Visual ◆ Paradigm

# Proxy Pattern Tutorial

Written Date : October 7, 2009

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

Modeling Design Pattern with Class Diagram
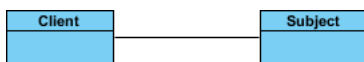
1.   Create a new project *Design Patterns*.
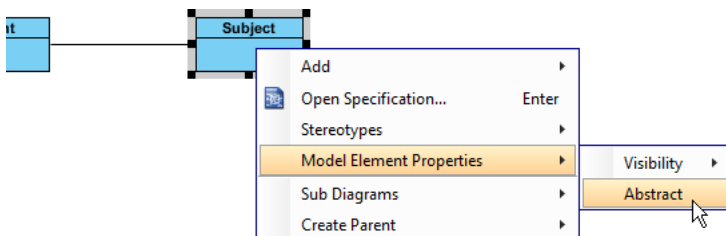
2.   Create a class diagram *Proxy*.
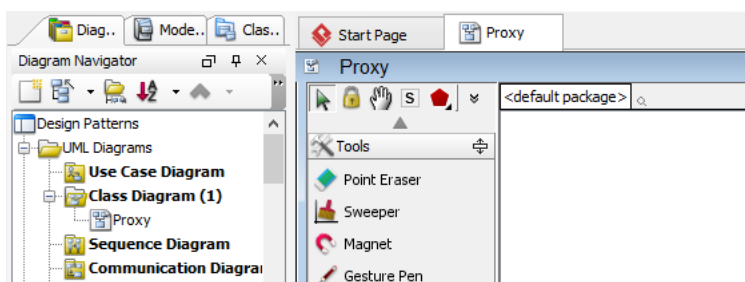
3.   Select **Class** from diagram toolbar. Click on the diagram to create a class. Name it as *Client*.
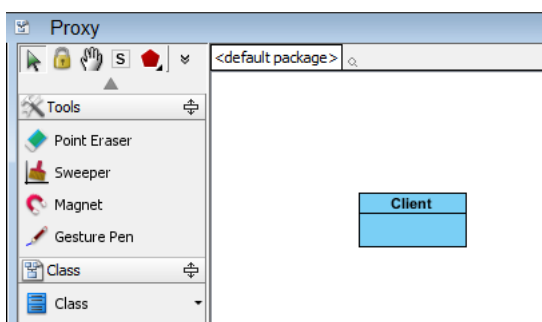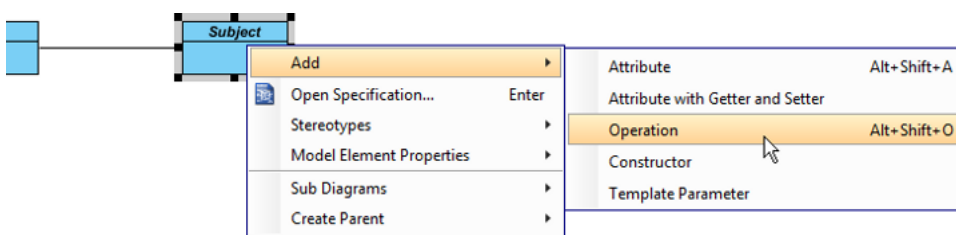
4. Move the mouse cursor over the *Client* class, and drag out **Association** > **Class** to create an associated class *Subject*.
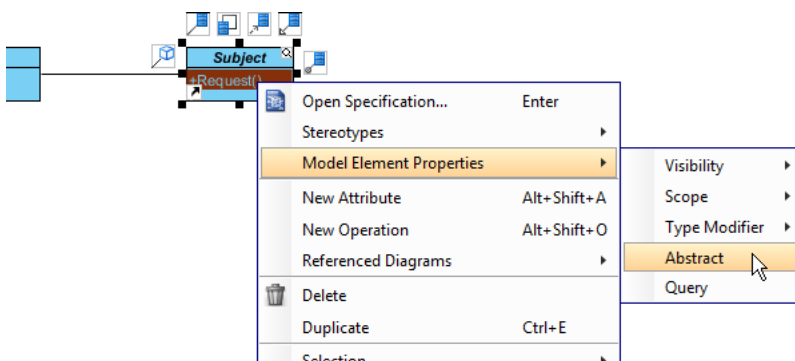

