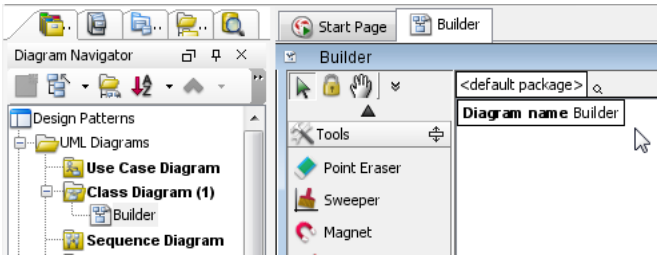
Builder Pattern Tutorial

Written Date : September 28, 2009

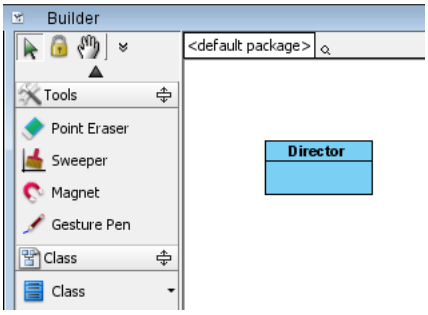
Modeling a Design Pattern with a Class Diagram

1. Create a new project named *Design Patterns*.

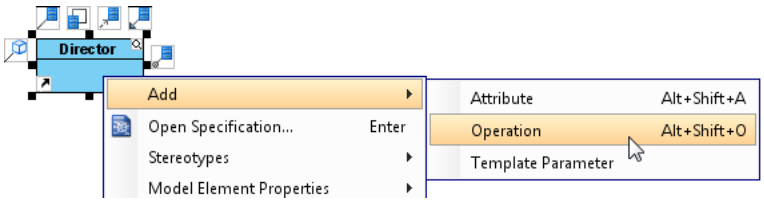
2. Create a class diagram named *Builder*.



3. Select **Class** from the diagram toolbar. Click on the diagram to create a class. Name it *Director*.

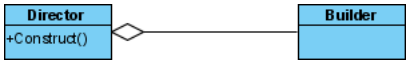


4. Right-click on *Director* and select **Add > Operation** from the popup menu.

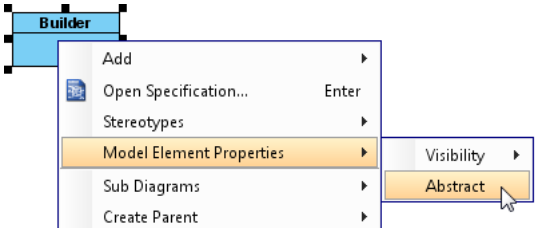


5. Name it *Construct()*.

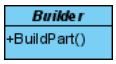
6. Move the cursor over *Director* and use the resource icon **Aggregation > Class** to create an associated class named *Builder*.



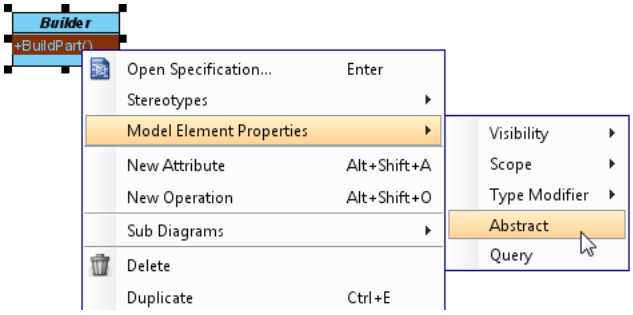
7. Set the *Builder* class as abstract by right-clicking on it and selecting **Model Element Properties > Abstract** from the popup menu.



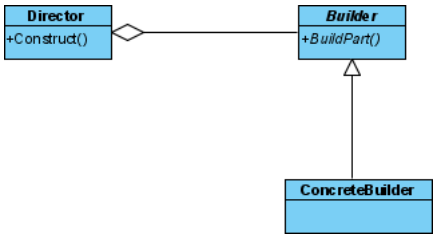
8. Create an operation named *BuildPart()* in the *Builder* class.



