



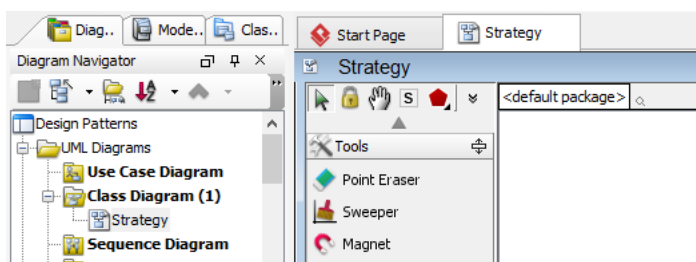
Strategy Pattern Tutorial

Written Date : October 27, 2009

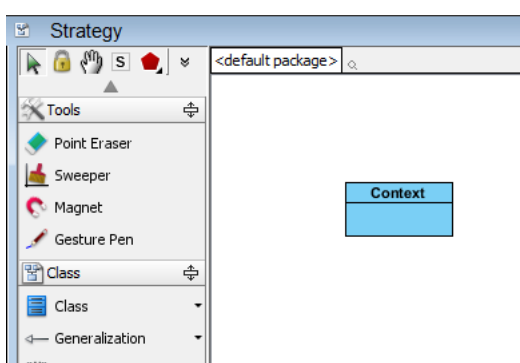
This tutorial is aimed to guide the definition and application of [Gang of Four \(GoF\)](#) strategy [design pattern](#). By reading this tutorial, you will know how to develop a model for the strategy pattern, and how to apply it in practice.

Modeling Design Pattern with Class Diagram

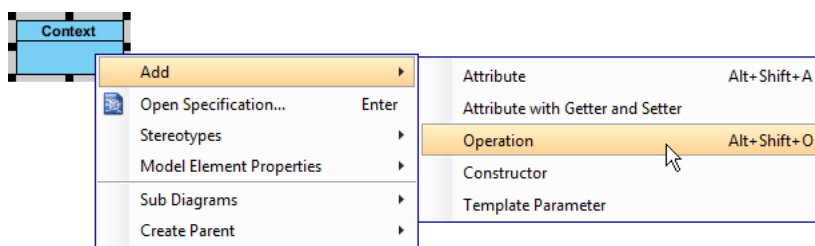
1. Create a new project *Design Patterns*.
2. Create a class diagram *Strategy*.



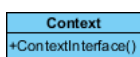
3. Select **Class** from diagram toolbar. Click on the diagram to create a class. Name it as *Context*.



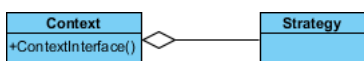
- Right click on the *Context* class, and select **Add > Operation** from the popup menu.



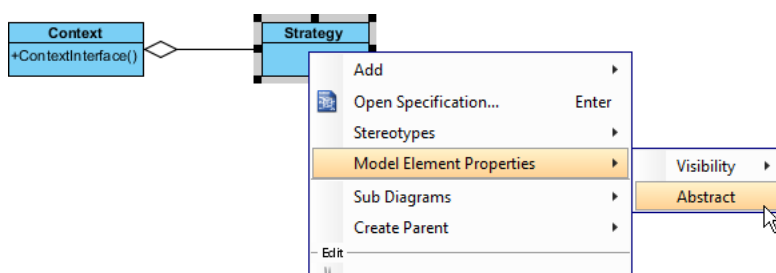
- Name the operation *ContextInterface()*.



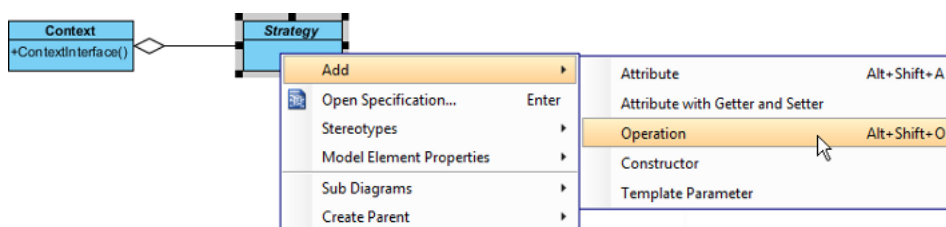
- Move the mouse cursor over the *Context* class, and drag out **Aggregation > Class** to create an associated class *Strategy*.



