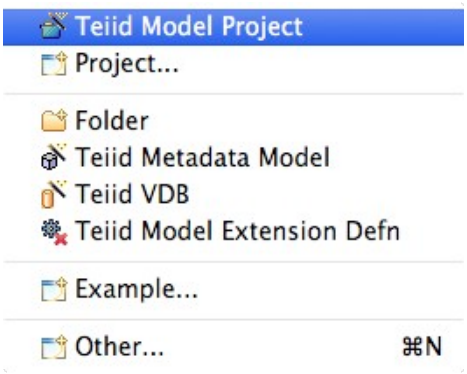
Lab 4. Create a Virtual Base Layer

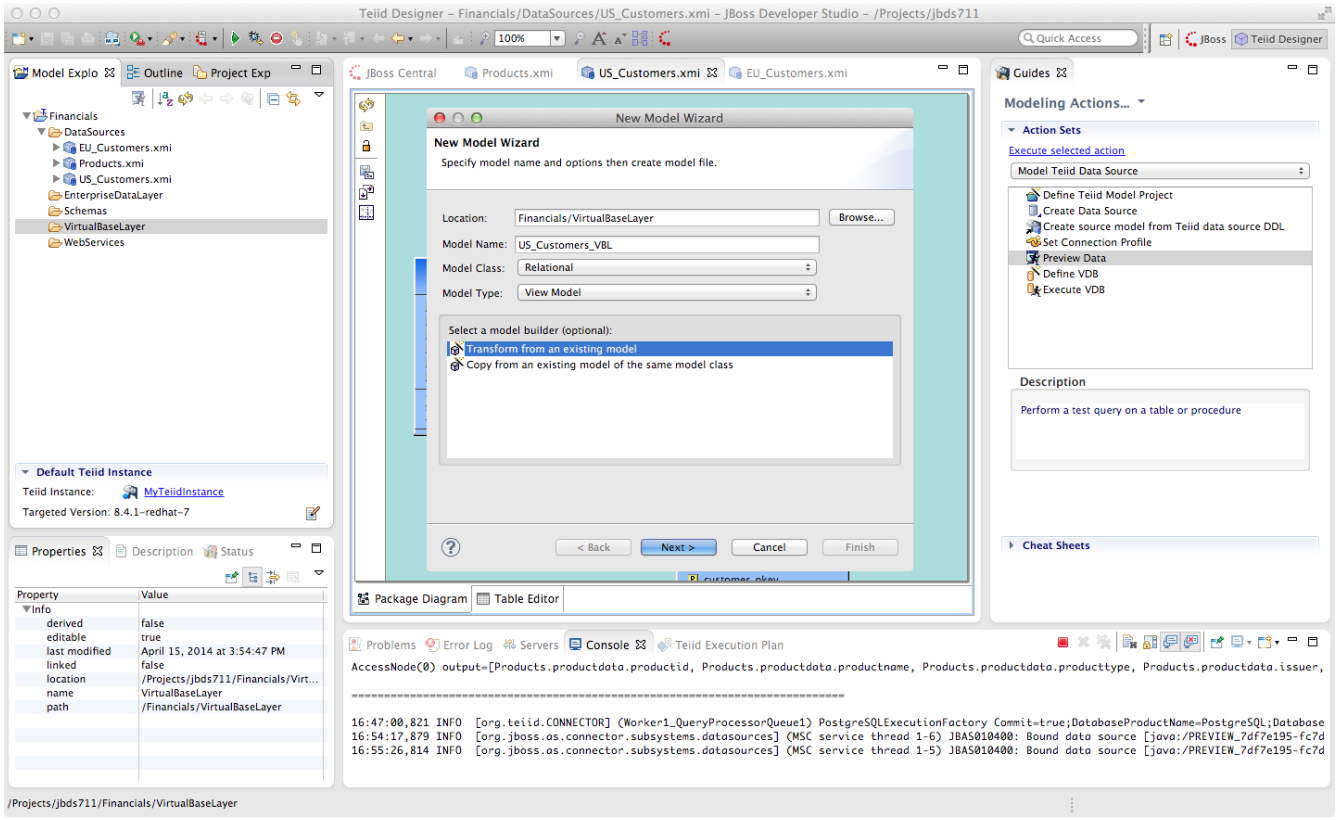
Beyond the obvious advantages of integrating disparate data sources using an intuitive, easy-to-use technology such as JBoss Data Virtualization, another significant advantage of creating a data services abstraction layer is to provide a level of isolation from the physical sources themselves. By creating a layered set of data service models, from fine-grained to more granular data services, the developer can build a stable data integration layer than can easily adapt to changes in the source systems. This is especially advantageous when the consumers of the data are separate from/have no control of the providers of the data. A recommended best practice to begin building this "future-proof" abstraction layer is to create a Virtual Base Layer (VBL), a one-to-one mapping of each physical source that isolates any future changes that may arise in the data source(s) to a specific transformation in the model. These VBL components can then be used as building blocks for higher level services; all of the transformations built on top of them will not need to be changed should the need arise to accommodate a change in the source(s).

