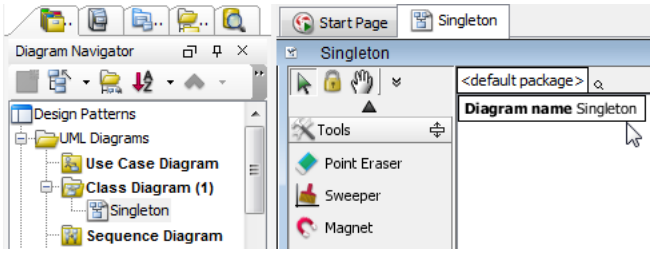
Singleton Pattern Tutorial

Written Date : September 30, 2009

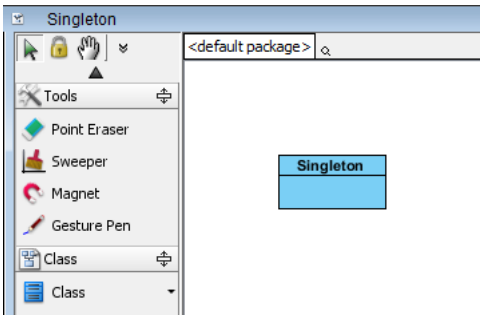
Modeling a Design Pattern with a Class Diagram

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

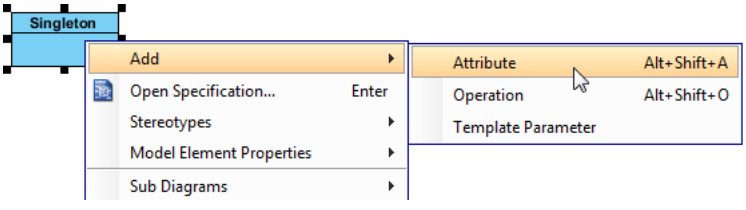
2. Create a class diagram named *Singleton*.



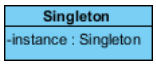
3. Select **Class** from the diagram toolbar. Click on the diagram to create a class and name it *Singleton*.



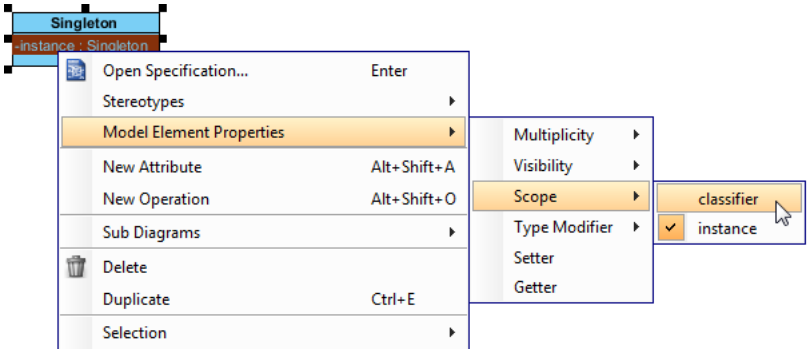
4. Right-click on the *Singleton* class and select **Add** > **Attribute** from the popup menu.



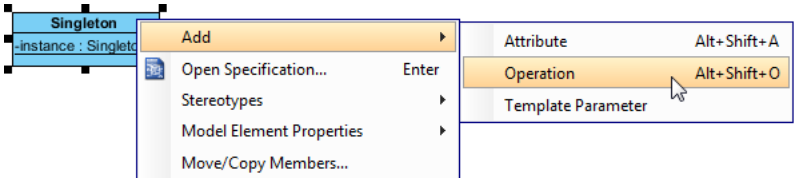
5. Name the attribute `instance` and set its type to `Singleton`.



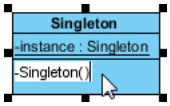
6. The `instance` attribute needs to be static. Right-click on the attribute and select **Model Element Properties** > **Scope** > **Classifier** from the popup menu.



7. Create a constructor for the *Singleton* class. Right-click on *Singleton* and select **Add** > **Operation** from the popup menu.

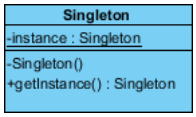


8. Name the operation `Singleton`, which is the same as the class name. Change the visibility from `+` (public) to `-` (private) in front of the operation name to indicate that this is a private constructor.