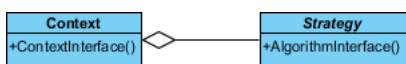
- Right click on *Strategy*, and select **Model Element Properties > Abstract** to set it as abstract.



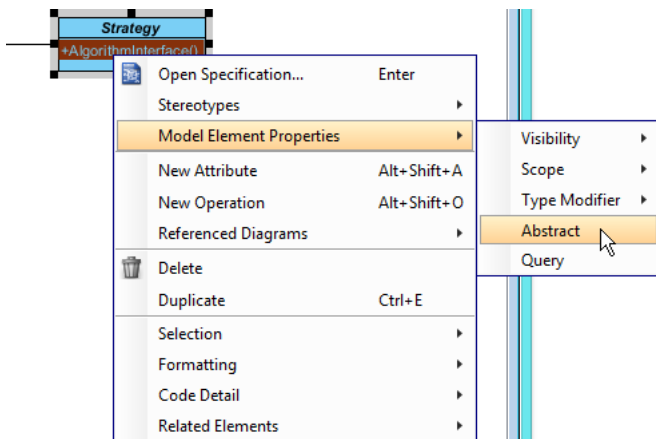
- Right click on the *Strategy* class, and select **Add > Operation** from the popup menu.



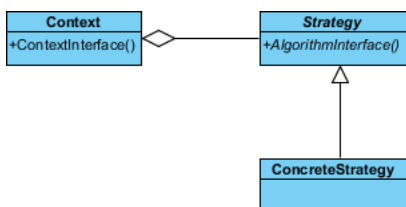
9. Name the operation *AlgorithmInterface()*.



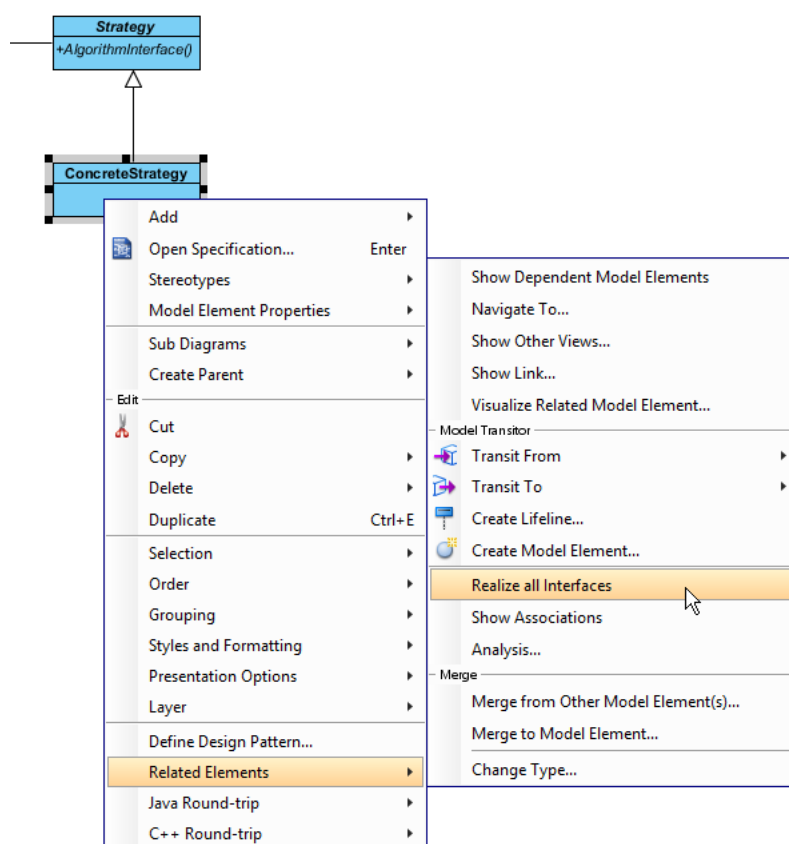
10. Right click on *AlgorithmInterface*, and select **Model Element Properties** > **Abstract** to set it as abstract.