4.1 Create a US_Customers_VBL

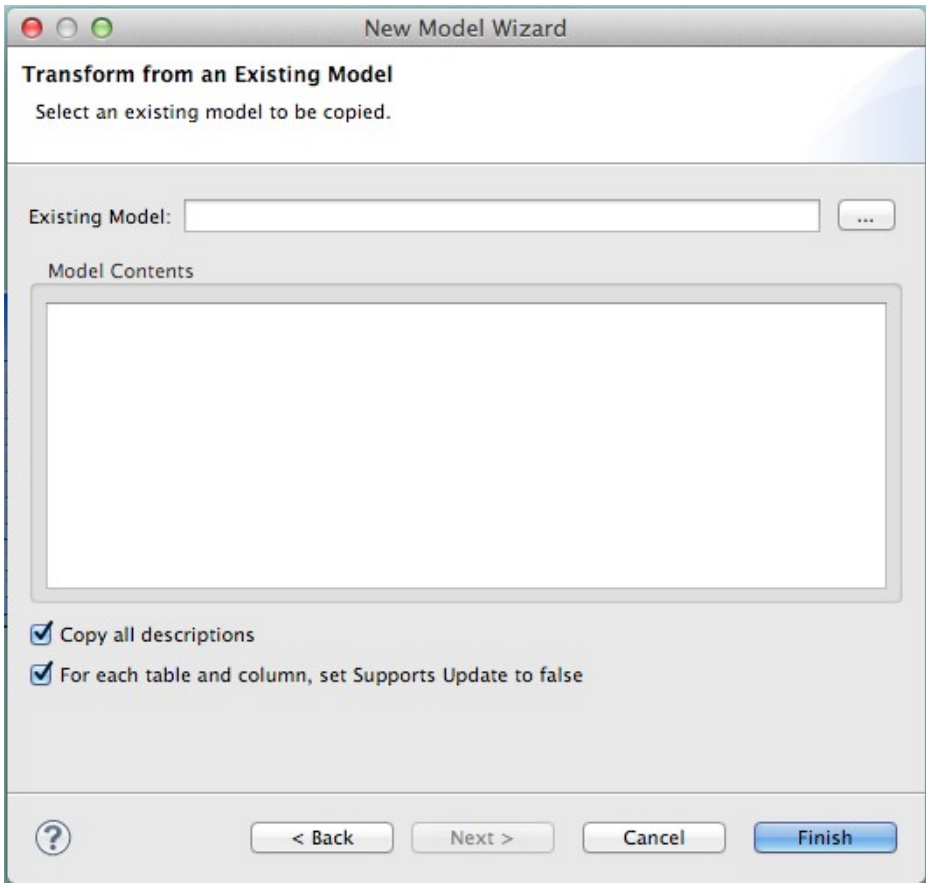
To create a VBL for each of the source metadata models that you have imported, right-click on the VirtualBaseLayer folder that you created earlier and select New → Teiid Metadata Model.