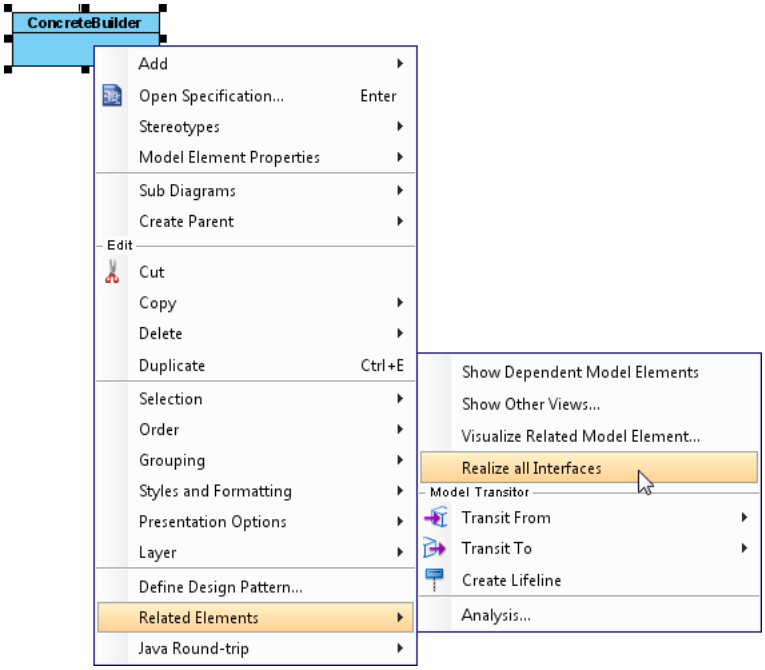
9. Set *BuildPart()* as abstract by right-clicking on it and selecting **Model Element Properties > Abstract** from the popup menu.



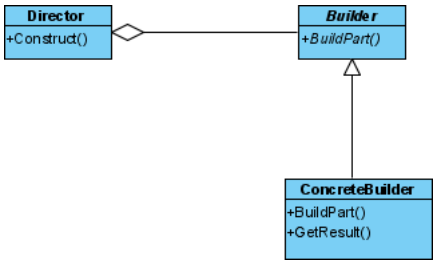
10. Move the cursor over *Builder* and use the resource icon **Generalization > Class** to create a subclass named *ConcreteBuilder*.



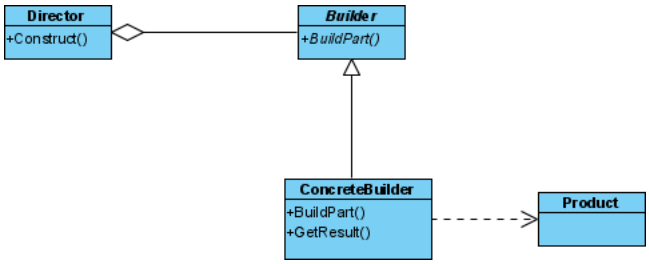
11. Inherit operations from *Builder* by right-clicking on *ConcreteBuilder* and selecting **Related Elements > Realize all Interfaces** from the popup menu.



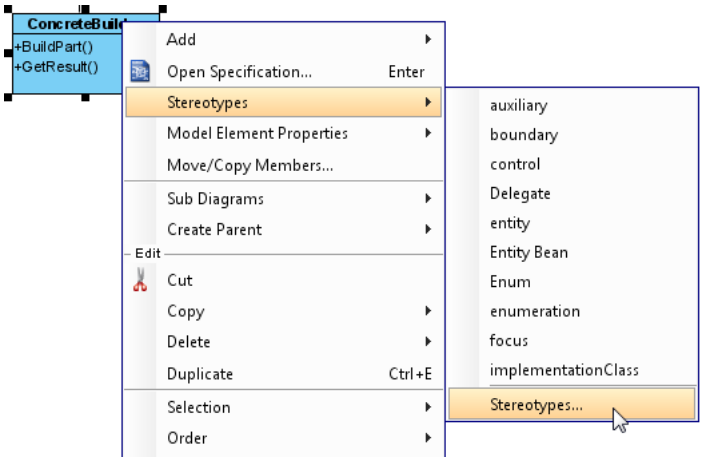
12. Add an operation named *GetResult()* to *ConcreteBuilder*. Up to now, the diagram should look like this:



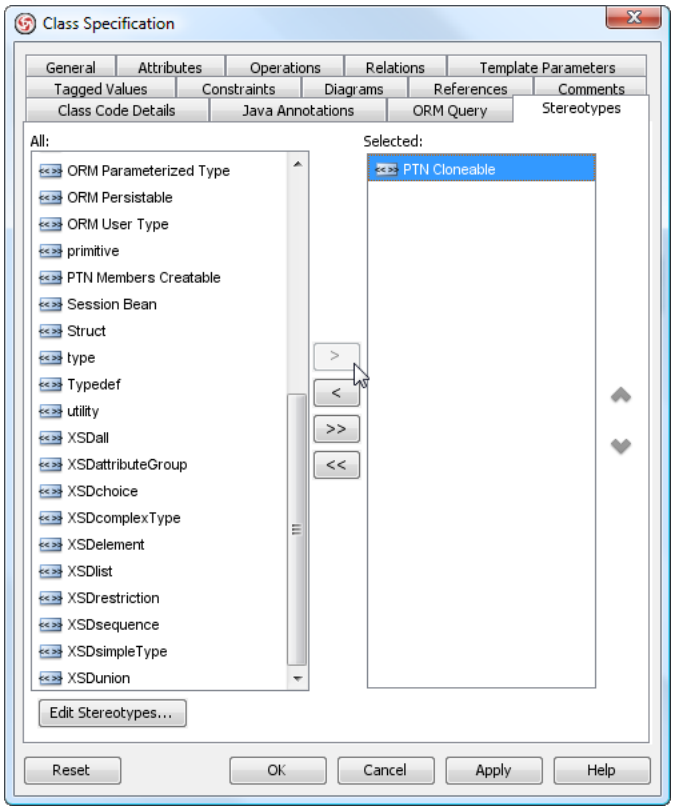
13. Move the cursor over *ConcreteBuilder* and use the resource icon **Dependency > Class** to create a depending class named *Product*.



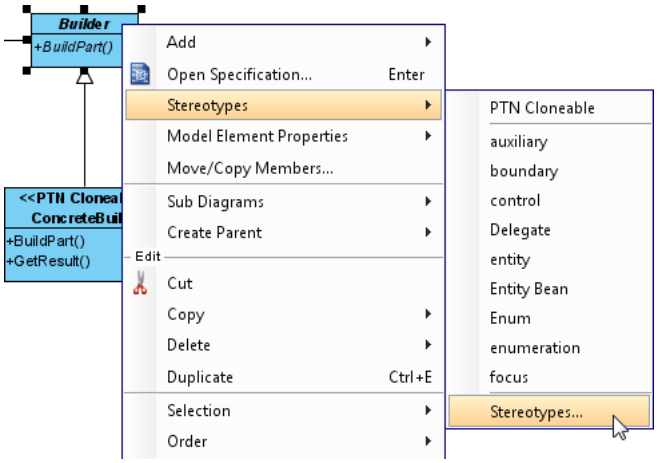
14. In practice, there can be multiple concrete builder classes. To represent this, you need to assign a special stereotype to the *ConcreteBuilder* class. Right-click on *ConcreteBuilder* and select **Stereotypes > Stereotypes...** from the popup menu.



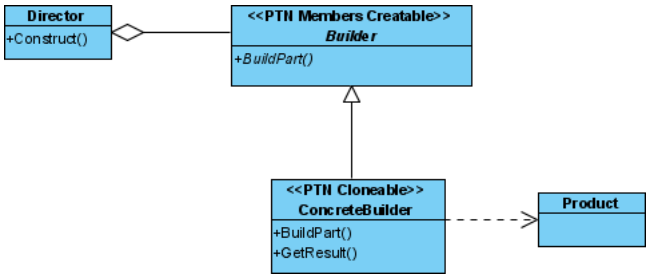
15. Select **PTN Cloneable** and click > to assign it to the **Selected Stereotype** list. Click **OK** to confirm.



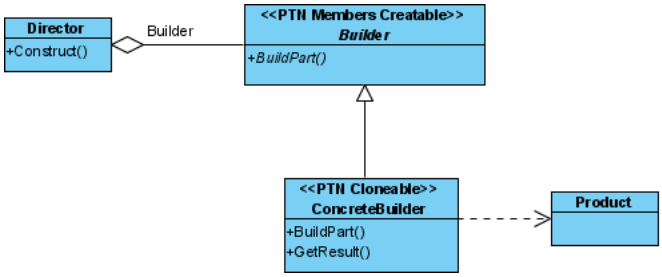
16. In practice, there may be multiple operations in the *Builder* class for creating parts. To represent this, you need to assign a special stereotype to the *Builder* class. Right-click on *Builder* and select **Stereotypes > Stereotypes...** from the popup menu.