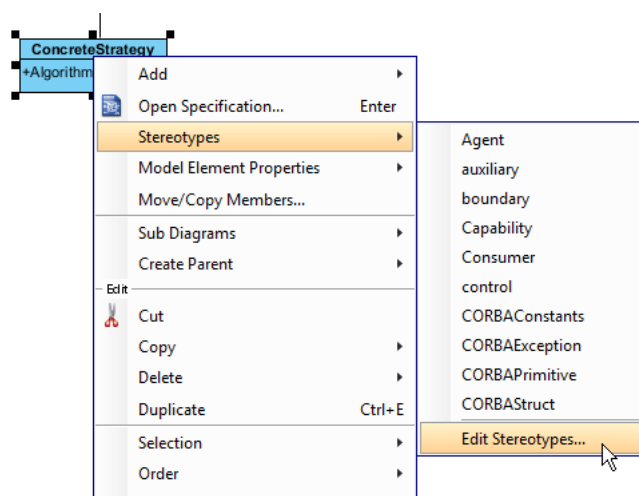
11. Move the mouse cursor over the *Strategy* class, and drag out **Generalization** > **Class** to create subclasses *ConcreteStrategy*.



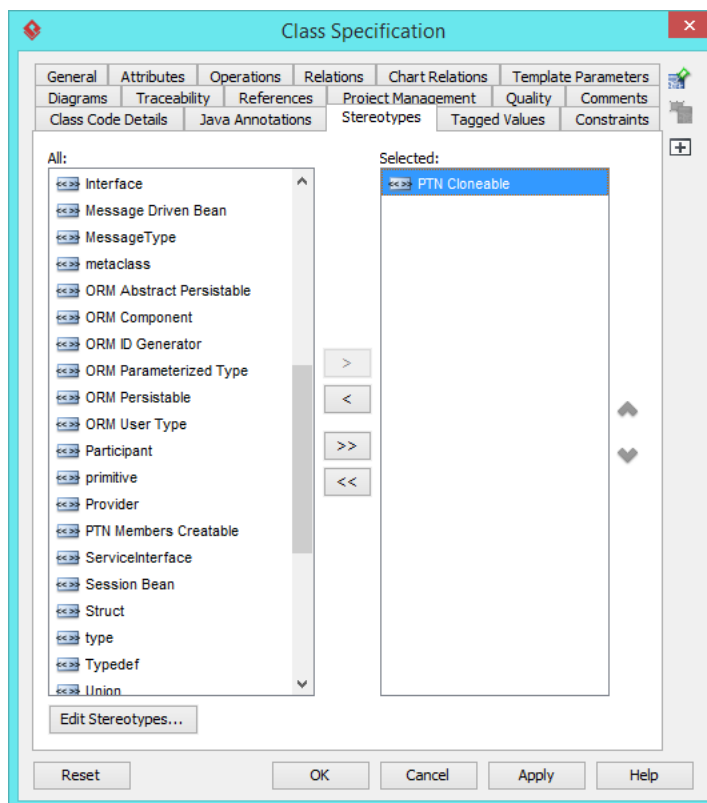
12. We need to make the concrete strategies inherit operations from the strategy class. Right click on *ConcreteStrategy* and select **Related Elements > Realize all Interfaces** from the popup menu.



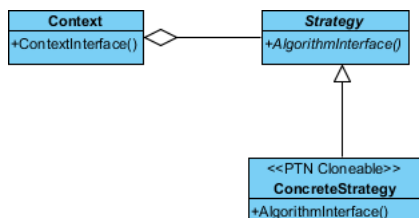
13. In practice, there may be multiple concrete strategies. To represent this, stereotype the class *ConcreteStrategy* as **PTN Cloneable**. Right right on *ConcreteStrategy* and select **Stereotypes > Stereotypes...** from the popup menu.



- In the **Stereotypes** tab of the **Class Specification** dialog box, select **PTN Cloneable** and click **>** to assign it to *ConcreteStrategy* class. Click **OK** to confirm.

