

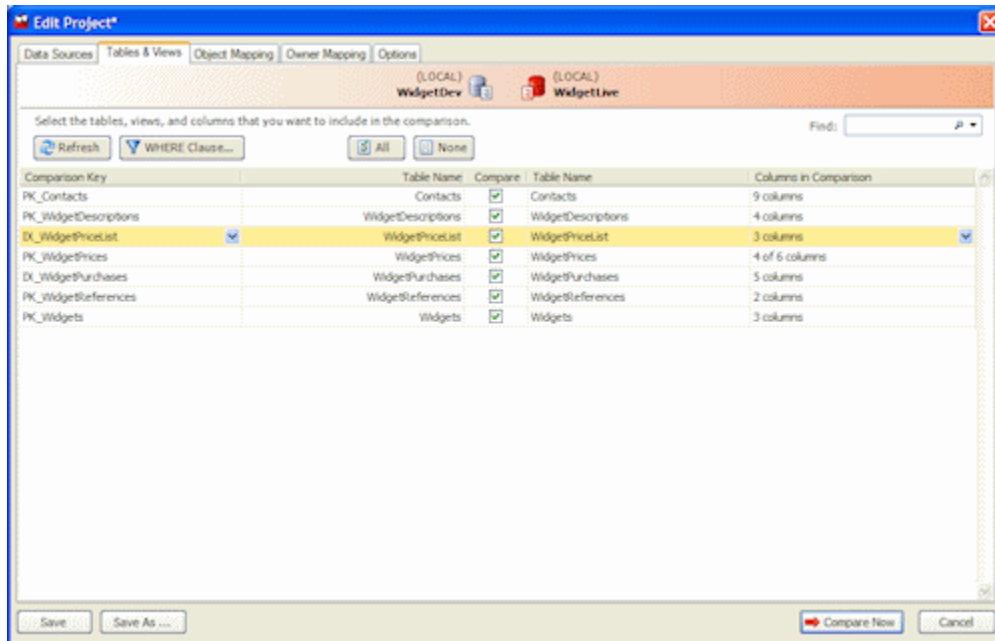
# Selecting tables and views

When you create a project and you have selected your data sources, you can specify which tables, views, and columns to compare. You enter this information using the **Tables & Views** tab on the Project Configuration dialog box.

The **Tables & Views** tab enables you to:

- [select the comparison key for each table or view](#)
- [select the tables and views that will be compared](#)
- [select the columns that will be compared](#)

SQL Data Compare lists the tables and views in the source and target. Tables and views with identical or similar names are displayed side-by-side:



Note that:

- You can filter the specific rows that will be compared by entering a WHERE clause. Filtering can improve the performance of SQL Data Compare. To filter rows, click **WHERE Clause**. The **WHERE Clause Editor** dialog box is displayed. For more information, see [Filtering the comparison results with a WHERE clause](#)
- Views are listed only if the data source is a database and the project option **Include views** is selected.
- You can change the order in which the tables and views are listed by clicking a column header. To sort by multiple columns, click a column header, then hold down SHIFT and click another column header.
- Only the tables and views that are mapped are listed. If you are setting up a new project and SQL Data Compare is unable to map a table or view, you can map the tables and views manually. For more information, see [Mapping objects](#)
- If you are editing an existing project and the structure of the database has changed since you last ran the project, the mappings may be incorrect. In this case, a warning symbol is shown to indicate that those changes affect your project configuration. For more information, see [Mapping errors](#)

## Selecting the comparison key

To match rows in the two data sources, SQL Data Compare requires a comparison key for each table or view.

SQL Data Compare automatically selects a comparison key when:

- tables contain a matching primary key, unique index, or unique constraint
- views contain a matching unique clustered index

For more information, see [What is a comparison key?](#)

If SQL Data Compare is unable to identify a suitable comparison key for a table or view, *Not Set* is shown in the **Comparison Key** box.

To set the comparison key for an object, click its **Comparison Key** box. A dialog box is displayed on which you can select the columns that will comprise the key:

Select the comparison key. Comparison keys enable row matching between the two data sources.

Comparison key: Please set a custom comparison key

Type	Column Name	Key	Column Name	Type
int identity (1,1)	RecordID	<input type="checkbox"/>	RecordID	int identity (1,1)
int	WidgetID	<input type="checkbox"/>	WidgetID	int
money	Price	<input type="checkbox"/>	Price	money
datetime	DateValidFrom	<input type="checkbox"/>	DateValidFrom	datetime
datetime	DateValidTo	<input type="checkbox"/>	DateValidTo	datetime
char(1)	Active	<input type="checkbox"/>	Active	char(1)

Close



- A comparison key cannot include columns whose data type is image, ntext, nvarchar(max), sql\_variant, text, varbinary(max), varchar(max), or xml.
- You cannot specify custom comparison keys if you are using a backup as a data source; however, you can select an alternative unique index or unique constraint.

For large databases, specifying a clustered index as the comparison key can result in improved performance.

## Selecting the tables and views

Select the tables and views you want to compare by selecting or clearing the appropriate check boxes in the **Compare** column. To compare all tables and views, click **All**; to clear all of the check boxes, click **None**.

By default, the first time that you run a project all tables and views with identical or similar names are selected for comparison.

If the structure of the data sources you are comparing has changed while you are working on the project, click **Refresh** to update the **Tables & Views** tab. For example, if a table has been added to the database, click **Refresh** so that you can include the new table in the comparison.

If a table or view has been added to a database since you last ran the project, SQL Data Compare does not select the table or view by default.

For more information, see [Mapping objects](#)

## Selecting columns

SQL Data Compare displays the number of columns that will be compared for each table or view. By default, the first time that you run a project, all columns with identical or similar names are selected for comparison.

You can filter the comparison to consider only specific columns.

To select the columns to compare in a table, click on it in the **Columns in Comparison** column. A dialog box is displayed with check boxes to include or exclude each column:

Select the columns to include in the comparison.

Type	Column Name	Compare	Column Name	Type
int	WidgetID		WidgetID	int
nvarchar(2000)	ShortDescription	<input checked="" type="checkbox"/>	ShortDescription	nvarchar(2000)
text	Description	<input checked="" type="checkbox"/>	Description	text
image	Picture	<input checked="" type="checkbox"/>	Picture	image

Close

You cannot exclude any columns that are used for the comparison key (indicated by .

A warning symbol is shown when [columns you are comparing cannot be mapped](#). This occurs if those columns have:

- column names that do not match
- incompatible data types



- you can compare a *timestamp* column with another *timestamp* column, but you can't deploy *timestamp* columns.
- you can compare an *xml* column with another *xml* column, but you must make sure that your XML schemas are compatible.

For more information about the data types that SQL Data Compare can compare, see [Which data types can be compared?](#)