17. Assign the stereotype **PTN Members Creatable** to the class. Click **OK** to confirm.

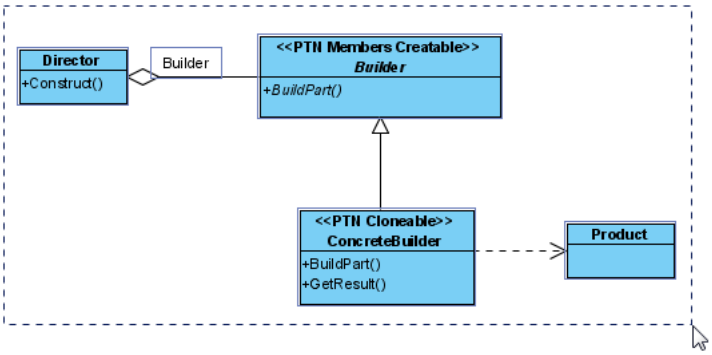


18. For the association end of the association between the *Director* and *Builder* classes, there is a role named *Builder*. Double-click on the *Director* end and enter *Builder* as the name. The diagram should now look like this:

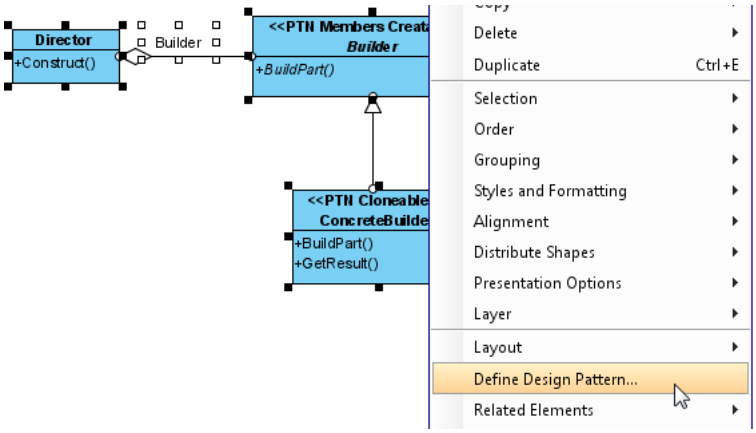


Defining a Pattern

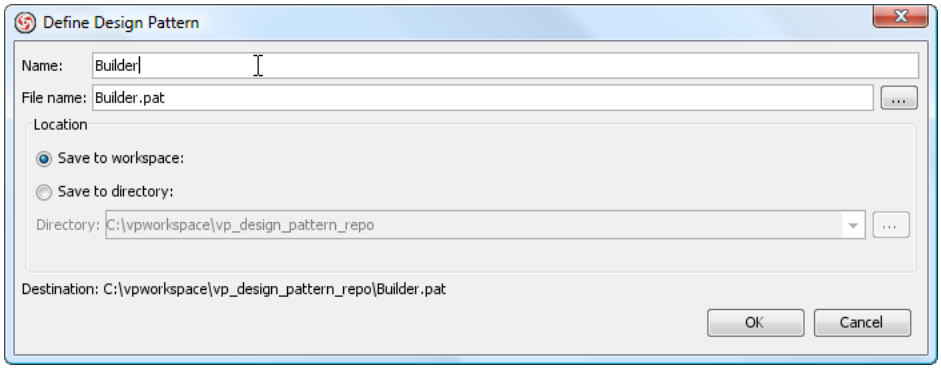
1. Select all classes on the class diagram.



2. Right-click on the selection and select **Define Design Pattern...** from the popup menu.



3. In the **Define Design Pattern** dialog box, specify the pattern name as *Builder*. Keep the file name as is and click **OK** to proceed.

