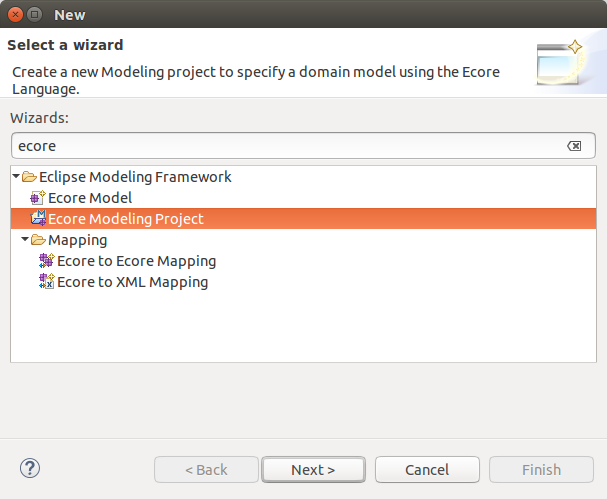
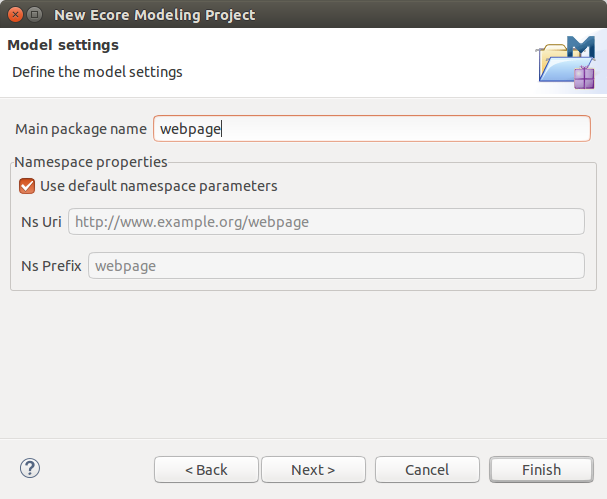
**Define a new EMF model and create Java code from it**

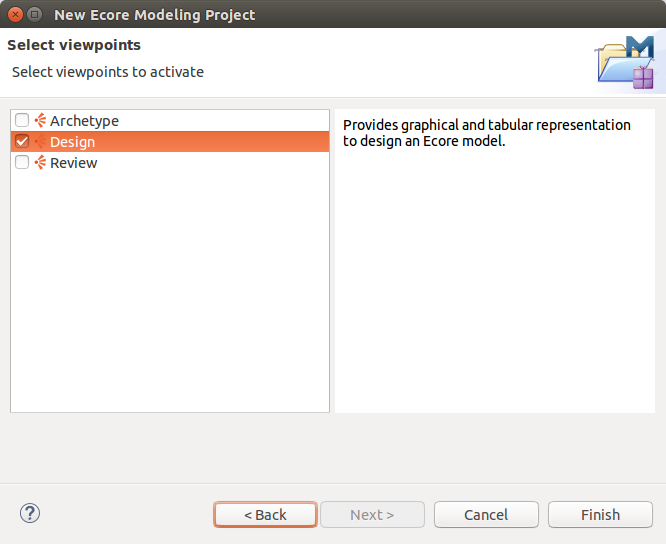
**3.1. Project and initial model creation**

Create a new project called *com.vogella.emf.webpage.model* via **File**  **New**  **Project…​**  **Ecore Modeling Project**.

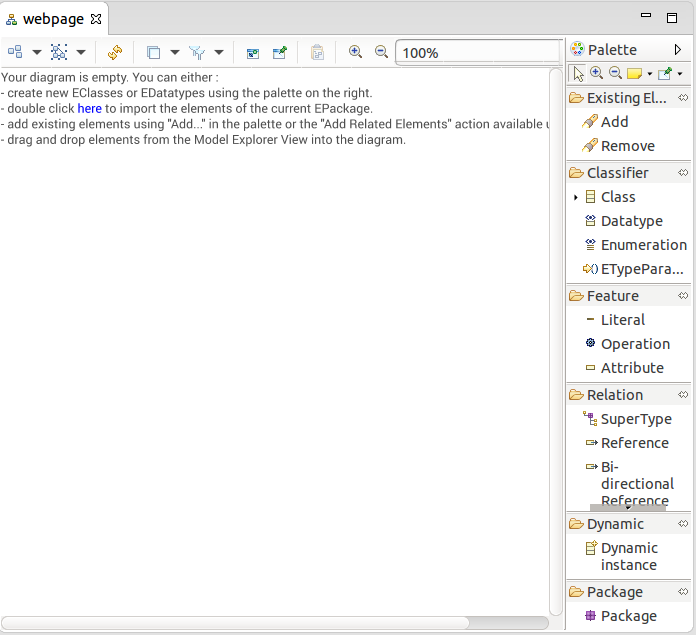


Enter *webpage.ecore* as the *Domain File Name* parameter.