5. Right-click on *Subject*, and select **Model Element Properties** > **Abstract** to set it as abstract.
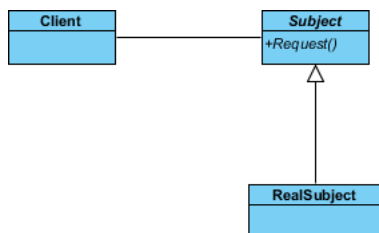

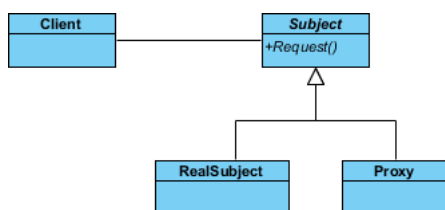6. Right-click on the *Subject* class, and select **Add** > **Operation** from the popup menu.

7. Name the operation *Request()*.

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

9. Move the mouse cursor over the *Subject* class, and drag out **Generalization** > **Class** to create a subclass *RealSubject*.


10. Repeat the previous step to create another subclass from *Subject*, namely *Proxy*.

11. In practice, there may be multiple requests. To represent this, stereotype the class *Subject* as **PTN Members Creatable**. Right-click on *Subject* and select **Stereotypes** > **Stereotypes...** from the popup menu.
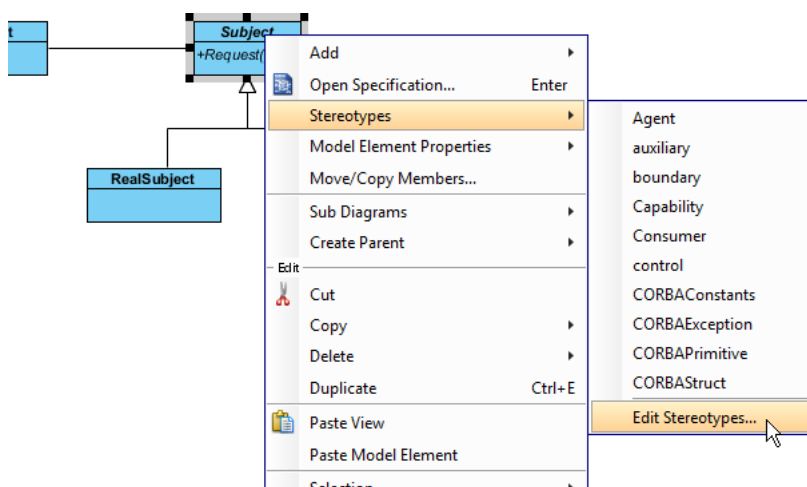
12. In the **Stereotypes** tab of the **Class Specification** dialog box, select **PTN Members Creatable** and click > to assign it to *Subject* class. Click **OK** to confirm.
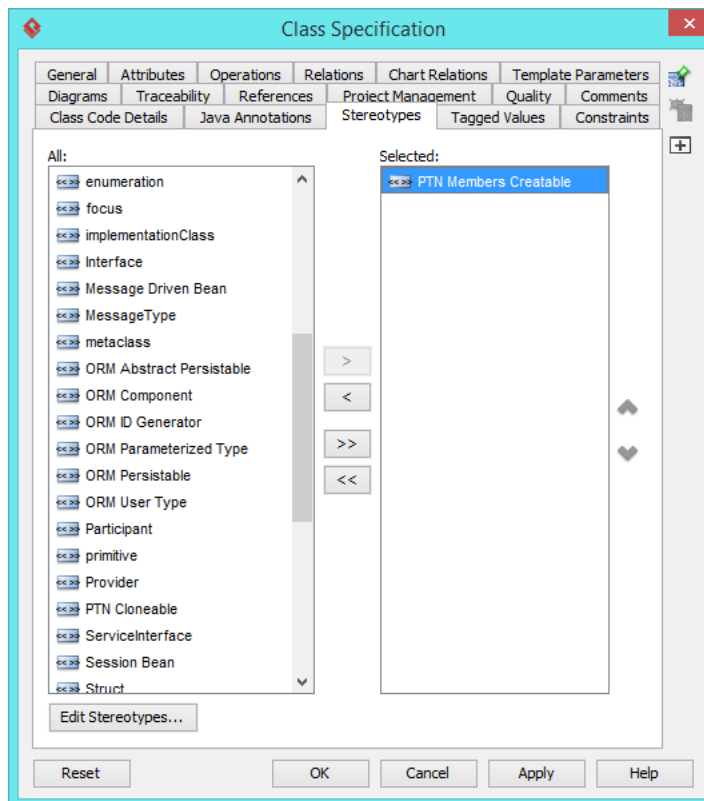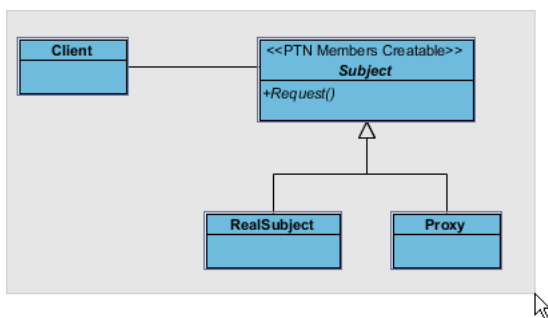