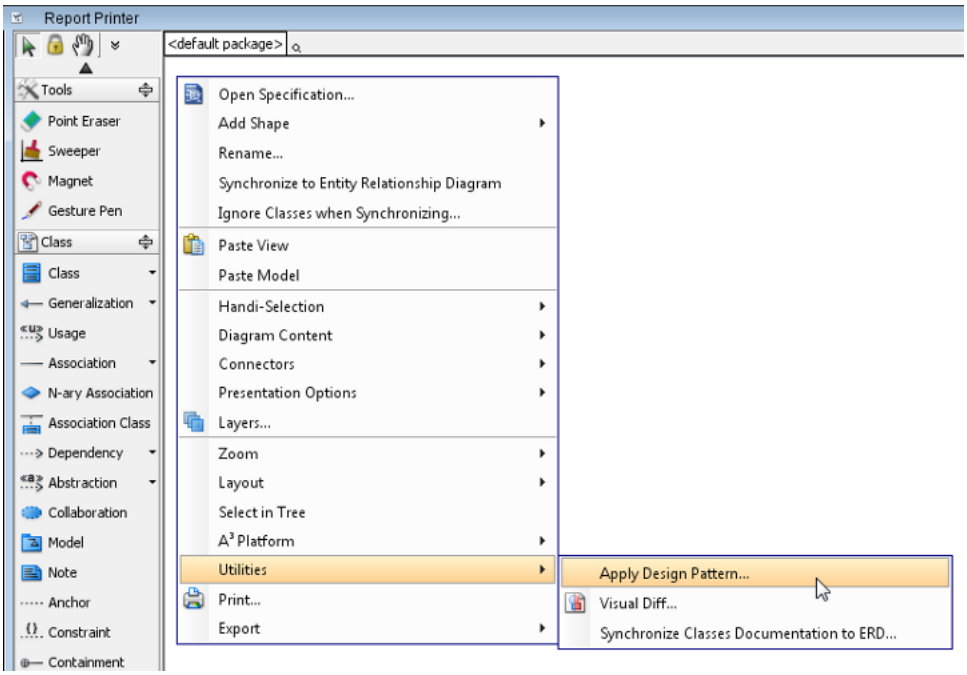


Applying a Design Pattern to a Class Diagram

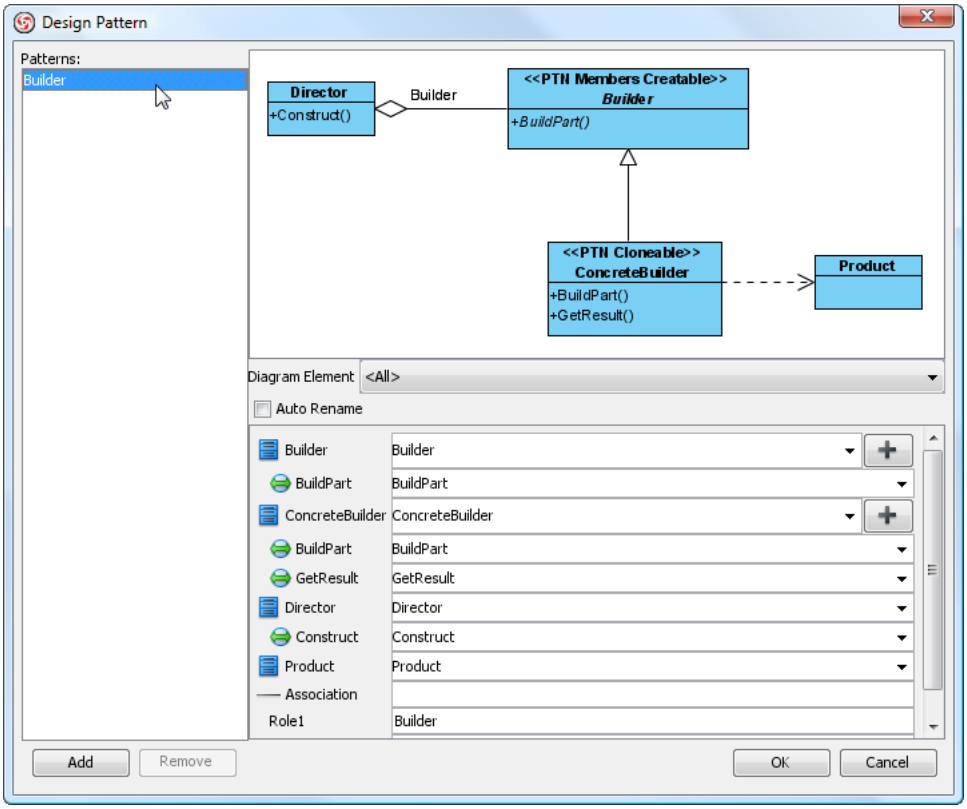
In this section, we will apply the builder pattern to model a report printing system that can print PDF, RTF, and HTML reports one after another, with the same content in each.