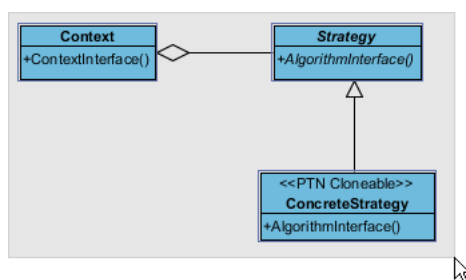


Up to now, the diagram should look like:

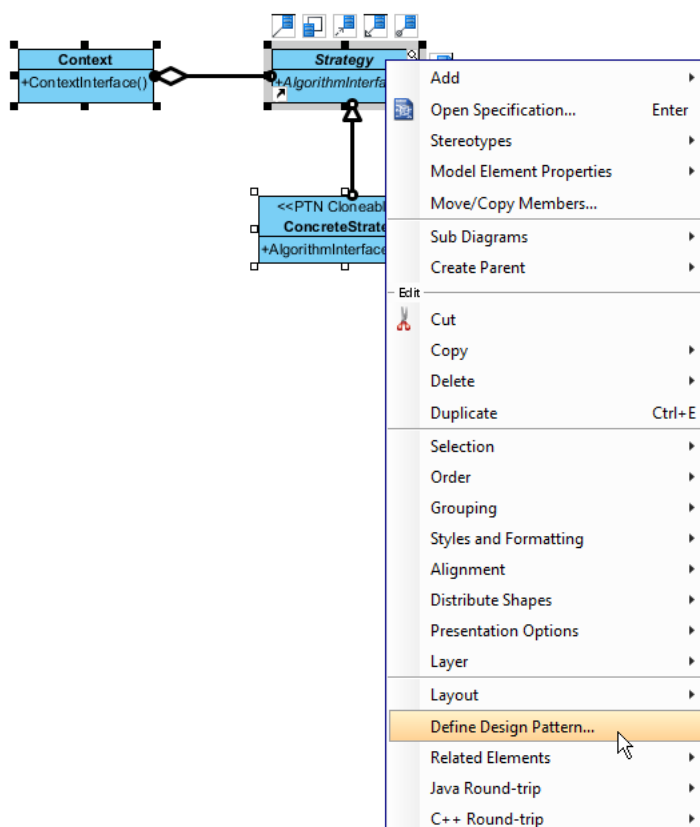


Defining Pattern

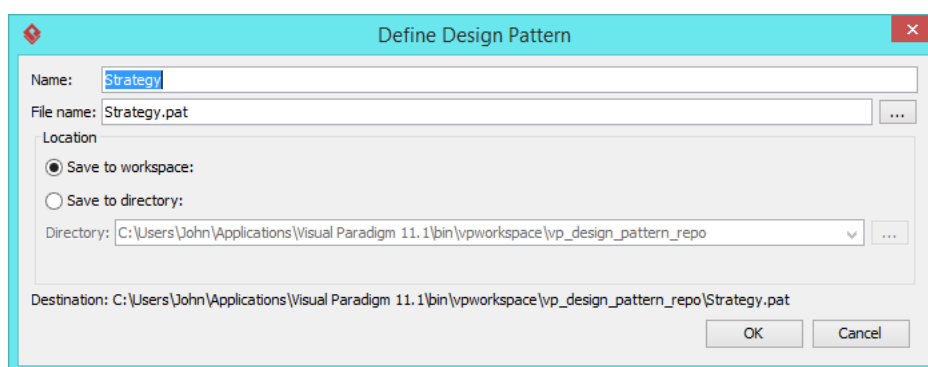
- Select all classes on the class diagram.



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



- In the **Define Design Pattern** dialog box, specify the pattern name *Strategy*. Keep the file name as is. Click **OK** to proceed.