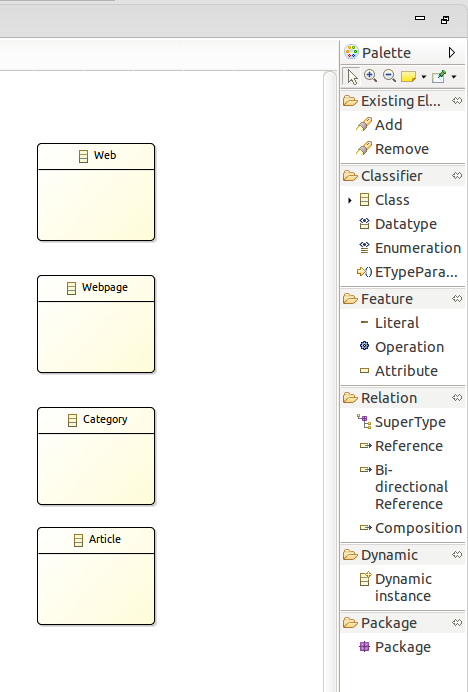


This should open a visual editor for creating EMF models.

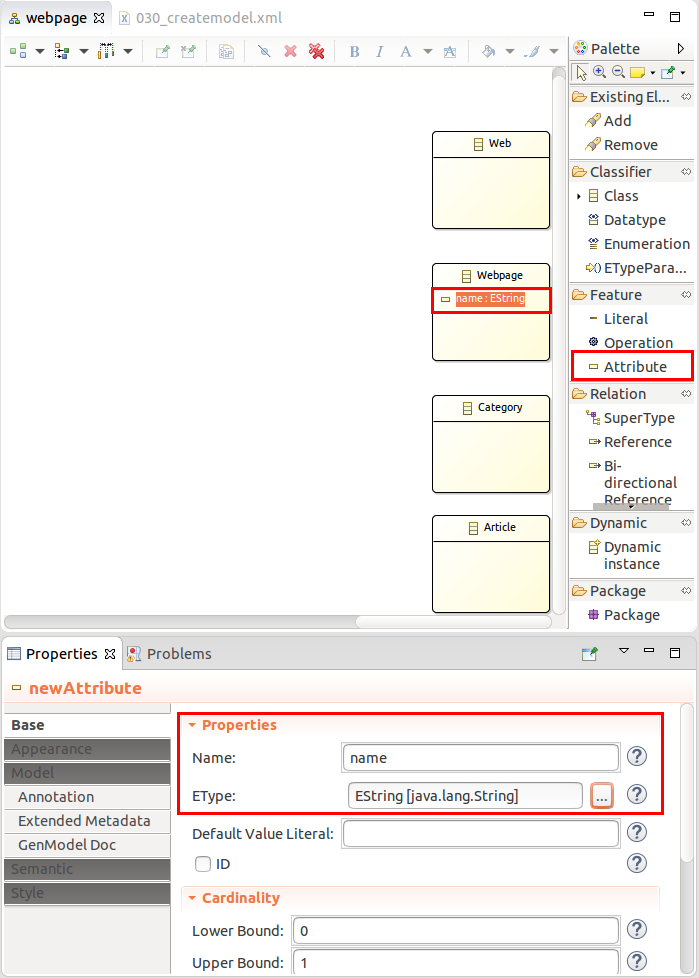


Open the *Properties* view via the menu **Window**  **Show View**  **Other…​**  **Properties**. This view allows you to modify the attributes of your model elements.