1. Create a new project named *Report Printer*.
2. Create a class diagram named *Report Printer*.

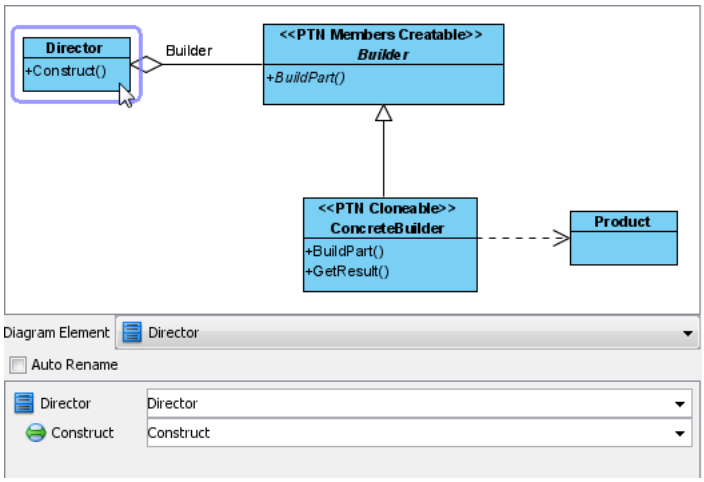
3. Right-click on the class diagram and select **Utilities > Apply Design Pattern...** from the popup menu.



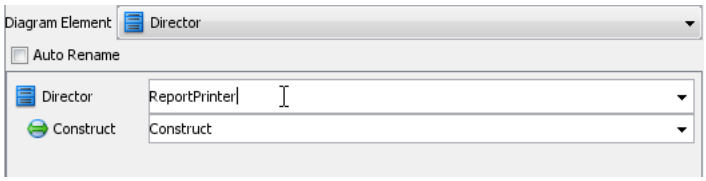
4. In the **Design Pattern** dialog box, select *Builder* from the list of patterns.



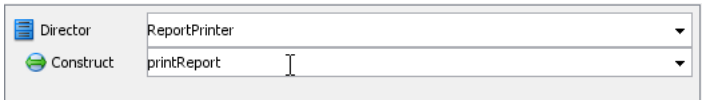
5. Click on *Director* in the preview.



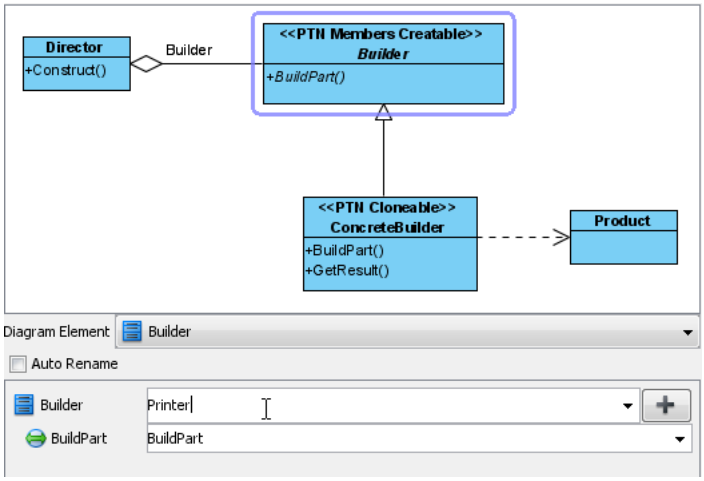
6. Rename *Director* to *ReportPrinter* in the bottom pane.



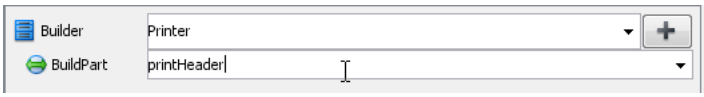
7. Rename the operation *Construct* to *printReport*.