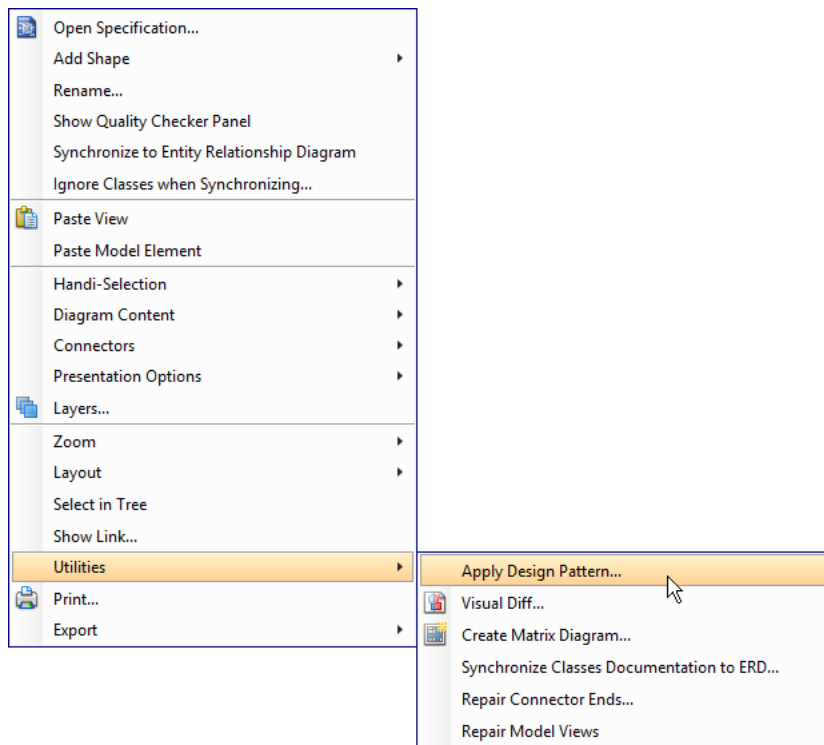


Applying Design Pattern on Class Diagram

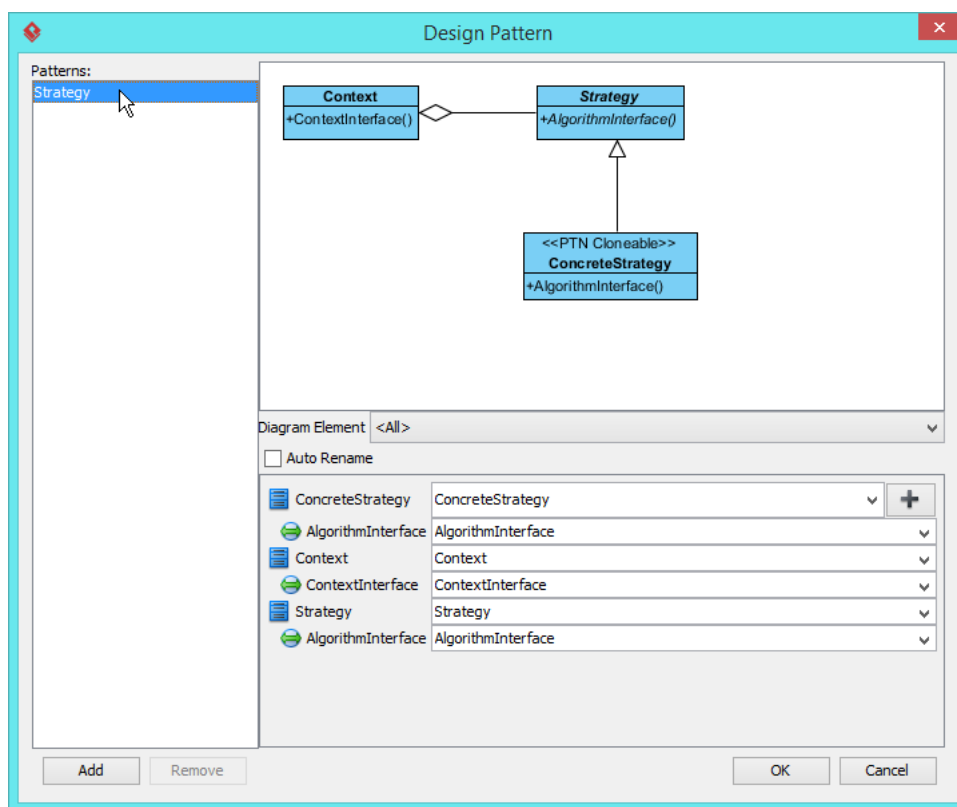
In this section, we are going to apply the strategy pattern in modeling a video game.

- Create a new project *Game*.
- Create a class diagram *Domain Model*.

