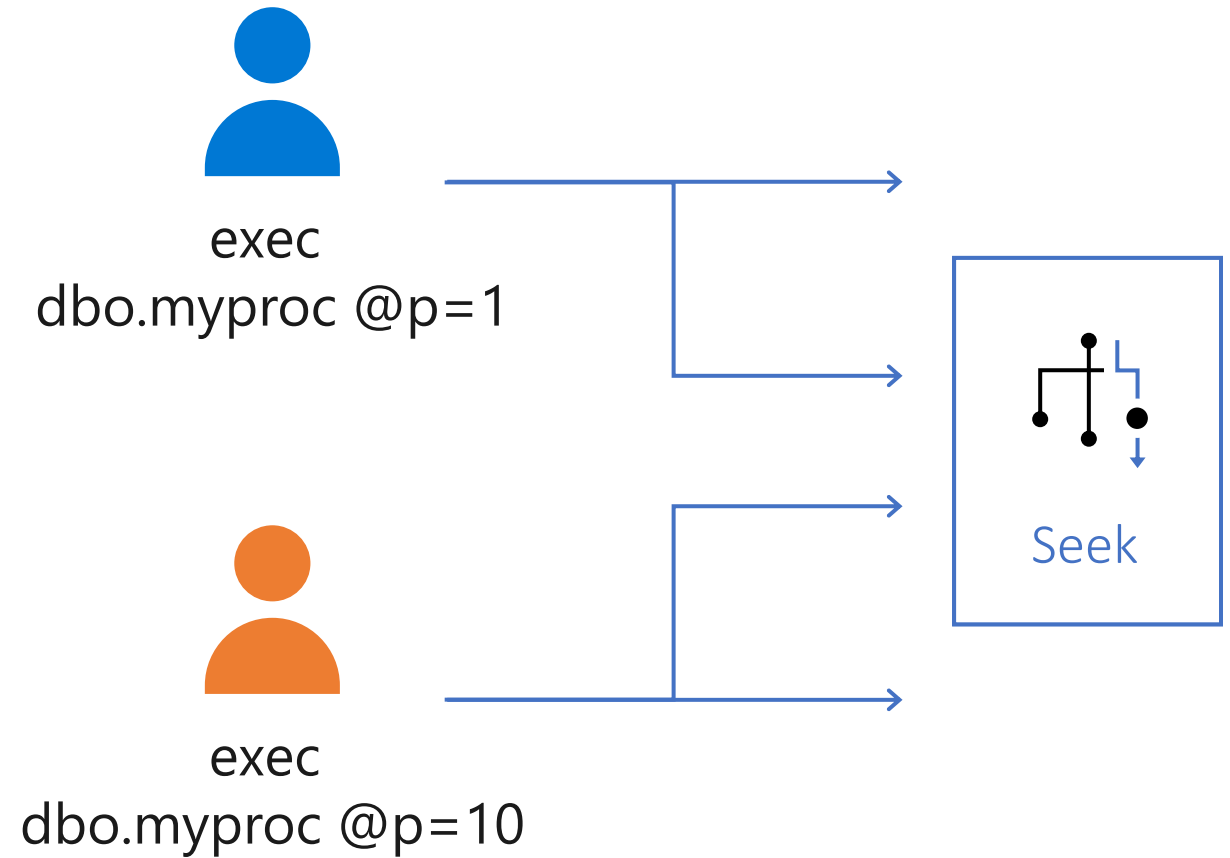
Parameter Sensitive Plan Optimization

- Parameter Sensitive Plan Optimization allows you to keep multiple active execution plans in the plan cache for a single parameterized query
- Each execution plan will be optimized and will host different data sizes depending on the values assumed by the parameters

Before SQL 2022



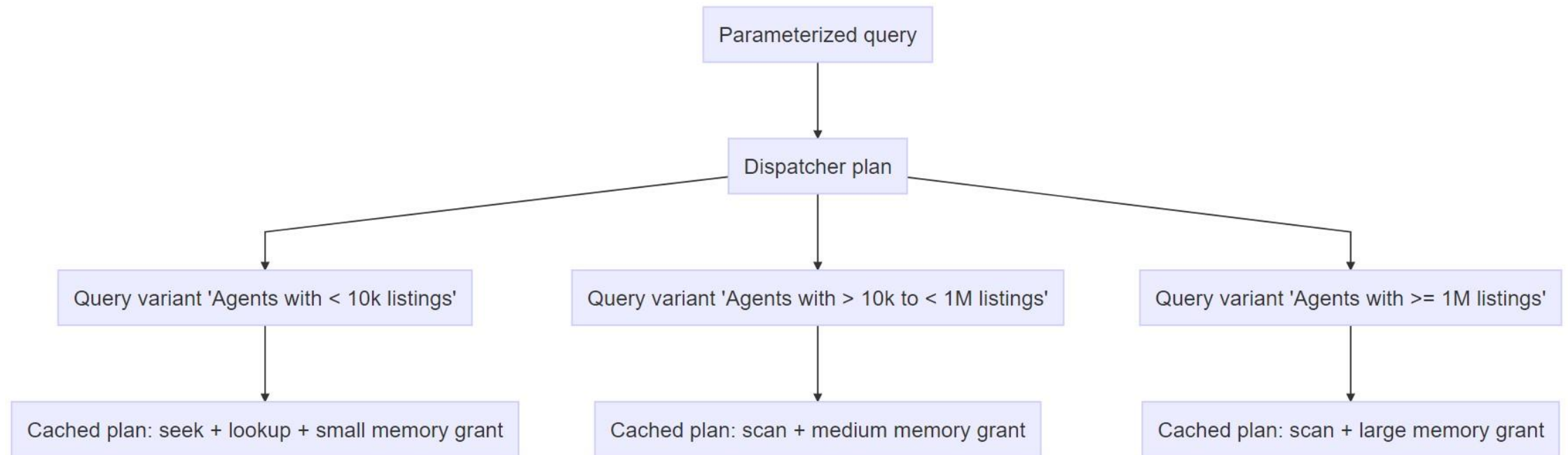
SQL 2022 PSP optimization



Dispatcher plan

- For execution plans suitable to be optimized with PSP, the initial compilation produces a dispatcher plan that contains the optimization logic
- The dispatcher plan maps to query variants based on the predicates cardinality range boundary values
- Dispatcher plans are automatically updated
- Execution plans linked to query variants are recompiled independently as needed

Dispatcher plan



DEMO

Resources

- [Intelligent query processing in SQL databases](#)
- [Query Processing improvements in the latest versions of SQL Server](#)
- [Parameters and Execution Plan Reuse](#)
- [Parameter Sensitivity](#)
- [Queries that have parameter sensitive plan \(PSP\) problems](#)
- [SQL Server 2022 Parameter Sensitive Plan Optimization](#)
- [sqlcmdcli](#)
- [Session materials](#)

Summary

- Query Store enabled by default in SQL Server 2022 combined with the new generation of Intelligent Query Processing allow to improve performance in some common scenarios, without changes to the T-SQL code
- Parameter Sensitive Plan (PSP) Optimization represents one of these improvements because it allows to keep multiple active execution plans in the plan cache for a single parameterized query solving the famous problem known with the name of “Parameter Sniffing”



Thanks!!!