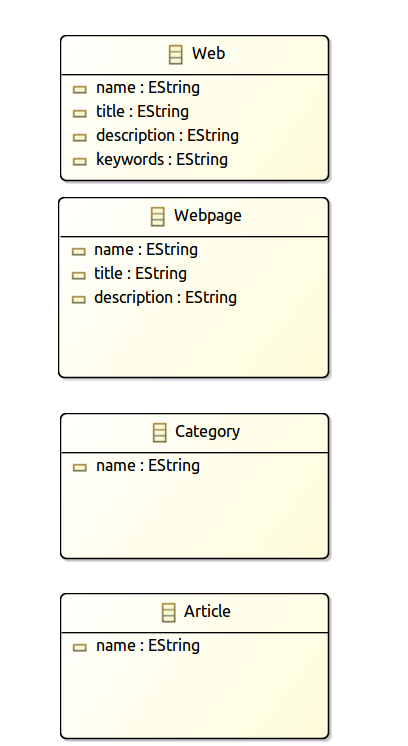
Click on *Class* and click into the editor to create a new class. Create the MyWeb, Webpage, Category and Article EClasses.



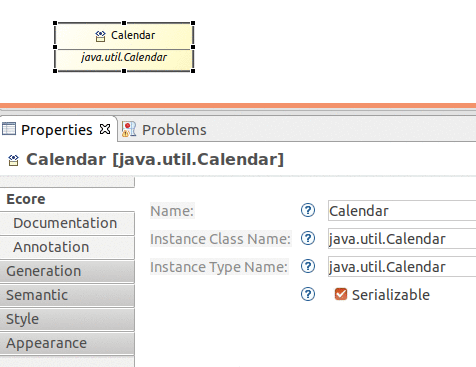
Use the *Attribute* node to assign the attribute called *name* to each object. This attribute should have the EString type.



Add the *title*, *description*, and *keywords* attributes to the *Web* and *Webpage* model elements.

