# RSM tips & tricks

Created by Unknown User (joshs@amdocs.com)

## RSM - Hide Attributes in Diagram

In addition to the class attributes that the Information Framework contains, RSM adds attributes for each navigable association to the source class (the source of the navigation). In case of bi-directional navigable association the attributes are added to both associated classes. As a convention in the Information Framework all these attribute names start with an underscore '_' so that it is easy to distinguish between these attributes and "business oriented" attributes.

When a class appears in the diagram and the associated class does not, these attribute may appear in the source class attribute list such as in the following diagram:
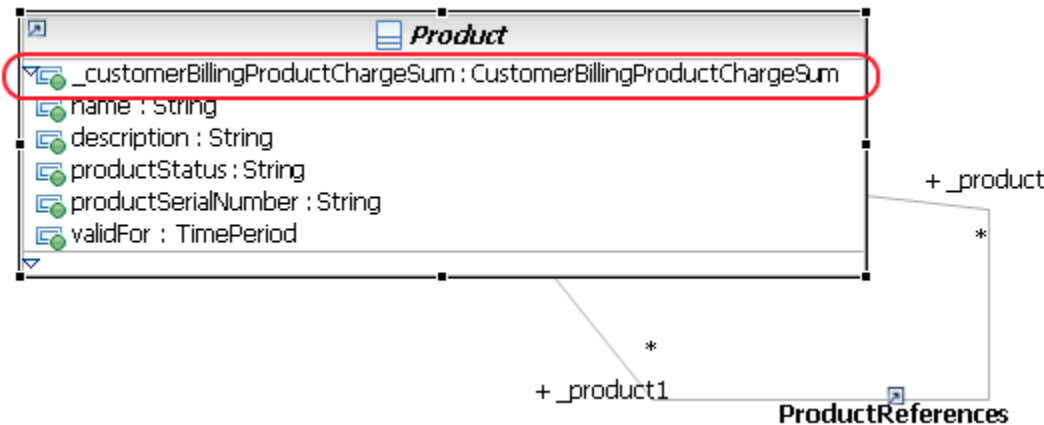


