Cloud On-Premise: Performance Maintenance of SQL Server WaspTrack Database

Jason - 2024-12-09 - InventoryCloud Getting Started

Periodic maintenance of your WaspTrack SQL database can improve performance of AssetCloud and InventoryCloud On-Premise.

Description

The WaspTrack database has two performance improving stored procedures that a Database Administrator may run periodically.

spWaspBarcodeInternalDefragTenant	Decreases fragmentation of data within the WaspTrack database.
spWaspBarcodeInternalRecompileStatistics	Recreates indices and execution plans for the WaspTrack database.

These stored procedures are intended for use by Database Administrators or IT professional charged with care and maintenance over the Wasp SQL Server instance.

This procedure assumes familiarity with SQL Server Management Studio (SSMS).

Frequency

Perform this maintenance if performance has slowed.

Alternatively, you might choose to run this maintenance every six months.

Caution

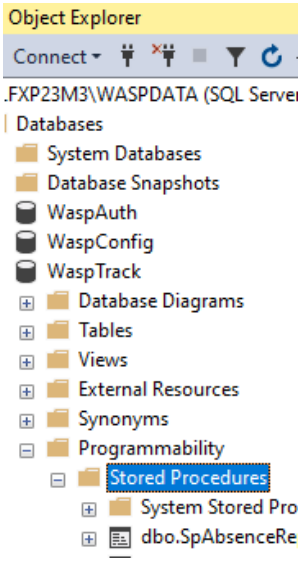
Performing any action on your production SQL databases involves risk.

Use care and take precautions such as backing up your databases before running the stored procedures.

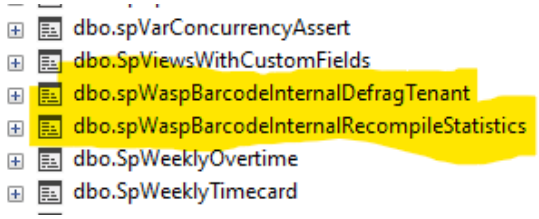
Procedure

Use SQL Server Management Studio to connect to your Wasp SQL instance.

The stored procedures are in the WaspTrack database under Programmability / Stored Procedures.



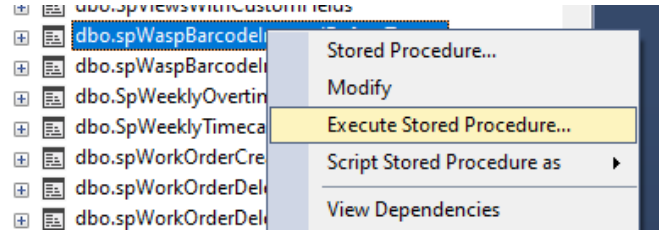
Scroll the Object Explorer until you find the procedures that begin with “dbo.SpWasp”.



DefragTenant

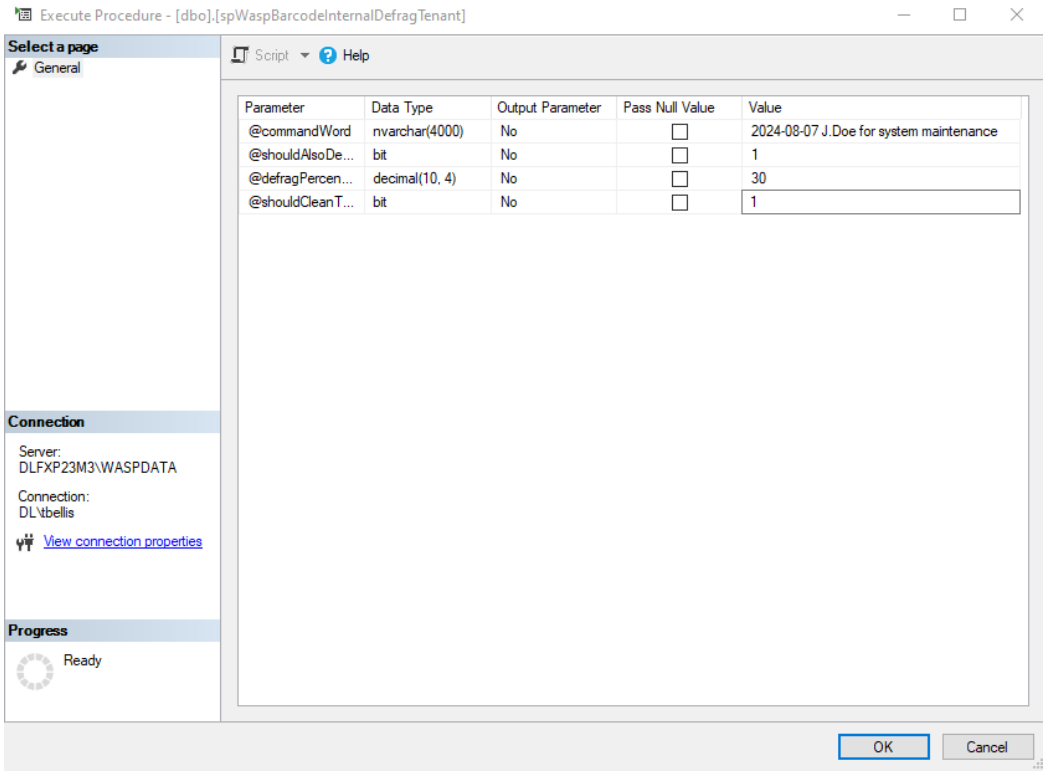
Right click on “dbo.spWaspBarcodeInternalDefragTenant”.