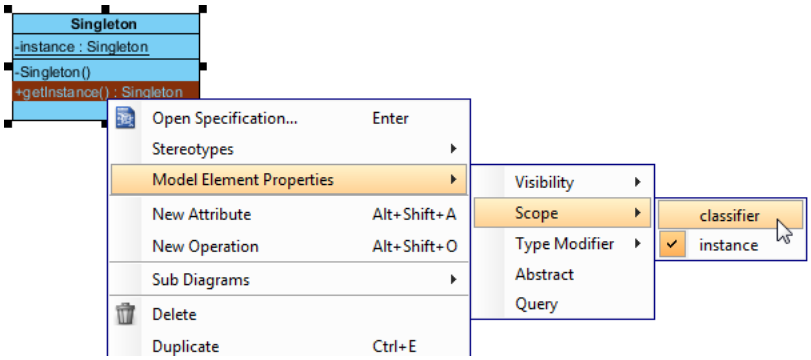


9. Right-click on *Singleton* and select **Add** > **Operation** from the popup menu again.

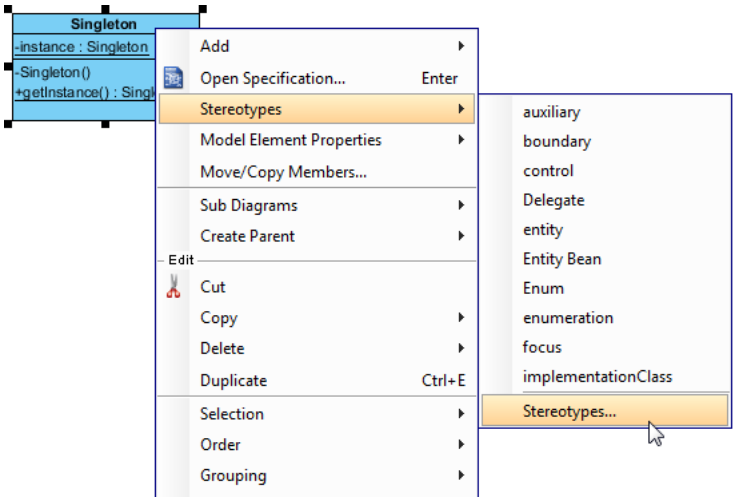
10. Name the operation `getInstance` and set its return type to `Singleton`.



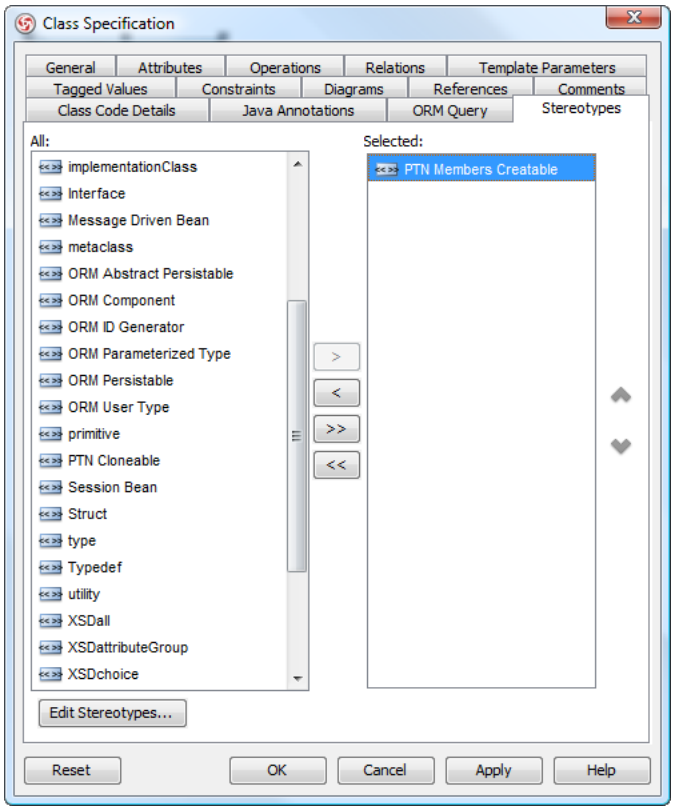
11. The `getInstance` operation needs to be static. Right-click on the operation and select **Model Element Properties** > **Scope** > **Classifier** from the popup menu.



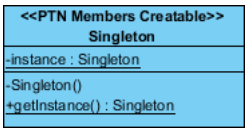
12. In practice, there may be operations for accessing data in the Singleton class. To represent this, stereotype the *Singleton* class as `PTN Members Creatable`. Right-click on the *Singleton* class and select **Stereotypes** > **Stereotypes...** from the popup menu.



13. In the class specification dialog box, select **PTN Members Creatable** and click > to assign it. Click **OK** to confirm.