As can be seen these attributes typically have long names and they clutter the diagram and screen space. In many cases you want to hide these attributes since they do not add any information in the context of the diagram. Hiding attributes is possible but nor intuitive:
Right click on the class and select *Filters | Sort/Filter Compartment Items...*
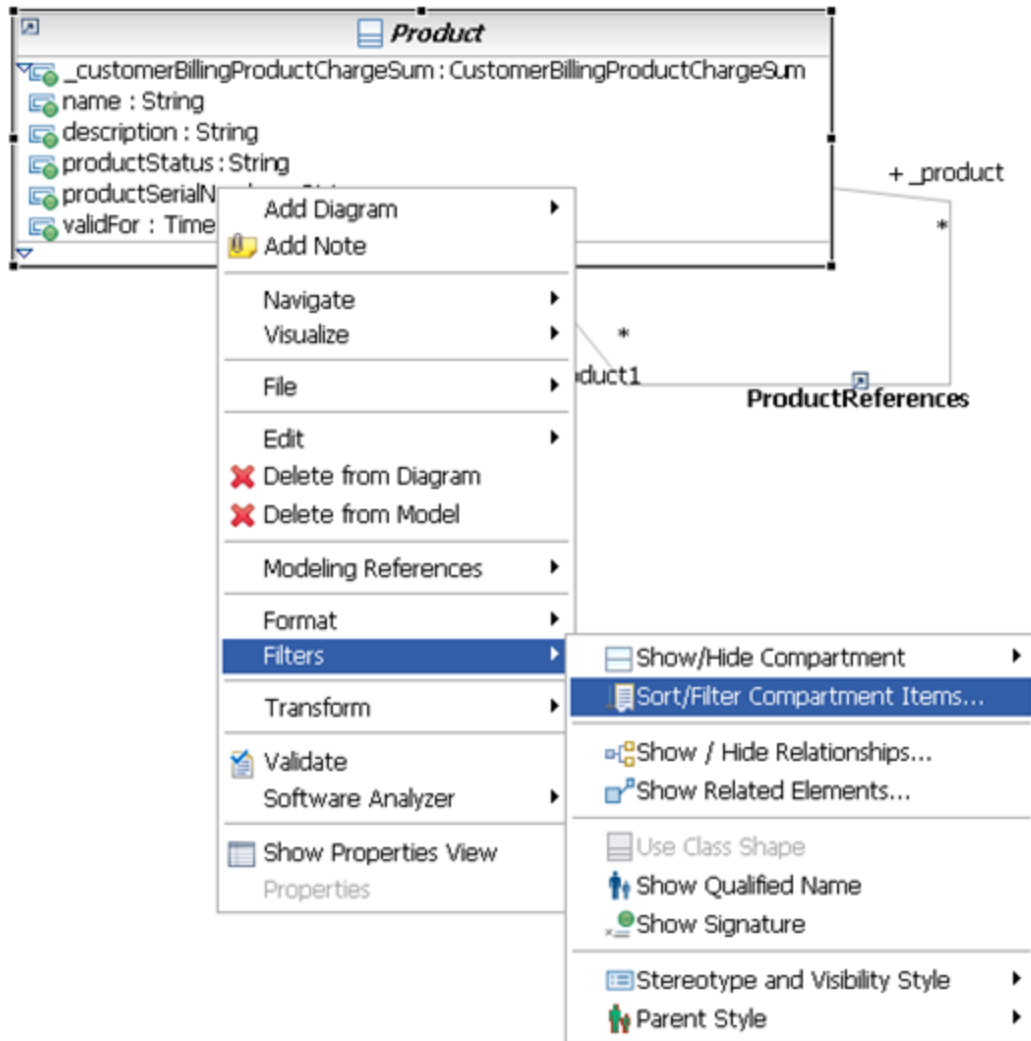
Then a in the dialog that pops up select attribute on the left tree control and unmark the attributes you want to hide.
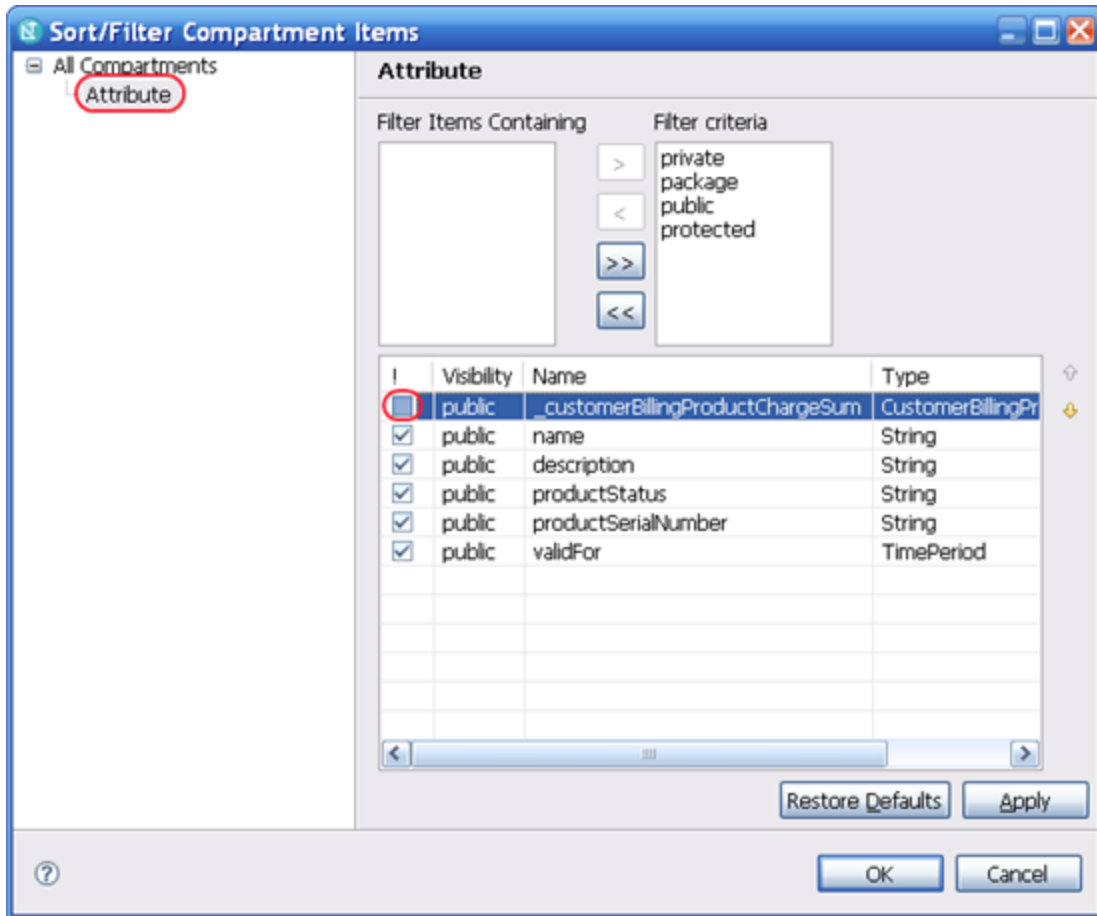
This will hide the attribute and now the same class will look like:
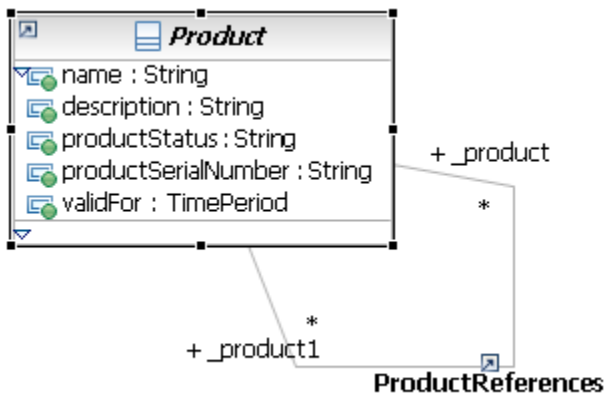


