Setting project options

The project configuration options enable you to modify the behavior of SQL Data Compare. For example, you can set an option so that SQL Data Compare ignores certain objects even if they are different, or so that it does not script certain properties for deployment (such as the collation order on columns).

When you create a new project, you should run the comparison with the default options, then review your comparison results. However, if your database is on a SQL Server with case-sensitive sort order, you must select the *Treat items as case sensitive* option. When you have reviewed your comparison results, you may then want to consider changing some of the options.

The options you set are saved for each project, and are modified on the Project Configuration dialog box.

To display the Project Configuration dialog box, click (Edit Project), or select **Project Options** from the Tools menu.

The options you can set are described below. Note that some of the options apply only to the comparison; they do not affect the deployment. Similarly, some options apply only to the deployment. Options affecting mapping and comparison (for example, those relating to case sensitivity) are not applied instantly. You must re-compare the data sources to apply these options.

Default options

To save the current selection of options as your defaults, click **Save As My Defaults**. To restore your defaults, click **My Defaults**. The saved defaults will be used for all new projects.

To reset all the options to their original settings, click **Red Gate Defaults**. The default options for a comparison project are as follows:

- Ignore spaces in object names
- Include identity columns
- Include timestamp columns
- Show identical values in results
- Disable foreign keys
- Reseed identity columns
- Include comment header in the deployment script

Use case sensitive object definition

Considers the case of the object names (tables, views, users, roles, schemas, indexes, and columns) when mapping. For example, [dbo].[Widget] will be mapped to [dbo].[wIDgEt].

> ⚠ Note that:
>
> - if the databases that you are comparing are running on a SQL Server that uses case-sensitive sort order, you should ensure that this option is selected.
> - if you compare a SQL Azure database with this option selected, SQL Data Compare may highlight false differences.

Ignore spaces in object names

Ignores spaces in object names (tables, views, users, roles, schemas, indexes, and columns) when mapping. For example, [dbo].[Widget Prices] will be mapped to [dbo].[WidgetPrices].

Ignore underscores in object names

Ignores underscores in the object names (tables, views, users, roles, schemas, indexes, and columns) when mapping. For example, [dbo].[Widget_Prices] will be mapped to [dbo].[WidgetPrices].

Include views

Includes views in the comparison. Generally, views can be deployed only if the referenced rows are from a single table, and the referenced columns are simple (for example, they must not include identity columns or computed columns).

Include identity columns

Includes identity columns in the comparison. You cannot deploy a view if it includes an identity column.

Include timestamp columns

Includes timestamp columns in the comparison. Timestamp columns cannot be deployed.

Trim trailing spaces

If the data in two columns differs only by the number of spaces at the end of the string, SQL Data Compare considers the data to be identical. This option does not apply to CLR or XML columns.

Trailing spaces are ignored during deployment, if this option is selected.

Force binary collation (case-sensitive)

For all string data types, forces binary collation irrespective of column collation, resulting in a case-sensitive comparison. When this option is selected and the comparison key is a string, this may result in slower performance because the indexes are not used.

Show identical values in results

If this option is not selected, identical values will not be stored on disk nor appear in the comparison results.

Use checksum comparison

Performs a checksum prior to comparison. The data is compared only if the checksums differ. On SQL Server 2000, db_owner permissions are required.

> ⚠ Note that:
>
> - if the data differs only in text or image columns, the checksums will be identical and the data may be flagged incorrectly as identical.
> - you cannot select this option when comparing scripts folders.
> - you cannot select this option if the **Show identical values in results** project option is selected.

Compress temporary files

Compresses the temporary files that SQL Data Compare generates while performing the comparison. This reduces the possibility of running out of temporary disk space when comparing very large databases.

When you select this option, you will not be able to sort the results of the comparison by clicking on a column header in the **Row Differences** pane.

Use maximum precision for floats

> ⓘ This option is only available in SQL Data Compare 10.4.

Select this option is you want SQL Data Compare to compare floating point values to the maximum 17 digits of precision. By default, SQL Data Compare compares floats to 15 digits of precision.

> ⚠ SQL Data Compare can't deploy a value if it is different only in the two additional digits of precision.

Disable foreign keys

Disables then re-enables foreign keys in the deployment SQL script. Note that in some circumstances foreign keys will be dropped and recreated rather than disabled and re-enabled.

Drop primary keys, indexes, and unique constraints

Drops then recreates primary keys, indexes, and unique constraints in the deployment SQL script.

> ⚠ Note that:
>
> - if the primary key, index, or unique constraint is the comparison key, it cannot be dropped.
> - if you deploy to a SQL Azure database, clustered index constraints are not dropped.

Don't use transactions in SQL scripts

Does not insert BEGIN TRANSACTION at the beginning of the deployment SQL script and COMMIT TRANSACTION at the end of the deployment SQL script.

Transport CLR data types as binary

Uses the binary representation of CLR types in the deployment SQL script.

> ⚠ Note that:
> - if this option is not selected SQL Data Compare uses the string representation.
> - if you deploy user-defined CLR types to a SQL Azure database, this option has no effect; SQL Azure does not support user-defined CLR types.

Disable DML triggers

Disables then re-enables DML triggers on tables and views in the deployment SQL script.

Disable DDL triggers

Disables then re-enables DDL triggers in the deployment SQL script.

Reseed identity columns

Reseeds identity columns so that identity values in the database you are updating match values in the source database.

> ⚠ if you deploy to a SQL Azure database, this option has no effect; SQL Data Compare does not reseed identity values.

Include comment header in the deployment script

Includes a comment at the beginning of the deployment script. The heading contains information about the data sources being deployed, and the version of SQL Data Compare.

Don't include comments in the deployment script

If comments are included, it is easier to locate objects in the deployment script. However, the script is smaller if comments are not included.

Force constraints to be re-enabled with CHECK

Forces any constraints (for example, those on foreign keys) disabled by SQL Data Compare to be re-enabled with CHECK.