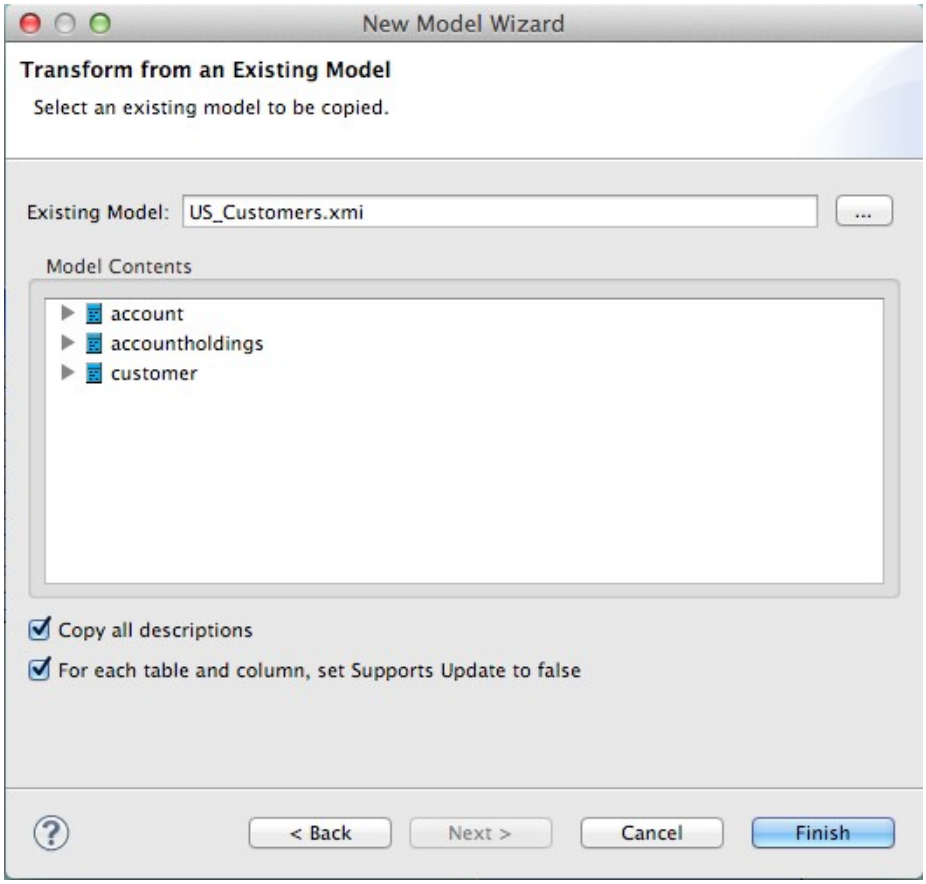
Enter US_Customers_VBL as the Model Name, Relational as the Model Class, and View Model as the Model Type. Select “Transform from an existing model” in the Select a model builder panel and click Next.



