Simio
Forward Thinking

Simio API Note: Custom Calculation Steps
July 2020 (Creation)

Contents

Overview

This API Note describes practical examples for creating and using the Simio User-Define Step. It is expected that these examples will grow over time. At present, the following are provided:

1. Interface with a SimioTable a Run-Time

2. Interface with SQL Server using EF (Entity Framework)

3.

For testing, sample models are provided, along with a section in this document to explain how the step works.

Example 1: Simio Table Interface Step

Some Background Information

Simio Tables are an important component of the Simio Data Driven model philosophy. However, for various reasons, such as performance and Simio architecture they do not exist at run-time as a matrix, but rather a collection of columns of various types, such as Properties, States, etc.

Accessing data in Simio tables does require some interfacing techniques that are not obvious. This example explores some of these techniques.

Model: Model1.spfx

VS Project: SimioTableInterfaceStep

Although there are several ways to interface with a Simio Table, this example demonstrates a preferred way if you are reading and writing to a State variable (and potentially reading Properties) from a Simio Table.

There trickiest part of this is to realize that a Repeating Group (RG) can be mapped to a Simio table, and then – using the GetRow() method of the RG reader, you can successively "map" the Repeating Group to whatever row of the table you desire.

So, the workflow is this:

In the Code:

In the DefineSchema method, create properties that you will use to map to the Simio table. Remember that you can only *write* to the Simio Tables' state columns.

The names do not have to match the table's column names since we will "map" these properties to the table columns in the Simio Project.

In the Simio Project:

Insert your Step into a Process. When you select it make sure you right click on the MyMappedFields, RightMouseClick and set the Referenced Property to Table1:
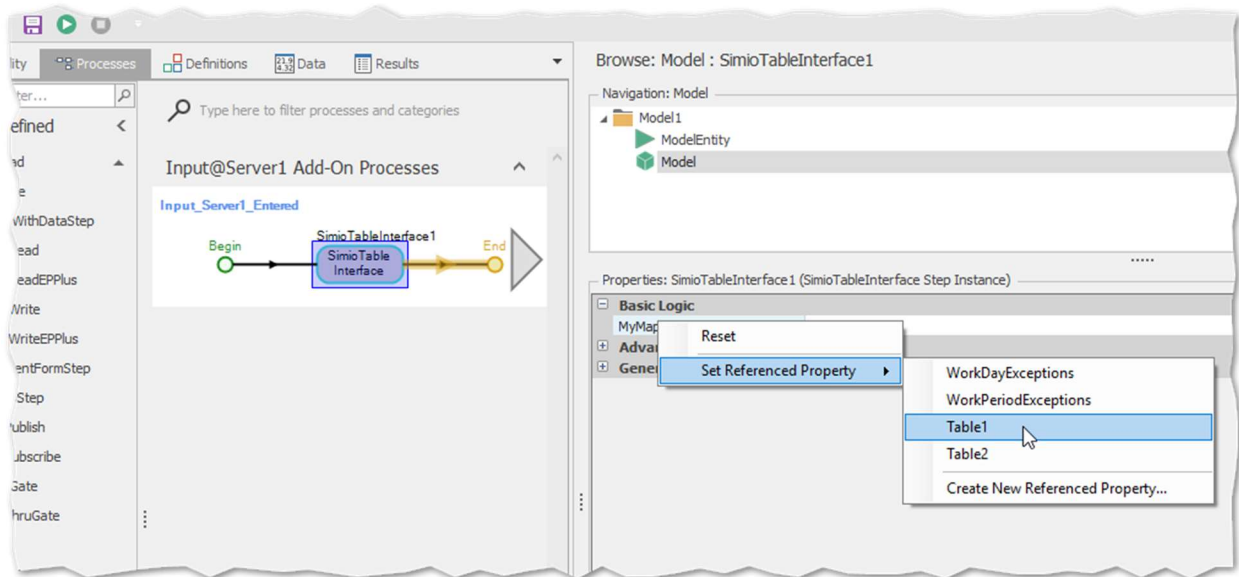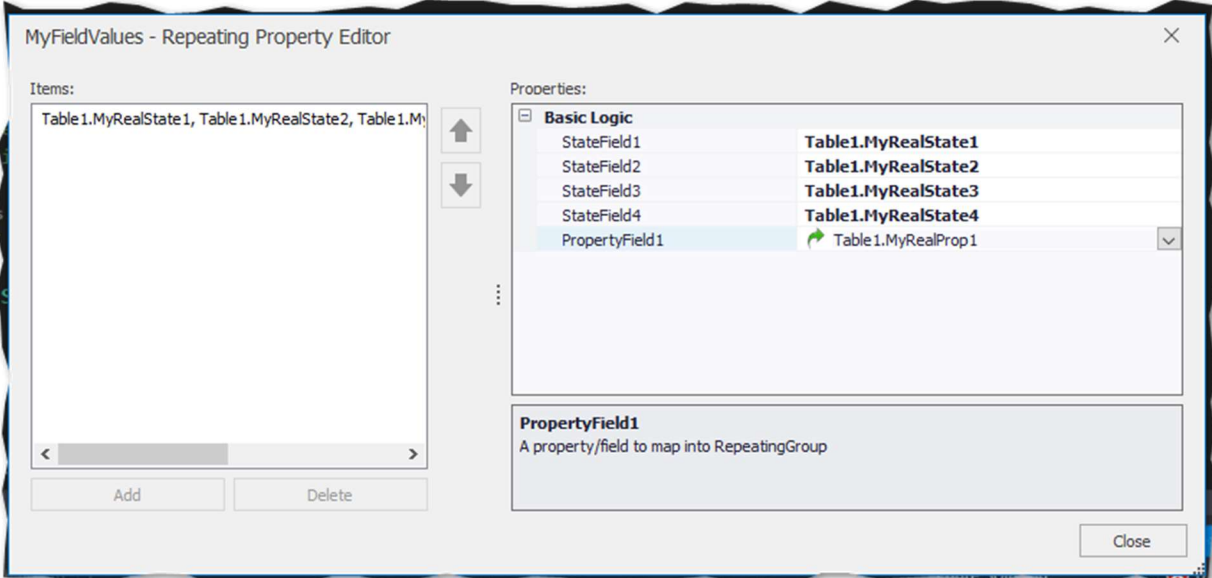
Note: If you do not see the curved green "reference" arrow, then this project **will not** work correctly!
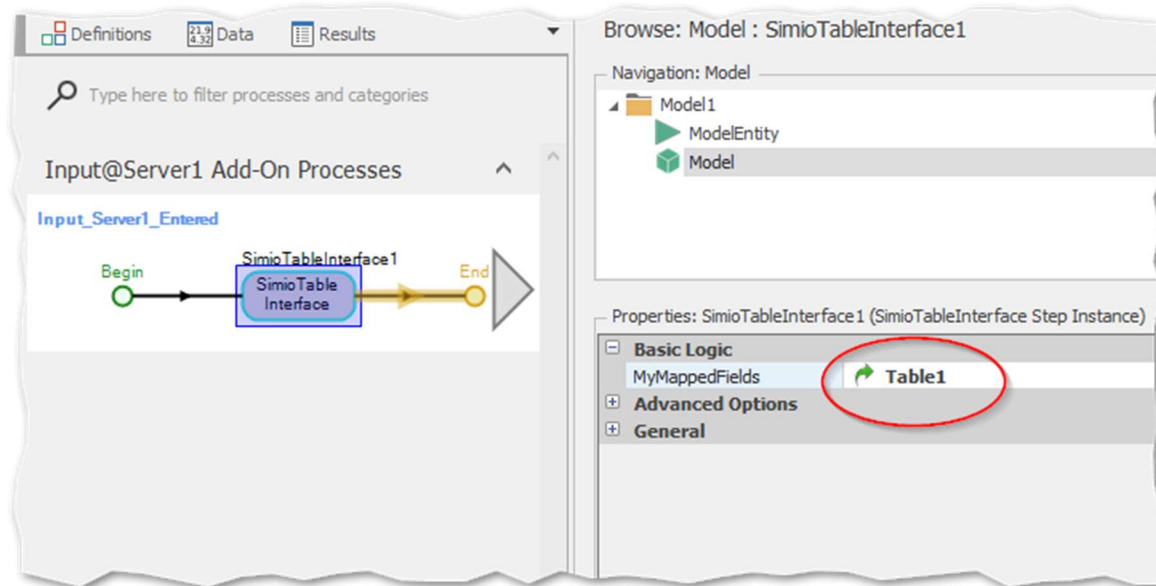
Then select the ellipses on the right to bring up the Repeating Property Editor and then enter the field mappings.

Note that here we are using the Simio generated field names (e.g. "StateField1"). In the code we will be using these names to access our data.

When you run the model, this is what you should see:

Simio
Forward Thinking

Notes on Use

Adding Logic

TroubleShooting

Example 2: Interfacing with SQL Server Using Entity Framework (EF) (In Progress)

Note: Currently under development

Some Background Information

Interfacing with relational data is perhaps the most common way to share information.

There are several good examples in the Simio literature for doing this is a simplistic fashion, but few that explore production-grade techniques that allow for a fast and secure way to do so.

That is the purpose for demonstrating how to do so with Entity Framework (EF).

All techniques have pro and con arguments, and EF is no exception. The biggest "con" may be that a supporting model must be built. Even so, the EF technique has become the foremost open source modern technique for interfacing to SQL Server.

The Model: Model2.spfx