Select “Execute stored procedure…”.



Fill out the parameters.

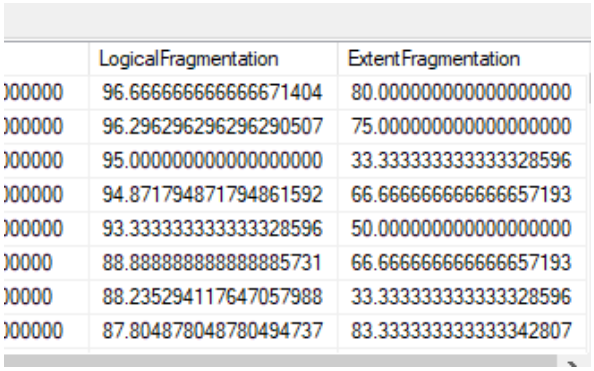
Note that if you send all nulls to the procedure, it merely generates a report of the current fragmentation.



@commandWord	Comment field. Suggest <date> <who> <why>.	2024-08-15 : j doe : system maintenance
@shouldAlsoDefrag	1 = yes, 0 or null = no	1
@defragPercentage	Amount of fragmentation that triggers the defrag on an object.	30
@shouldCleanTable	1 = yes, 0 or null = no	1

When ready press OK. Query should execute successfully.

The first portion of the results is the ‘before’ state of WaspTrack. The last two columns (LogicalFragmentation, ExtentFragmentation) are the key statistics of interest.



	LogicalFragmentation	ExtentFragmentation
)00000	96.666666666666671404	80.000000000000000000
)00000	96.296296296296290507	75.000000000000000000
)00000	95.000000000000000000	33.333333333333328596
)00000	94.871794871794861592	66.666666666666657193
)00000	93.333333333333328596	50.000000000000000000
)0000	88.888888888888885731	66.666666666666657193
)0000	88.235294117647057988	33.333333333333328596
)00000	87.804878048780494737	83.333333333333342807

If you want to see the ‘after’ state of WaspTrack, run the procedure again, but this time pass in all nulls.

Note that the fragmentation of some tables is of no concern. Static tables like [dbo].[api2cart_supported_platform] may be ignored.

LogicalFragmentation	ExtentFragmentation
50.000000000000000000	0.000000000000000000
50.000000000000000000	0.000000000000000000
50.000000000000000000	0.000000000000000000
33.333333333333328596	0.000000000000000000
33.333333333333328596	0.000000000000000000
33.333333333333328596	0.000000000000000000
33.333333333333328596	0.000000000000000000
33.333333333333328596	0.000000000000000000
33.333333333333328596	0.000000000000000000

The fragmentation statistics, especially for “ExtentFragmentation”, should be significantly improved.

Recompile Statistics

The second stored procedure is “dbo. spWaspBarcodeInternalRecompileStatistics”. It is typically of less concern than fragmentation.

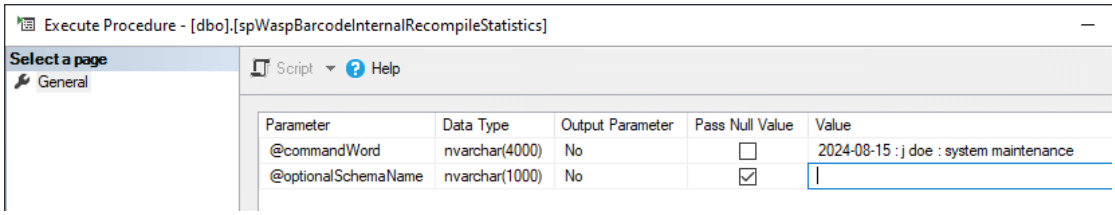
An explanation of this procedure requires a simplified story about how SQL Server runs a query. It builds an execution plan to process the request. It keeps this plan so it doesn’t waste time recreating it each time. On rare occasions, SQL Server becomes confused and makes crazy and horribly inefficient execution plans. Most customers will never have this happen.

This stored procedure rebuilds indices and execution plans, preventing or fixing problems with crazy plans.

Right click on “dbo. spWaspBarcodeInternalRecompileStatistics” and select “Execute Stored Procedure…”

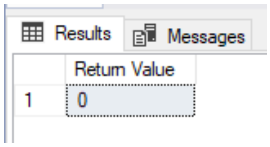
Fill out the parameters or pass nulls.

@commandWord	Comment field. Suggest <date> <who> <why>.	2024-08-15 : j doe : system maintenance
@optionalSchemaName	Do not use.	Pass Null Value



Click OK to run the procedure.

The result is trivial.



If you are curious about what the procedure did, you may look at the messages tab.

Final Notes

Your DBA may schedule these stored procs to run with appropriate parameters.

Your Wasp application should perform better if fragmentation or poor execution plans were present.

These stored procedures normally have little impact on performance while running. Still, since they are

manipulating the WaspTrack database, it is best practices to run when Wasp is not otherwise in use.