# RSM - Extending SID

There are many reasons you may want to extend the SID model without touching the original, among them:

- You extend the model for your own use, and you want to be able to easily upgrade to a new version of the core model without losing your changes.
- You are planning a contribution and you want to do all the work without impacting the model until the contribution is team approved. Then you want to easily deploy your changes to the model without the need to retype all the information.

**The technique described below should be used only when you <u>add</u> entities and associations to SID and should be avoided when <u>changing</u> existing SID entities.** Look in GB922-1U on how to extend the SID without changing the core entities.
 RSM supports the above requirements (and others) by using project dependencies. The way to do this in RSM is to import the SID model to RSM (either from a model release for the first use case, or possibly from SVN for the second use case). and create a workspace with the 2 models - SID model_profiles and SID Phase <> Models. This is how it looks in RSM Project Explorer (for a project which is connected to SVN)

blocked URL

Then create a new model project using the wizard, and in the last step of the wizard (referenced Projects) select your existing SID project.

blocked URL

Now you can add your own classes in the new model and drag existing SID classes to the diagram so that your new classes are associated with existing SID entities. In the new model the diagrams look like:

blocked URL

Note the reference icon which specifies entities from the original SID model. If you create a new diagram in the SID model (this is just to explain RSM capabilities, changing the SID is not recommended!) and drag to it the Customer and MyCustomerExtension classes it will look like:

blocked URL

If you plan on using SID based model with your extensions you are done here and can continue working with your model. If you plan on moving your changes to the core SID in a later phase, the only thing you need to do (after creating SVN project and locking all the files) is drag all the new extension entities to their correct place in the SID model. RSM takes care of the rest.

After doing so the diagrams is the SID project looks like:

blocked URL

And the diagram in the extension model looks like

blocked URL