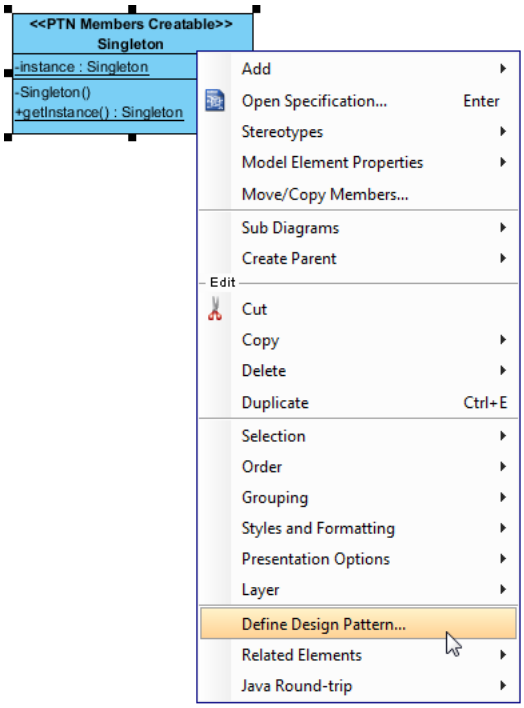


At this point, the diagram should look like this:

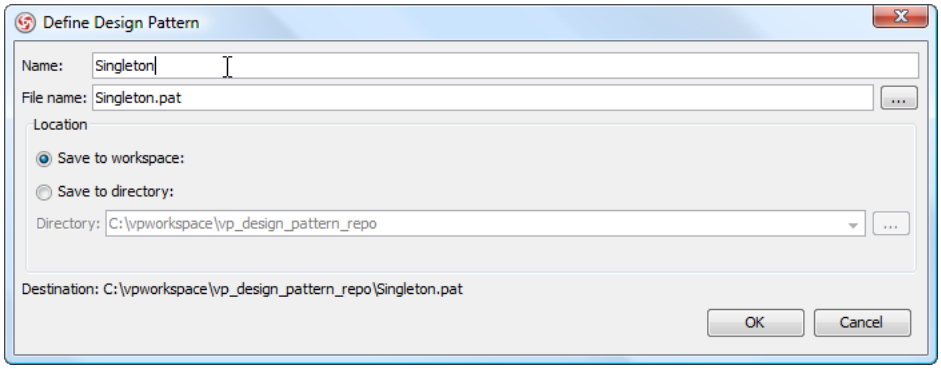


Defining the Pattern

1. Right-click on the *Singleton* class and select **Define Design Pattern...** from the popup menu.



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

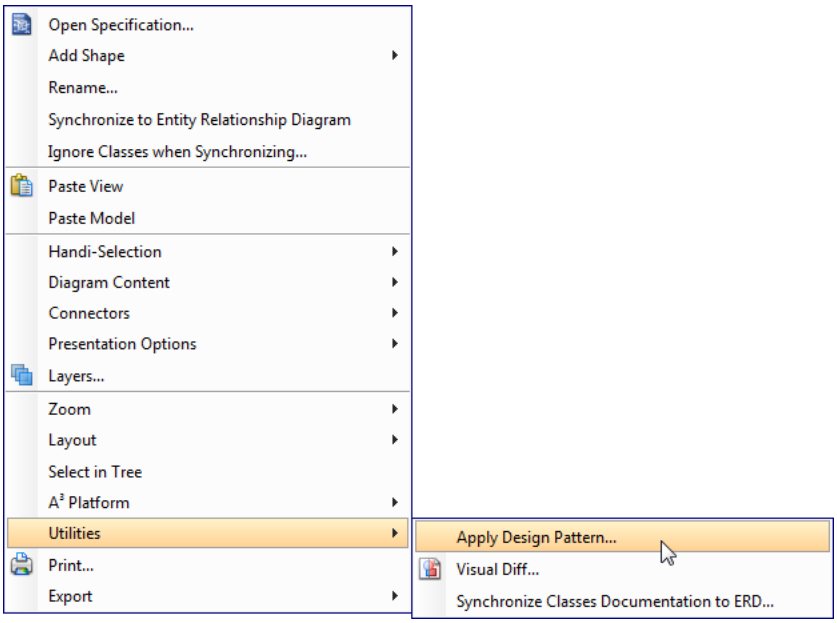


Applying a Design Pattern to a Class Diagram

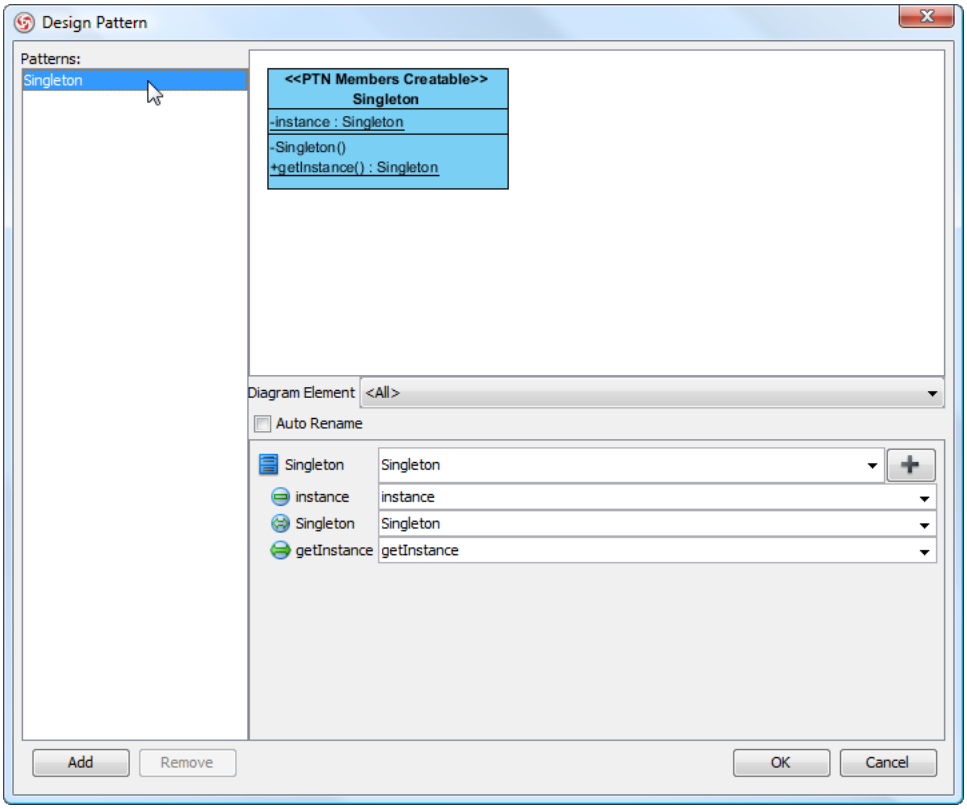
In this section, we are going to apply the singleton pattern in modeling a class registry.

1. Create a new project named *Class Registry*.
2. Create a class diagram named *The Registry*.

