

Clustered index delivery in releases after 2021 R1: precautions to be taken

Clustered index definition & benefits

“A clustered index is a structure in which the actual table data resides in the leaf-level pages of the index. The data is physically sorted in the order of the clustering key¹.”

These two properties make it an ideal candidate for the 4GL framework which is designed to favor development simplicity and efficiency, allowing the developer to bring features to the market quicker. This design philosophy means the entire record is read from the database table unless otherwise specified, and it is read in order, also unless otherwise specified. Normally, to satisfy a query in this context SQL Server will perform a secondary read. This secondary read can be eliminated most of the time using a clustered index. Additionally, as the data is already sorted in the key order, a sort operation can often be skipped by the database. Concurrency is also improved using a clustered index as the desired row level locking is more likely to occur.

We constantly seek at managing the growth of our customers by increasing the performance of the database; providing guidance on how to use clustered index when possible for a standard use of Sage X3 becomes important. And being able to deliver some default values is an enhancement we delivered.

Dictionary implementation

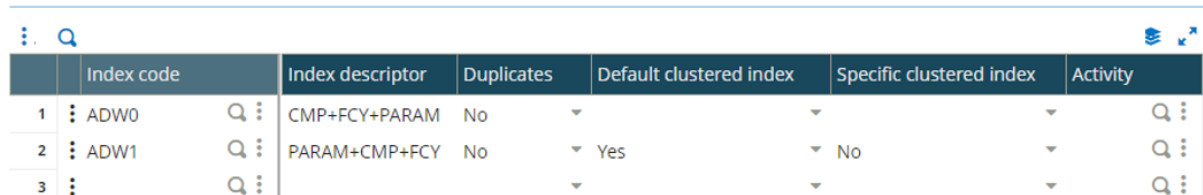
In the previous implementation, it was possible to describe a clustered index for every table, but this was always a customer decision that couldn't be delivered in the standard product based on best practices, and this was clearly a need.

Different performance analysis showed that we can deliver in most of the cases an index that is likely to enhance the overall performance on standard functions when selected as clustered, or to decrease lock contention on some tables that are updated frequently.

For this reason, we have implemented in 2020 R4 a new way of defining a clustered index on SQL server tables, to deliver clustered indexes by default on the tables that haven't a clustered index defined yet. The new index definition in the table dictionary page looks like this:

Index

Index



	Index code	Index descriptor	Duplicates	Default clustered index	Specific clustered index	Activity
1	ADW0	CMP+FCY+PARAM	No			
2	ADW1	PARAM+CMP+FCY	No	Yes	No	
3						

Clustered index deactivated

¹ Administering Microsoft SQL Server 2012 Databases 70-462, Orin Thomas, Peter Ward Bob Taylor

If you have defined a clustered index that you want to use in the customer context, there should be a yes on the right line of the *Specific clustered index* column. This value will always have the priority. This allows to oversee any standard clustered index delivered to take in account specific user cases.

If you want on purpose to deactivate any clustered index on a table, you must check the *Clustered index deactivated*. Default clustered index and specific ones won't be used if the box is checked.

Otherwise, if a standard clustered index has been delivered in an update, it will be used. The validation of the corresponding table will be automatically performed: the indexes will be rebuilt, and the table data will be sorted according to the clustered index.

[Online help about this feature is available here](#)

Clustered index delivery

We started to deliver in 2021 R1 a first set of clustered indexes on supervisor tables (dictionaries, parameters values, texts...).

We plan to deliver additional clustered indexes in the next releases, and this includes additional supervisor tables, but also application tables. The reason why we deliver these updates by increment is to allow to spread the additional update workload across several releases: a customer that upgrades in baby steps (to the latest release every time as soon as a release is delivered) will have a painless and seamless path to optimize the database.

The choice of these clustered indexes is based on the experience accumulated about the most used indexes for the corresponding tables. If you have specific processes that access data frequently with another index, the choice of another clustered index might be relevant in some cases: you can always use the corresponding *Specific clustered index* column if needed. If you had already created a clustered index in previous releases of X3, the update to 2020 R4, where this feature was introduced, keeps your customer-defined clustered index as specific one, so you won't lose them, even if an

Precautions to be taken

Creating a clustered index moves data at the same place as the clustered index, which can have an impact if the physical organization of your database storage is based on different file groups. Make also sure you have enough space to store your transaction log records, as this type of reorganization can generate an important volume of logs.

In summary, customer on SQL server updating Sage X3 to the latest releases must take the following precautions:

- Always perform a backup before starting the operations
- If using snapshots, bear in mind they can be very large based on the amount of data in tables that will be clustered and their numbers.
- Check the list of tables newly clustered between the two releases; the *patch finder* tool will give you this information (select the origin and final release and select the tables).
- If some of these tables have significant volumes, and if you store data and index in separate group of files, check if you have enough space in the corresponding files and check the size for transaction log records.

Using patch finder

Checking the size is especially important if the tables on which a clustered index will be created have a huge volume. You can easily know which are the tables that will be updated by using the *patch finder* tool. Select the origin and final release, select the object *ATB* as shown below:

The screenshot shows the PatchFinder application interface with the following sections:

- Product / Releases:** A diagram showing release timelines for Sage X3, Sage X3 HR & Payroll, and Sage X3 Warehousing. The selected origin and final releases are:
 - Sage X3: 2019 R2 (12.0.17) to 2020 R3 (12.0.23)
 - Sage X3 HR & Payroll: 2019 R1 (12.0.16) to 2020 R3 (12.0.23)
 - Sage X3 Warehousing: 2019 R4 (12.0.19) to 2020 R2 (12.0.22)
- Product Area:** A list of product areas with checkboxes:
 - Select all (checked)
 - Finance (checked)
 - Distribution (checked)
 - Platform (checked)
 - Manufacturing (checked)
 - PJM (checked)
- Type:** A list of types with checkboxes:
 - Select all (checked)
 - Bug (checked)
 - Enhancements (checked)
 - Entry Point (checked)
 - Others (checked)
- Search Field Criteria:** Search criteria settings:
 - Objects (selected) with input field containing "ATB"
 - Approximate search (unchecked)
 - Key (unselected)
 - All fields (unselected)

A large blue "Search" button is located at the bottom of the interface.