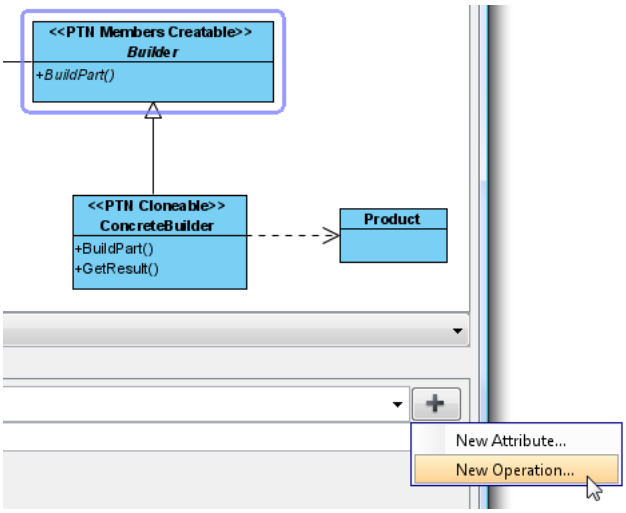
8. Rename the class *Builder* to *Printer*.



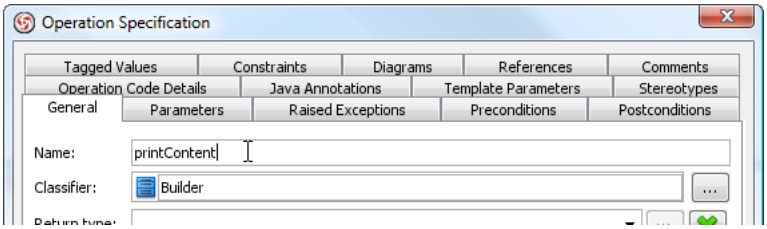
9. Rename the operation *BuildPart* to *PrintHeader*.



10. We need three operations in the *Printer* class for printing the header, content, and footer. Click the + button and select **New Operation...** to add more operations.

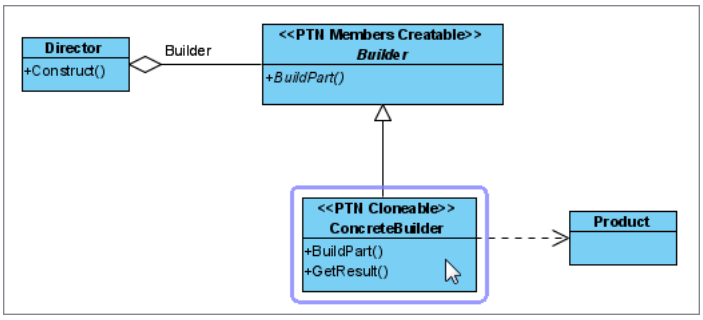


11. In the **Operation Specification**, name the operation "printContent." Check the **Abstract** box at the bottom of the dialog box and click **OK** to confirm.

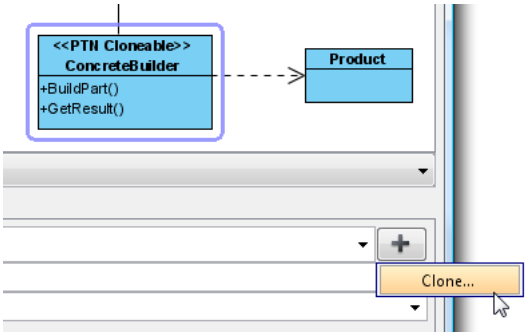


12. Repeat steps 10 and 11 to create the *printFooter* method.

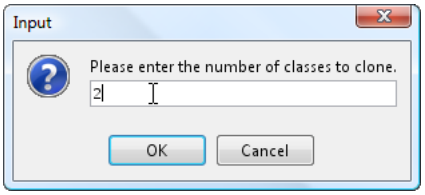
13. Now, let's clone the *ConcreteBuilder* class to create three subclasses as printers for PDF, RTF, and HTML reports. Select *ConcreteBuilder* in the preview pane first.



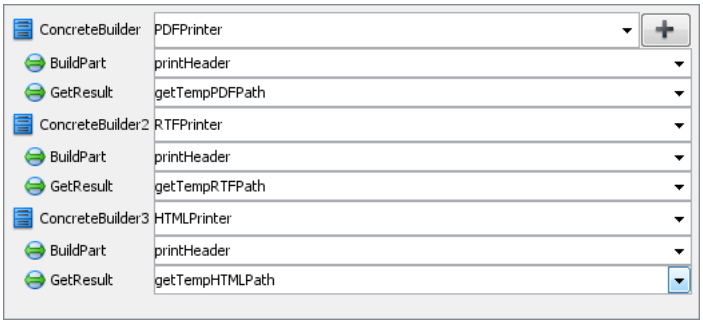
14. Click the + button and select **Clone**.



