# A ConfigurationProvider getting its data from a SQL database, accompanied by methods to maintain the database.

This .NET Core C# solution is made up of two projects:

* **JsonSqlConfigDb** contains the ConfigurationProvider and the EF Core (Entity Framework Core) supported classes for connecting to and maintaining the database, in this version a SQL Server database.
* **JsonSqlConfig** is the startup project. It opens up a web api with a Swagger interface and a log console. It also contains database settings, logging settings and launch settings.

The underlying database has only one table, **JsonUnit**. It is in a sense a mirror image of the **JsonElement** .NET class but with two twists:

* It has a **Path** column that enables easier retrieval of data for the ConfigurationProvider. The path is a colon separated string supporting an hierarchical configuration property name. The database maintenance methods set the value of this column.
* It has a **Group** name column that enables maintaining a possibly large configuration set by using smaller chunks of properties. Thus, a group would be a subset of the whole configuration, representing properties for a distinct part of the application. All the database maintenance methods take a **Group** parameter, designating the name of the targeted property group in the database.

The (web) interface for database maintenance accepts any valid Json string as input and it is broken up into elements and stored as a set of rows in the **JsonUnit** table.

Some documentation is found in the **docs** folder:

* **JsonSqlConfig.drawio.pdf** is a graphic data flow depiction
* **Repo setup.docx** has instructons for setting up the repo runtime environment
* **JsonSqlConfig.json** is just an example input for the web API Swagger interface

For running the solution use Visual Studio, SQL Server Management Studio and the EF Core CLI, see **Repo setup.docx**