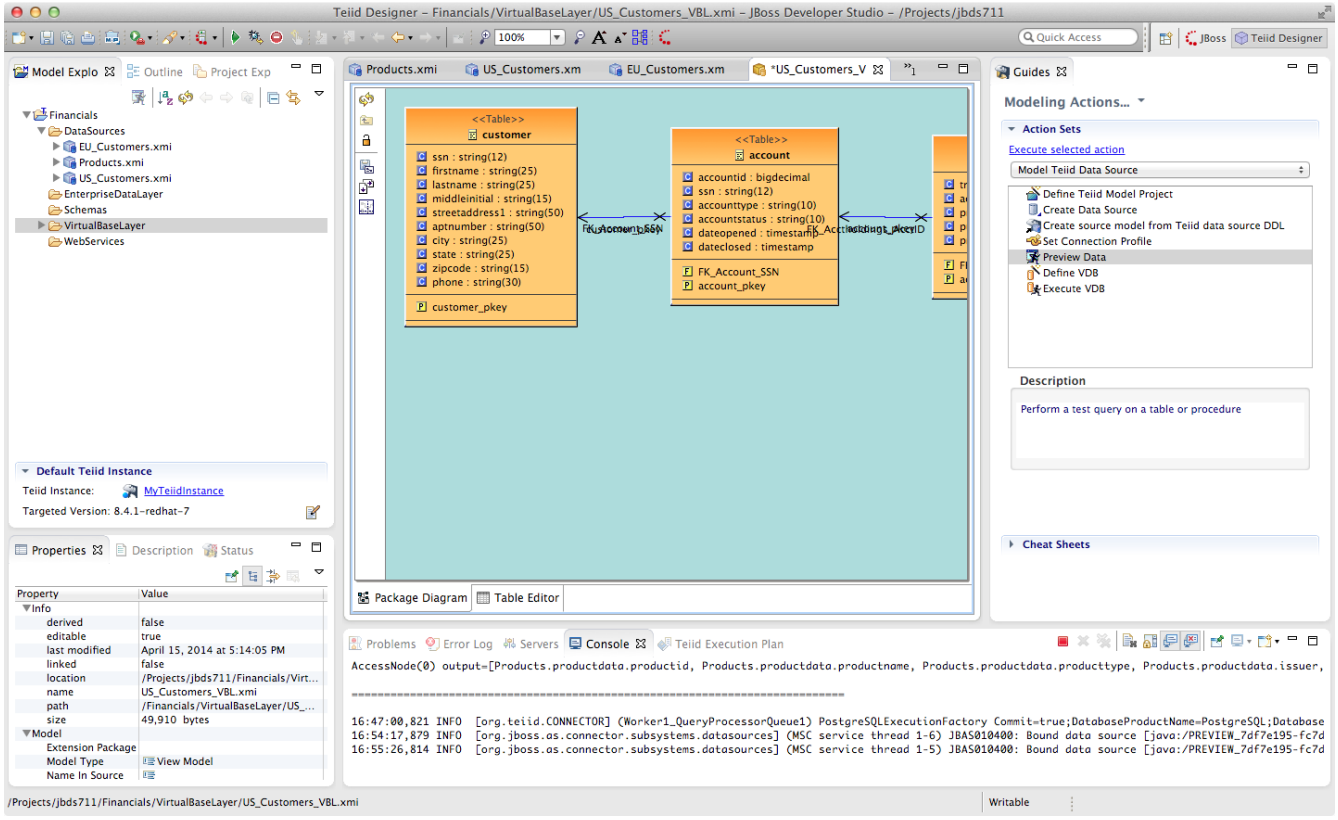
The “Transform from an Existing Model” wizard will appear.



