<https://learn.microsoft.com/en-us/sql/relational-databases/performance/query-store-for-secondary-replicas?view=sql-server-ver16>

**Query Store for secondary replicas**

1. [See also](https://learn.microsoft.com/en-us/sql/relational-databases/performance/query-store-for-secondary-replicas?view=sql-server-ver16#see-also) ->

* [ALTER DATABASE SET options (Transact-SQL)](https://learn.microsoft.com/en-us/sql/t-sql/statements/alter-database-transact-sql-set-options?view=sql-server-ver16)
* [sys.database\_query\_store\_options (Transact-SQL)](https://learn.microsoft.com/en-us/sql/relational-databases/system-catalog-views/sys-database-query-store-options-transact-sql?view=sql-server-ver16)
* [sys.query\_store\_replicas](https://learn.microsoft.com/en-us/sql/relational-databases/system-catalog-views/sys-query-store-replicas?view=sql-server-ver16)
* [sys.query\_store\_plan\_forcing\_locations (Transact-SQL)](https://learn.microsoft.com/en-us/sql/relational-databases/system-catalog-views/sys-query-store-plan-forcing-locations-transact-sql?view=sql-server-ver16)
* [sys.sp\_query\_store\_force\_plan (Transact-SQL)](https://learn.microsoft.com/en-us/sql/relational-databases/system-stored-procedures/sp-query-store-force-plan-transact-sql?view=sql-server-ver16)
* he Query Store for secondary replicas feature enables the same Query Store functionality on secondary replica workloads that is available for primary replicas. When Query Store for secondary replicas is enabled, replicas send the query execution information that would normally be stored in the Query Store back to the primary replica. The primary replica then persists the data to disk within its own Query Store. In essence, there is one Query Store shared between the primary and all secondary replicas. The Query Store exists on the primary replica and stores data for all replicas together. Currently, Query Store for secondary replicas is available with SQL Server 2022 (16.x) instances configured in availability groups.
* **Important**
* Query Store for secondary replicas is a *preview* feature. It is not intended for production deployments. See: [**SQL Server 2022 (16.0) release notes**](https://learn.microsoft.com/en-us/sql/sql-server/sql-server-2022-release-notes?view=sql-server-ver16).
* You must enable trace flag 12606 before you can enable Query Store for secondary replicas. To enable the [**trace flags**](https://learn.microsoft.com/en-us/sql/t-sql/database-console-commands/dbcc-traceon-trace-flags-transact-sql?view=sql-server-ver16):
* In Windows, launch [**SQL Server Configuration Manager**](https://learn.microsoft.com/en-us/sql/relational-databases/sql-server-configuration-manager?view=sql-server-ver16).
* In the list of **SQL Server Services**, right-click on the **SQL Server** instance service for your SQL Server 2022 (16.x) instance. Select **Properties**.
* Select the **Start Parameters** tab. In the **Specify a startup parameter:** field, add the values: -T12606 and select **Add**.
* The SQL Server instance service must be restarted before the changes will take effect.
* **Enable Query Store for secondary replicas**
* Before using Query Store for secondary replicas on a SQL Server instance, you need to have an [Always On availability group](https://learn.microsoft.com/en-us/sql/database-engine/availability-groups/windows/overview-of-always-on-availability-groups-sql-server?view=sql-server-ver16). Then, enable Query Store for secondary replicas using [ALTER DATABASE SET options (Transact-SQL)](https://learn.microsoft.com/en-us/sql/t-sql/statements/alter-database-transact-sql-set-options?view=sql-server-ver16).
* If Query Store is not already enabled and in READ\_WRITE mode on the primary replica, you must enable it before proceeding. Execute the following for each desired database on the primary replica:
* SQLCopy
* ALTER DATABASE [Database\_Name] SET QUERY\_STORE = ON;
* GO
* ALTER DATABASE [Database\_Name] SET QUERY\_STORE
* ( OPERATION\_MODE = READ\_WRITE );
* To enable the Query Store on all secondary replicas, connect to the primary replica and execute the following for each desired database. Currently, when the Query Store for secondary replicas is enabled, it is enabled for all secondary replicas.
* SQLCopy
* ALTER DATABASE [Database\_Name]
* FOR SECONDARY SET QUERY\_STORE = ON (OPERATION\_MODE = READ\_WRITE );
* GO
* To disable the Query Store on all secondary replicas, connect to the primary replica and execute the following for each desired database:
* SQLCopy
* ALTER DATABASE [Database\_Name]
* FOR SECONDARY SET QUERY\_STORE = OFF;
* GO
* You can validate that Query Store is enabled on a secondary replica by connecting to the database on the secondary replica and executing the following:
* SQLCopy
* SELECT desired\_state, desired\_state\_desc, actual\_state, actual\_state\_desc, readonly\_reason
* FROM sys.database\_query\_store\_options;
* GO
* The following sample results from querying [sys.database\_query\_store\_options](https://learn.microsoft.com/en-us/sql/relational-databases/system-catalog-views/sys-database-query-store-options-transact-sql?view=sql-server-ver16) indicate that the Query Store is in a READ\_CAPTURE\_SECONDARY state for the secondary. The readonly\_reason of 8 indicates that the query was run against a secondary replica. These results indicate that Query Store has been enabled successfully on the secondary replica.
* Expand table

| * **desired\_state** | * **desired\_state\_desc** | * **actual\_state** | * **actual\_state\_desc** | * **readonly\_reason** |
| --- | --- | --- | --- | --- |
| * 4 | * READ\_CAPTURE\_SECONDARY | * 4 | * READ\_CAPTURE\_SECONDARY | * 8 |

* Once enabled, you can use [sys.query\_store\_replicas](https://learn.microsoft.com/en-us/sql/relational-databases/system-catalog-views/sys-query-store-replicas?view=sql-server-ver16) to verify the health of the Query Store on the secondary replica.
* To disable Query Store for secondary replicas, connect to the database on the primary replica and run the following code:
* SQLCopy
* ALTER DATABASE CURRENT
* FOR SECONDARY SET QUERY\_STORE = OFF;
* GO
* **Replica sets**
* Currently, when the Query Store for secondary replicas is enabled, it is enabled for all secondary replicas.
* A **replica set** is defined as being all unnamed replicas that share a role (primary, secondary, geo secondary, geo primary), or as being an individual named replica. The data stored about queries can be analyzed as workloads on a replica set basis. Query Store for replicas provides the ability to monitor and adjust the performance of any unique, read-only workloads that might be executing against secondary replicas.
* **Performance considerations for Query Store for secondary replicas**
* The channel used by secondary replicas to send query information back to the primary replica is the same channel used to keep secondary replicas up to date. Data is stored in the same tables on the primary replica that Query Store uses for queries executed on the primary replica, which causes the size of Query Store to grow.
* Thus, when a system is under significant load, you may notice some slowdown because of the channel being overloaded. Further, the same adhoc query capture issues that exist for Query Store today will continue for workloads run on secondary replicas. Learn more about how to [Keep the most relevant data in Query Store](https://learn.microsoft.com/en-us/sql/relational-databases/performance/best-practice-with-the-query-store?view=sql-server-ver16#keep-the-most-relevant-data-in-query-store).