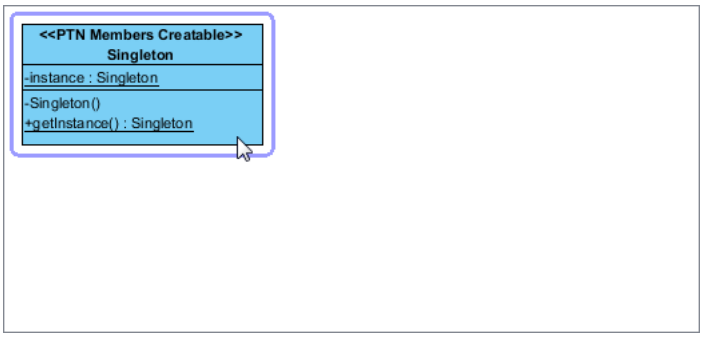
3. Right-click on the class diagram and select **Utilities** > **Apply Design Pattern...** from the popup menu.



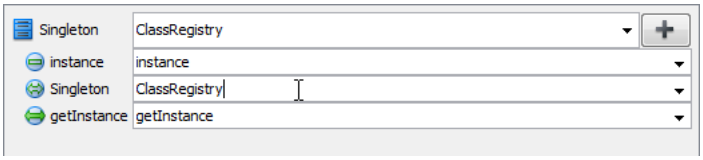
4. In the **Design Pattern** dialog box, select *Singleton* from the list of patterns.



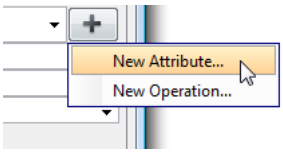
5. Click on *Singleton* in the overview.



6. Rename the *Singleton* class and its constructor to *ClassRegistry* in the bottom pane.

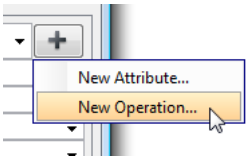


7. We need to add an attribute for holding the classes that users register. Click the **+** button and select **New Attribute...** from the popup menu.