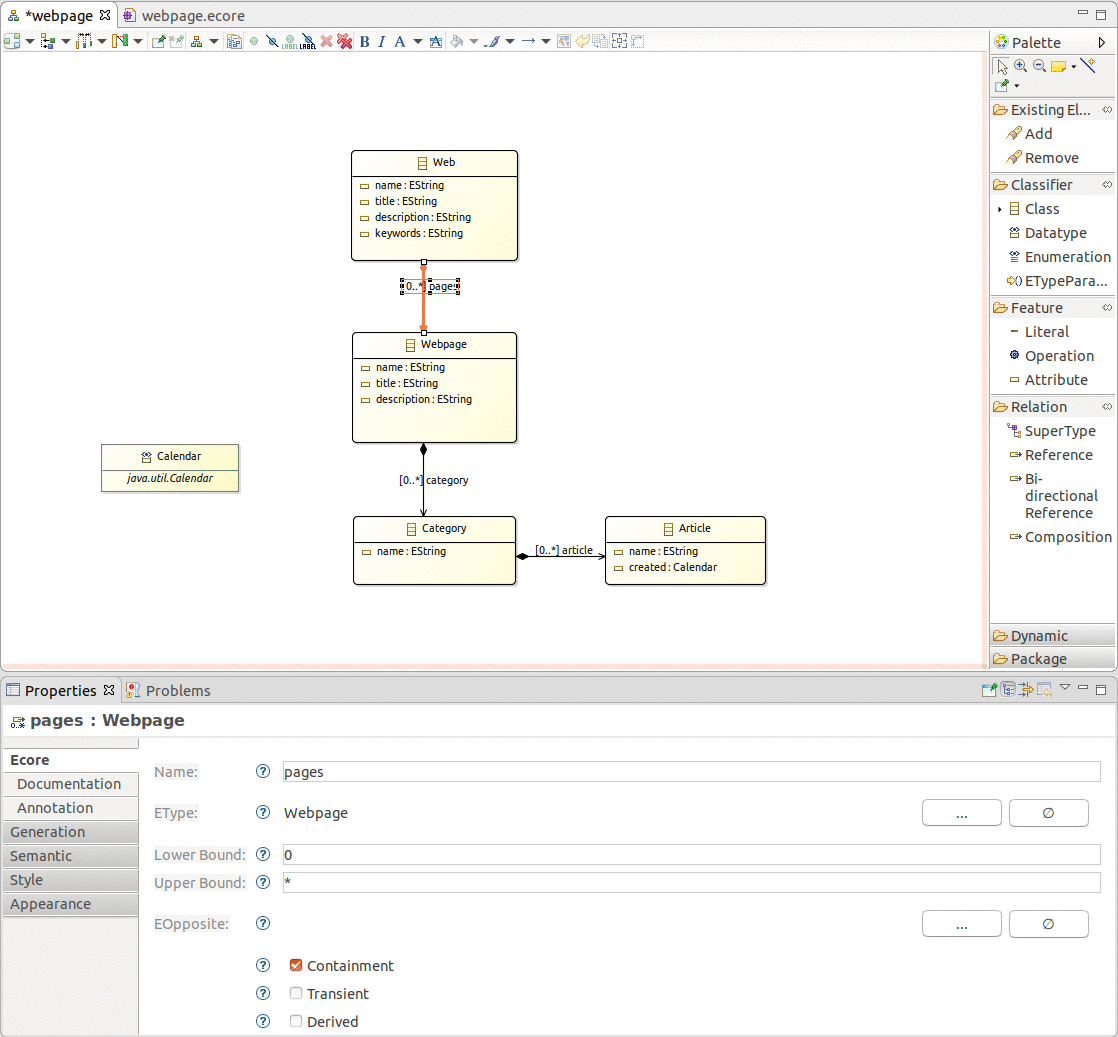


We want to use the data type calendar in our model. Select Datatype and drag it into your model. Assign the name *Calendar* to it. Use java.util.Calendar as type parameter.



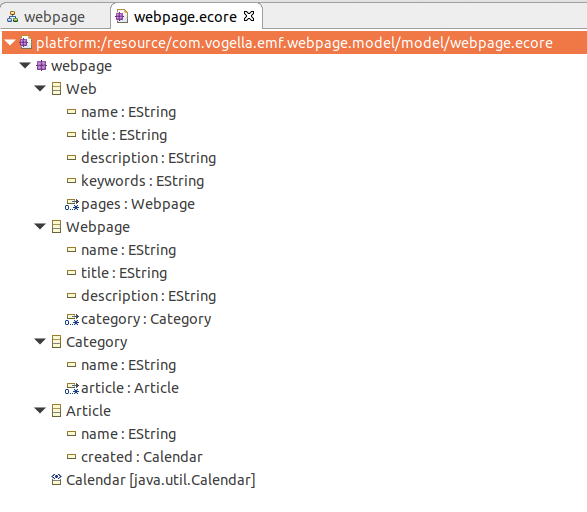
Add a new Attribute called *created* to Article and use the new type *Calendar*.

Select *References* and create an arrow similar to the following picture. Make sure the upper bound is set to \* and that the Containment property is flagged.



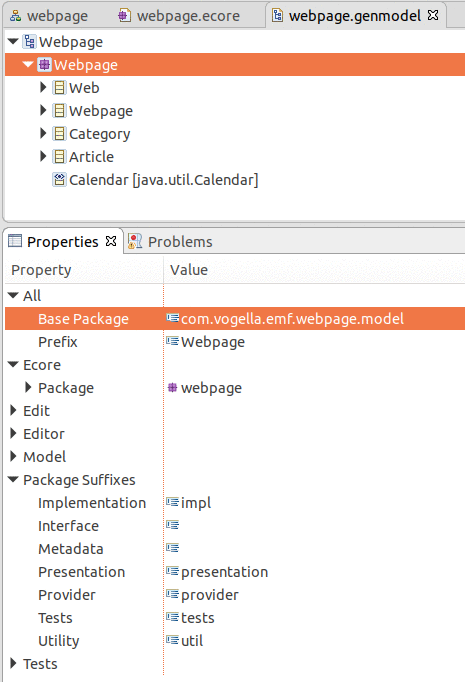
**3.2. View Ecore diagram**

Close the diagram and open the *webpage.ecore* file. The result should look like the following screenshot.



**Set the package**

Open the *webpage.genmodel* and select the *Webpage* node. Set the base package property to *com.vogella.emf.webpage.model*.

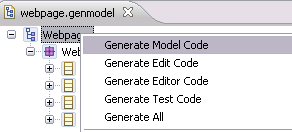


**Generating the domain classes**

**Generating Java code**

Based on the *.genmodel* files, you can generate Java code.

Right-click on the root node of the *.genmodel* file and select *Generate Model Code*. This creates the Java implementation of the EMF model in the current project.



**Review the generated code**

The generated code will consist of the following:

* *model.webpage* — Interfaces and the Factory to create the Java classes
* *model.webpage.impl* — Concrete implementation of the interfaces defined in model
* *model.webpage.util* — The AdapterFactory

The central factory has methods for creating all defined objects via createObjectName() methods.

For each attribute the generated interface and its implementation contain getter and (if allowed in the model definition) setter methods. Each setter also has a generated notification to observers of the model. This means that other objects can attach them to the model and react to changes in the model.