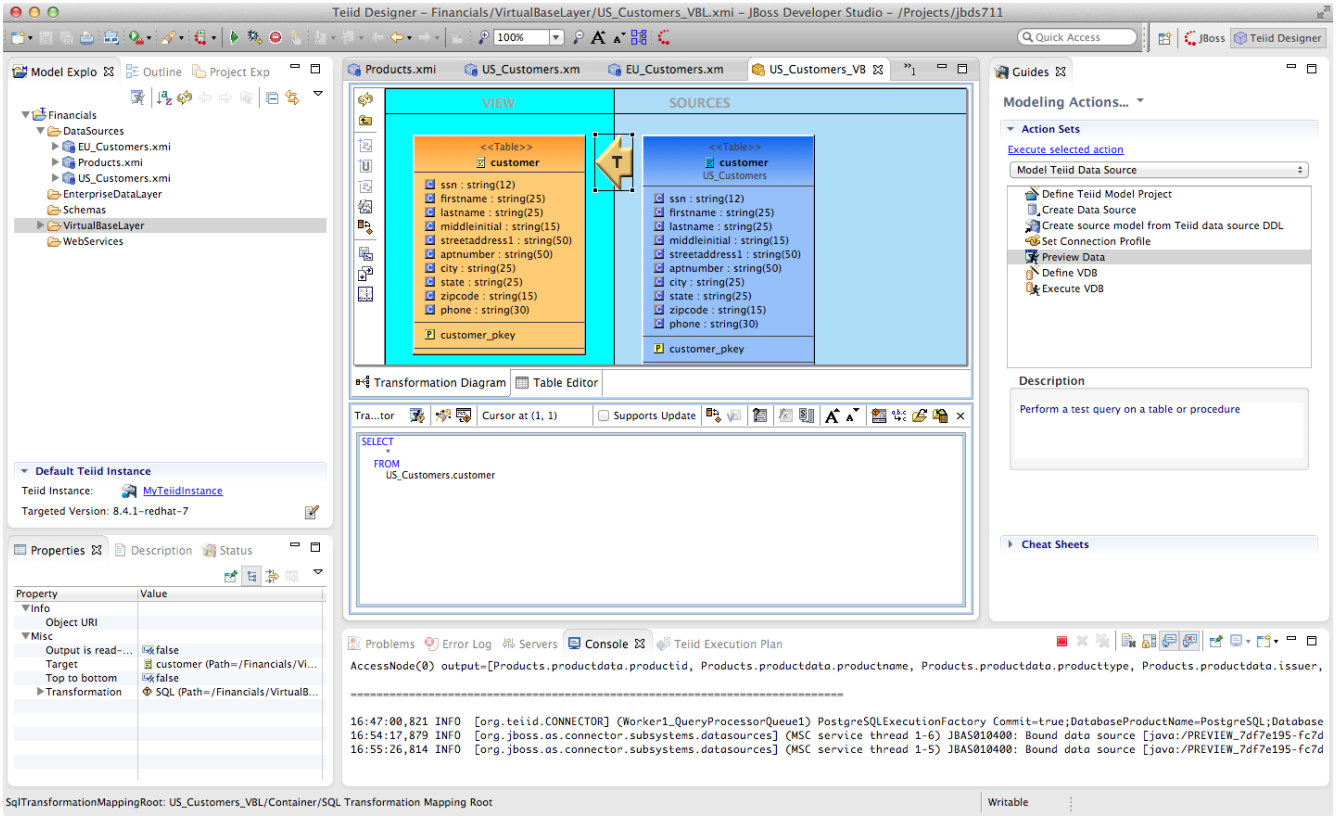
Select the chooser icon (...) to the right of Existing Model and open Financials → DataSources → US_Customers. Click “OK”. You will be returned to the New Model Wizard. We are going to copy all model elements so simply click “Finish” on the screen as indicated below.



The Package Diagram and Model Explorer for the US_Customers_VBL virtual model will now open in the workspace. Observe that the virtual models are rendered in yellow whereas physical models are rendered in blue. Save the changes to the project (do this periodically as you progress) and note that you can also do previews of the virtual model.



Double-click on the customer table in the US_Customer_VBL model. This will open the Transformation Editor.



