

Entity Framework : How do you refresh the model when the db changes?

[Ask Question](#)

▲
52
▼
★
9

If you build the edmx file from the database and then the db changes, how do you get the model to pick up the change?

Do you delete the whole model and regenerate or can you detect changes?

I see a post mentioning a "Update Your Model" rt. mouse command on the edmx file, but I dont see it.

[Updating your edmx to reflect changes made in your db \(.net linq-to-entities\)](#)

I'm just starting out.

entity-framework

edited May 23 '17 at 12:26



Community ♦

1 1

asked Jun 1 '10 at 5:48



ChadD

10.1k 42 153 274

4 It is not on edmx file,

but in graphical interface. Open edmx in graphical editor by double left click and you'll find 'Update model' there. — [LukLed](#) Jun 1 '10 at 9:11

8 Answers



59



Are you looking at the designer or code view? You can force the designer to open by right clicking on your EDMX file and selecting `Open With -> ADO.NET Entity Data Model Designer`

Right click on the designer surface of the EDMX designer and click `Update Model From Database...`

All entities are refreshed by default, new entities are only added if you select them.

EDIT: If it is not refreshing well.

- Select all the tables and view-s in the EDMX designer.
- Delete them.
- Then, update model from database

edited Nov 11 '16 at 21:11



[Tomi](#)

2,175 1 8 19

answered Jun 2 '10 at 15:13



[jhappoldt](#)

1,531 13 23

I did not see the "Open With -> ADO.NET Entity Data Model Designer" option when right mouse clicking on the edmx file, but the other option was available. I'd like to know why, but you provided a valid answer. ty. –

[ChadD](#) Jun 2 '10 at 15:33

right click edmx file and choose open with then choose data model designer –

[Mehmet Eren Yener](#)
Feb 11 '14 at 15:07

4 Is it possible to do this programmatically? I would like to incorporate this as part of my build process. – [GARRY](#)
Oct 5 '14 at 20:28

1 Just changed ForeignKeys. They didn't refresh. Had to delete table from the model, and add again. – [Tomi](#)
Oct 19 '16 at 21:50

if you find the update doesn't work as expected can see the reference ---

– stackoverflow.com/questions/15545469/... – [yu yang Jian](#)
Mar 16 '17 at 8:54



I have found the

17

designer "update from database" can only handle small changes. If you have deleted tables, changed foreign keys or (gasp) changed the signature of a stored procedure with a function mapping, you will eventually get to such a messed up state you will have to either delete all the entities and "add from database" or simply delete the edmx resource and start over.

answered Sep 1 '10 at 6:34



Roman

171 1 2

The 'update from database' wizard replaces the SSDL but only applies some changes to the CSDL. As a result, when you have more complex changes the SSDL and CSDL tend to get out of sync, causing the 'messed up state' you're seeing. I have an add-in that addresses this issue for the EF4 designer in VS2010; it compares each of the three layers (DB, SSDL, CSDL), shows all differences and allows you to selectively sync up diffs. You can download the add-in and get a trial license at huagati.com/edmxtools if you want to try it out. – KristoferA
Sep 2 '10 at 2:44



This might help you guys.(I've applied this to my Projects)



Here's the 3 easy steps.

1. Go to your **Solution Explorer**. Look for **.edmx** file *(Usually found on root level)*
2. Open that .edmx file, a *Model Diagram* window appears. Right click anywhere on that window and select "**Update Model from Database**". An *Update Wizard* window appears. Click Finish to update your model.
3. Save that .edmx file.

That's it. It will sync/refresh your Model base on the changes on your database.

For detailed instructions. Please visit the link below.

[EF Database First with ASP.NET MVC: Changing the Database](#) and updating its model.

edited Jun 10 '17 at 4:12

answered Jun 9 '17 at 13:20



[Bim Garcia](#)

49 2

Noted. Thanks for advice. – [Bim Garcia](#)
Jun 10 '17 at 3:19



1



Update CodeFirst Model is not possible automatically. I don't recommend either. Because one of the benefits of code first is you can work with POCO classes. If you changed this POCO classes you don't want some auto generated code to destroy your work.

But you can create some template solution add your updated/added entity to the new model. then collect and move your new cs file to your working project. this way you will not have a conflict if it is a new entity you can simply adding related cs file to the existing project. if it is an update just add a new property from the file. If you just adding some couple of columns to one or two of your tables you can manually add them to your POCO class you don't need any extra works and that is the beauty of Working with Code-First and POCO classes.

answered Jul 26 '17 at 23:13



[Navid Golforoushan](#)

368 5 12



I don't know why it doesn't update automatically but after you build the solution, whatever model classes or templates you deleted or modified will be updated from the database once you **"Rebuild the solution"**.

It works on my Visual Studio 2015!

answered Mar 7 '18 at 13:43



[Willy David Jr](#)

4,083 2 17 27



I just had to update an .edmx model. The model/Run Custom Tool option was not refreshing the fields for me, but once I had the graphical designer open, I was able to manually rename the fields.

answered Mar 21 '14 at 19:39



[TedC](#)

41 2



Here:

1. Delete the Tables from the EDMX designer

2. Rebuild Project/SLN (this will clear the model class)
3. Update Model from Database(read all the tables you want)
4. Rebuild project/SLN (this will recreate your model class including the new columns)

answered Aug 29 '18 at 14:04



CyberNinja

500 4 14



I've been working on a project, not too large, that incorporates Entity Framework, about a dozen tables, and about 15 stored procs and functions. After weeks of development, attempting to refresh my tables and stored procs has yielded mixed results as far as successfully updating the model. Sometimes the changes are effective, most times they are not. Simple column changes (changing order, adding, removing, or renaming) sometimes works, most times does not. Visual Studio seems to have more problems with refreshing than just adding new. It also

exhibits more problems with stored proc changes not being reflected, especially when columns are added or deleted or renamed. I have not detected any consistent behavior so i can't say "This type of change will always be updated and this type of change will not".

End result, if you want 100% effective solution, delete the EDMX file from the project, "Add new" item to project (ADO.NET Entity Data Model), and make sure you use the same name for the Model Name. This works every time.

answered May 30 '18 at 16:33



[Joseph Mineo](#)

1