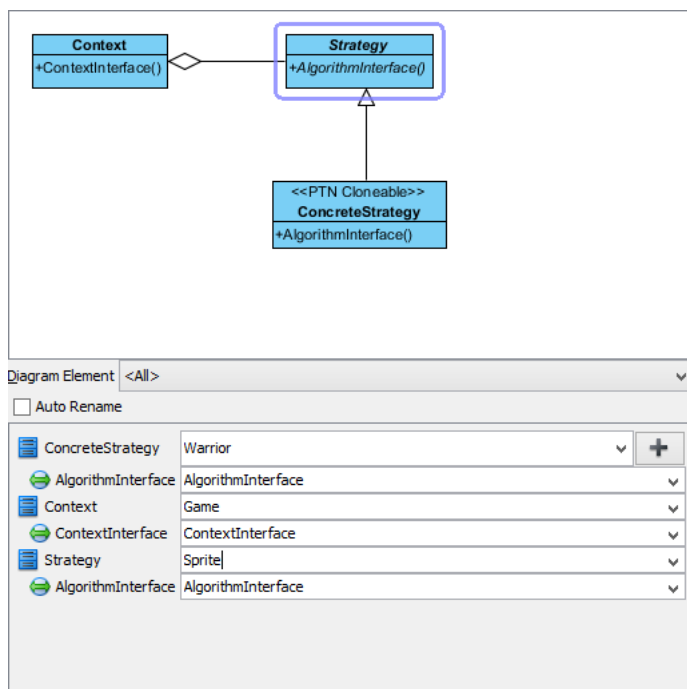
3. Right click on the class diagram and select **Utilities > Apply Design Pattern...** from the popup menu.



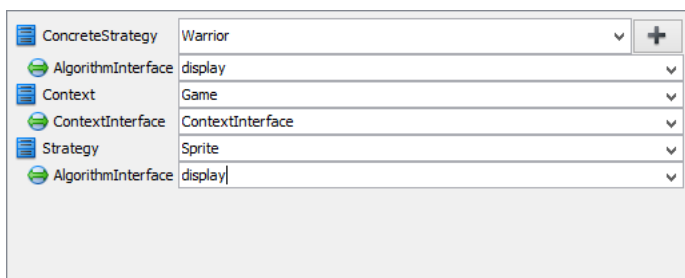
- In the **Design Pattern** dialog box, select *Strategy* from the list of patterns.



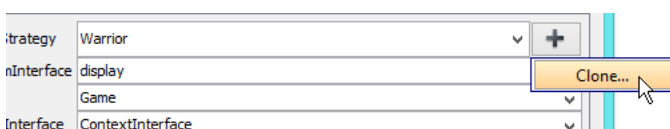
- At the bottom pane, rename *Context*, *Strategy* and *ConcreteStrategy* to *Game*, *Sprite* and *Warrior*.



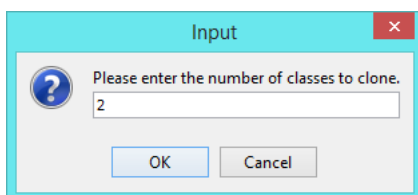
6. Rename operations *AlgorithmInterface* to display.



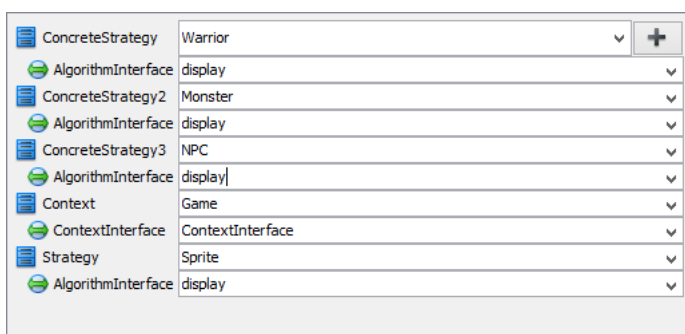
7. We need 2 more concrete strategy for *Monster* and *NPC*. Click on the + button at the *ConcreteStrategy* row and select **Clone...** from the popup menu.



8. Enter 2 to be the number of classes to clone. Click **OK** to confirm.

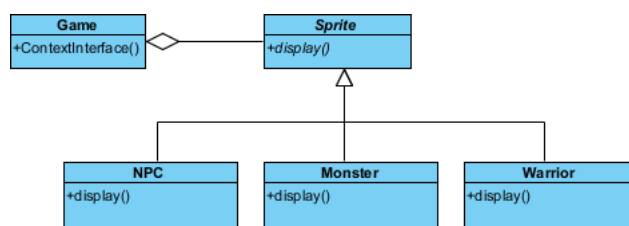


9. Rename *ConcreteStrategy2* and *ConcreteStrategy3* to *Monster* and *NPC*, and operations *AlgorithmInterface* to *display*.



10. Click **OK** to apply the pattern to diagram.

11. Tidy up the diagram. Here is the result:



Resources

1. [Design Patterns.vpp](#)
2. [Strategy.pat](#)

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/>)