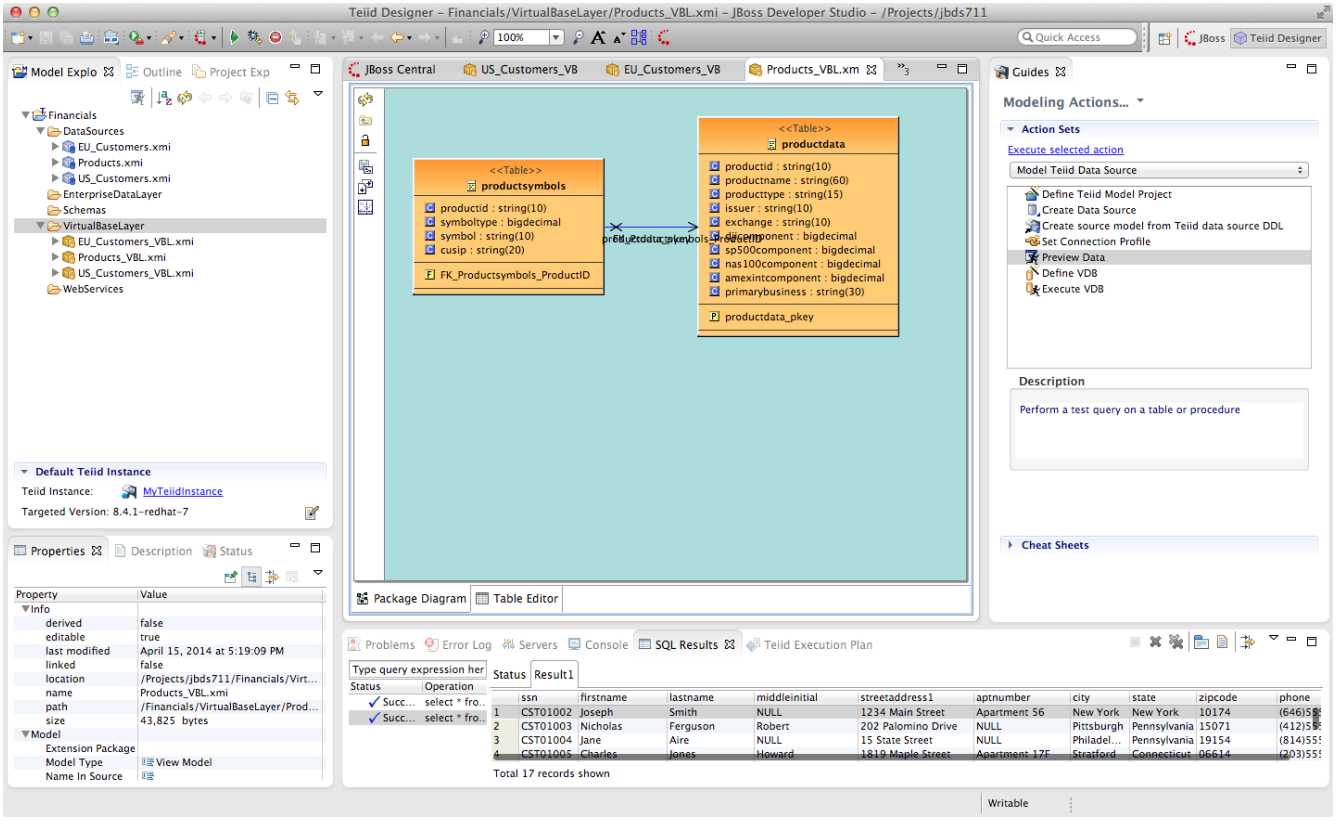
Note that the transformation has already been created for you, and as noted above it is a simple one-to-one mapping of the underlying physical source. By building up the data services in layers like this, it allows the designer to keep the transformational logic for each view fairly simple, and complex data transformations are achieved by uses several layers of such views. Now, in a traditional relational database, such a design would have a fairly heavy performance penalty at runtime to deal with all of these layers of views, which is why in a traditional database you'd see a use case like this defined as a single view defined with a very lengthy and complex SQL statement. The reason this is not an issue with JBoss Data Virtualization is because the Query Engine in the JBoss Data Virtualization Server compresses all of these layers at run time down to a single (potentially highly complex) query that is then optimized to run (in parallel if possible) against the backend data sources. Thus there is no penalty to using layered views with JBoss Data Virtualization.

Note that the "transformation language" in the Transformation Editor is ANSI-standard SQL. We don't make you learn a new language to define transformations, you just use normal SQL syntax, which is something that we find most data architects are already fairly comfortable with. We will be spending much more time with the Transformation Editor further on in the lab.

4.2 Create the EU_Customers_VBL and Products_VBL Models

Create VBLs for the other two physical models you have imported (Products_VBL and EU_Customers_VBL).

Be sure to create them within the VirtualBaseLayer folder. Right-clicking on the folder to get to the New → Teiid Metadata Model wizard. Save your changes. When you are finished, your project should look like the illustration below.



Congratulations, you have completed Lab #4.