

## SQL Schema Compare Comparison Options

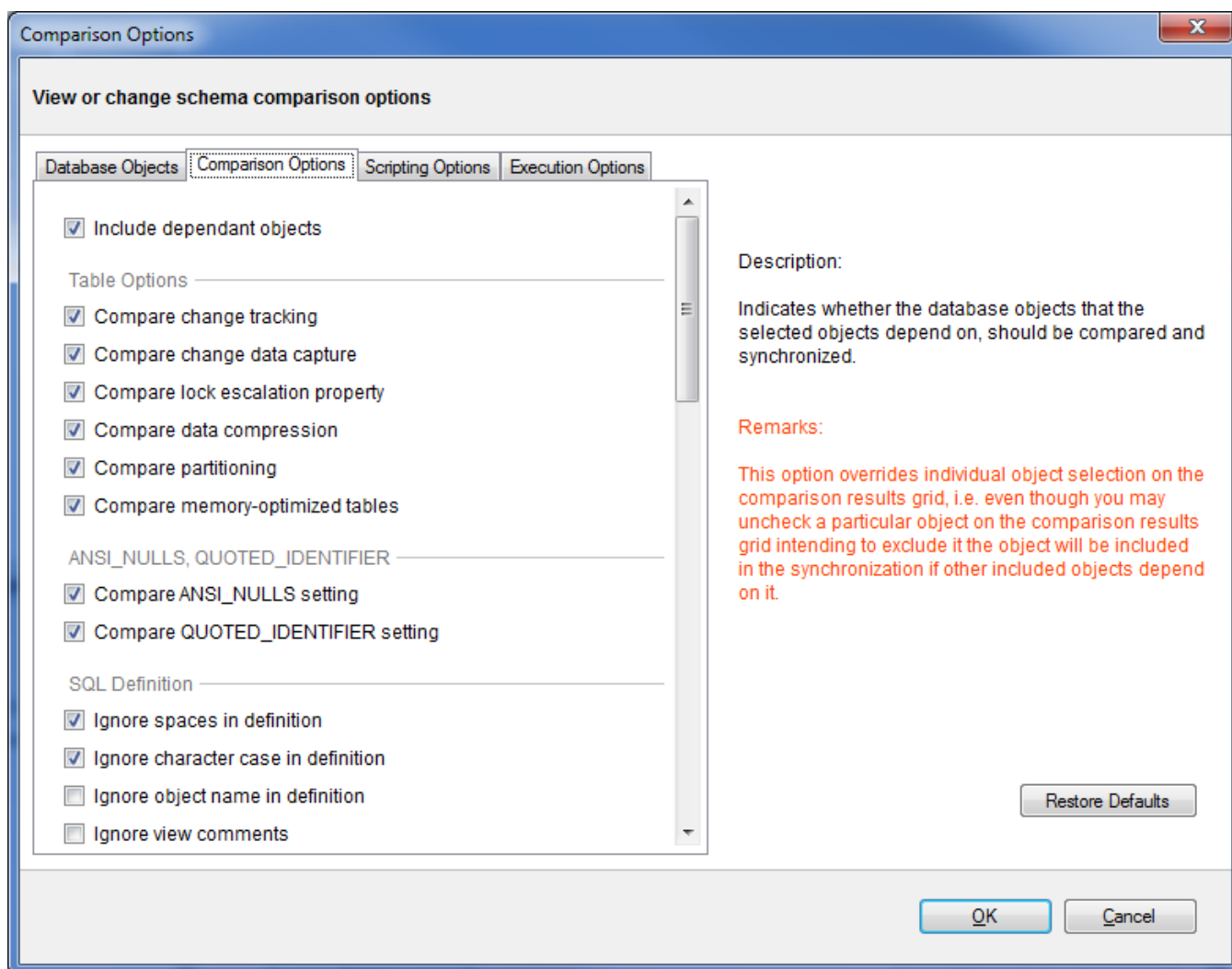
When are two database objects equal? The answer is: it depends! It depends on what you want to compare those objects on - for example: is "Equal" equal to "equal"? The comparison options allow you to have complete control over the comparison engine which considers those options before deciding to mark a pair of objects as equal or different.

So, when two object are found to be different by the comparison engine while you believe they should be equal, or they are found to be equal while you believe they should be different don't panic - investigate the differences and carefully review the comparison options - more likely than not one of those options holds the answer to your surprise.

The comparison options tab provides a description and important remarks for each option while also allowing you to turn those options ON or OFF. To view the description and remarks for a given option simply move the mouse over that option as shown on the image below.

Most of the comparison options are self-explanatory, and for those that are not the description and remarks provided on the "comparison options" window are sufficient. However, there are two options that merit special attention:

- **Script New Columns As NULL.** In cases when the synchronization of the schema requires adding a NOT NULL column on the target but, no default constraint has been specified then SQL Server will reject the action and the synchronization will fail. In such scenarios the best course of action would be to add a default constraint on the source, however, if for one reason or another you cannot add the constraint on the source you can add the column to the target as a "NULL ALLOWED" column to ensure the schema synchronization operation completes successfully. After the synchronization you can connect to the target and handle the changing of that particular column from NULL ALLOWED to NOT NULL.
- **Script New Constraints WITH NOCHECK.** When the schema synchronization requires adding new constraints like foreign key constraints and check constraints on tables that contain data there is a it is possible that the existing data violates those constraints in which case SQL Server will reject the action and the synchronization script will fail to execute. To bypass this obstacle and allow the synchronization operation to go through you can choose to add those constraints WITH NO CHECK first and then check to see if the existing data violates the constraint. When this option is ON then Idera SQL schema compare will generate two synchronization scripts that are executed one after the other:
  - **Synchronization Script.** This will create the new constraints with NOCHECK so the script will execute successfully.
  - **Constraint Enabling Script.** This script enables the constraints that were created WITH NOCHECK and runs on a separate transaction. If the existing data violates one or more of those constraints then the script will fail but that failure does not affect the overall synchronization of the schemas. In case of failure you will need to address the constraint violations on case by case bases.



Total compare from data to schema with IDERA [SQL Comparison Toolset](#). [Learn more](#) > >

<a href="#">IDERA Website</a>	<a href="#">Products</a>	<a href="#">Purchase</a>	<a href="#">Support</a>	<a href="#">Community</a>	<a href="#">About Us</a>	<a href="#">Resources</a>	<a href="#">Legal</a>
-------------------------------	--------------------------	--------------------------	-------------------------	---------------------------	--------------------------	---------------------------	-----------------------