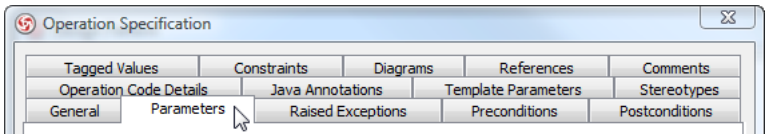
8. In the **Attribute Specification** dialog, enter `classMapping` as the attribute name and `Map` as the type.

9. We need to add operations for registering a class and retrieving a class by type. Click the **+** button and select **New Operation...** from the popup menu.

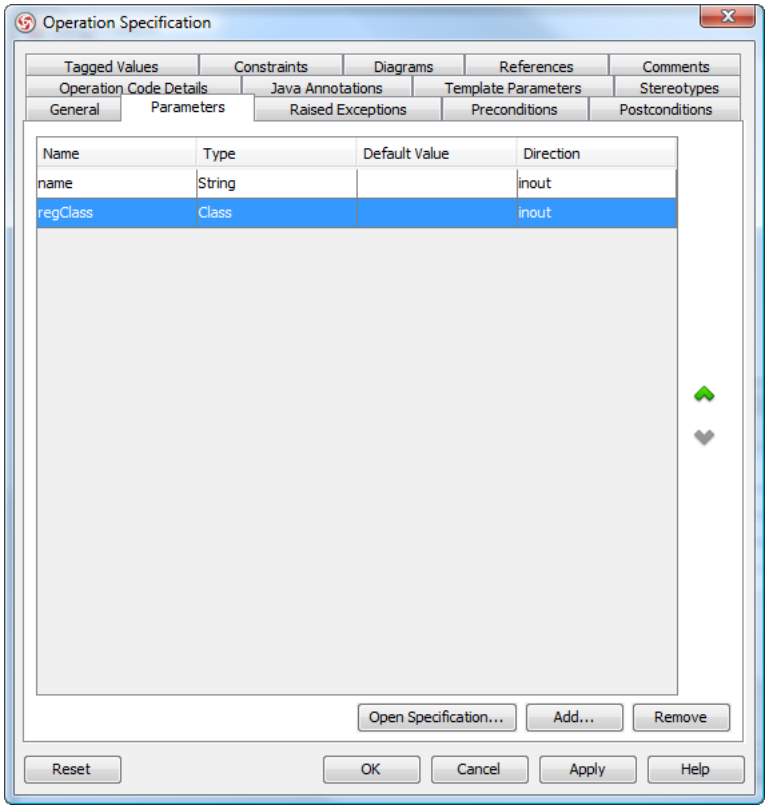


10. In the **Operation Specification** dialog box, enter `registerClass` as the operation name.

11. Open the **Parameters** tab.

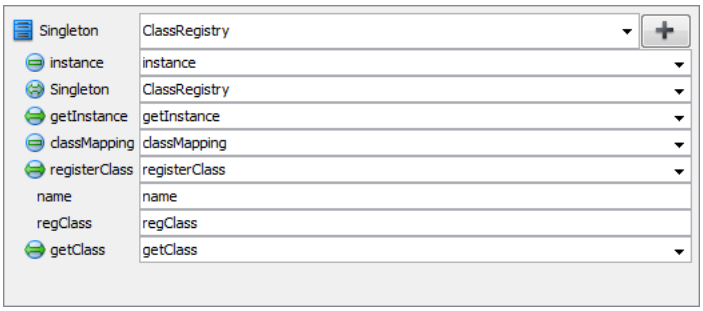


12. Click **Add...** at the bottom of the specification dialog box.
13. In the **Parameter Specification** dialog box, enter `name` as the parameter name and set `String` as the type. Click **OK**.
14. Repeat the previous two steps to add a parameter named `regClass` and set its type to `Class`. Click **OK** to confirm.

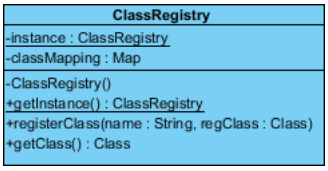


15. Click the **+** button again and select **New Operation...** from the popup menu.

16. In the **Operation Specification** dialog box, enter `getClass` as the name and set `Class` as the return type.



17. Click **OK** to apply the pattern to the diagram. This is the result:



Resources

1. Design Patterns.vpp
2. Singleton.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/)