15. Select *2* as the number of classes to clone.

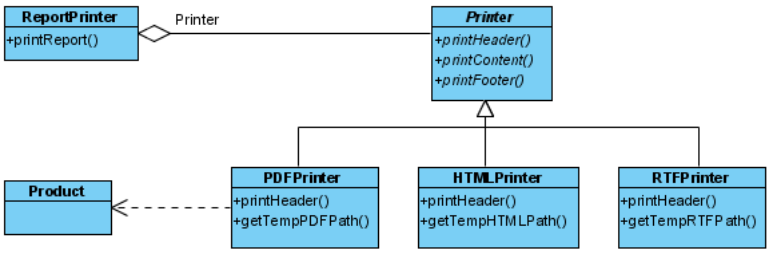


16. Rename the classes and operations as shown below:

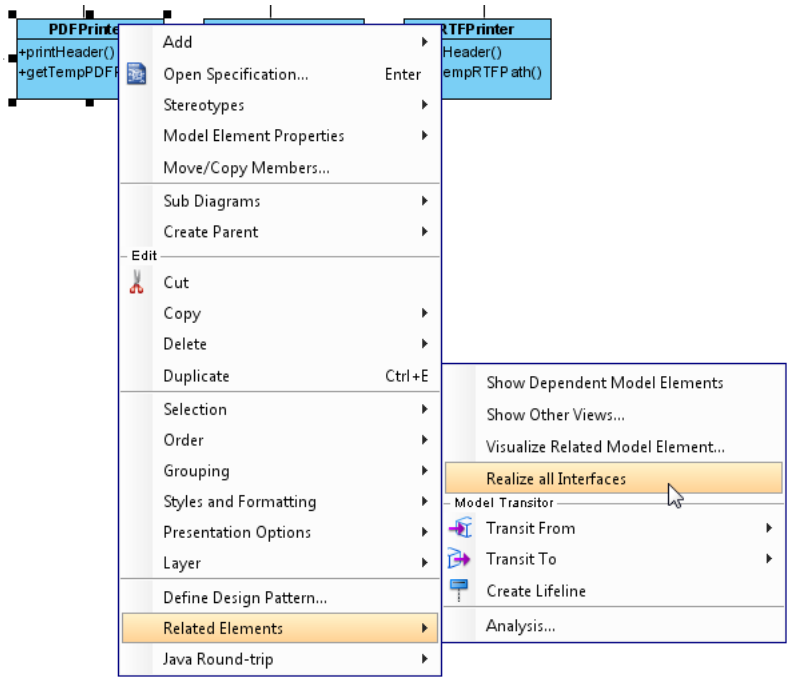


17. Click **OK** to confirm the changes and apply the pattern to the diagram.

18. Tidy up the diagram to make it look like the one below:

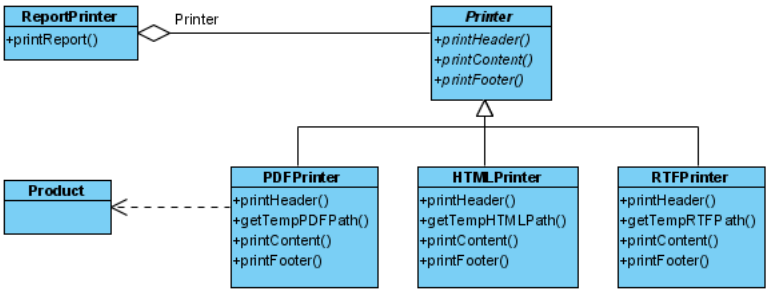


19. Inherit the operations from the *Printer* class to the concrete printer classes. Right-click on the *PDFPrinter* class and select **Related Elements > Realize all Interfaces** from the popup menu.



20. Do the same for the *HTMLPrinter* and *RTFPrinter* classes.

21. Rename the role *Builder* to *Printer*. The result should look like this:



Resources

1. Builder.pat
2. Design Patterns.vpp

Related Links

- Full set of UML tools and UML diagrams

Visual Paradigm home page
(https://www.visual-paradigm.com/)

Visual Paradigm tutorials
(https://www.visual-paradigm.com/tutorials/)