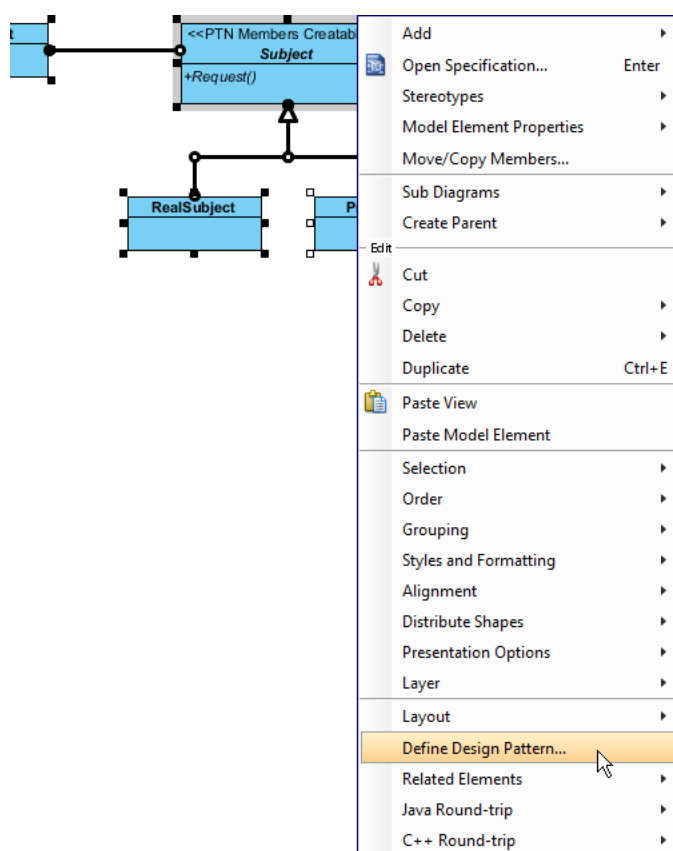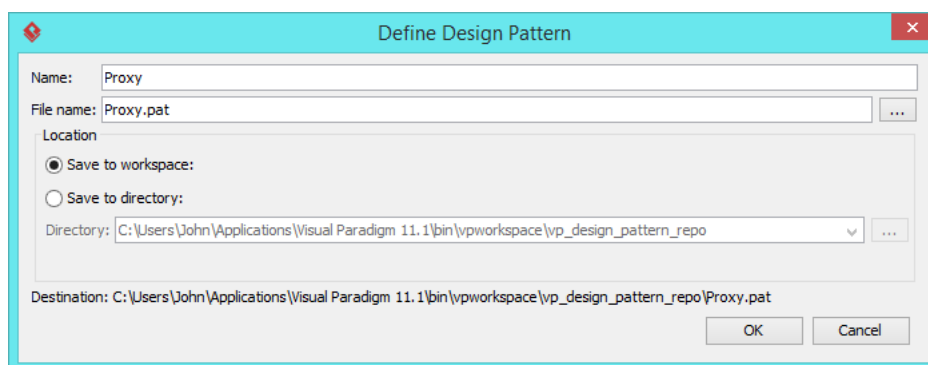Up to now, the diagram should look like this:
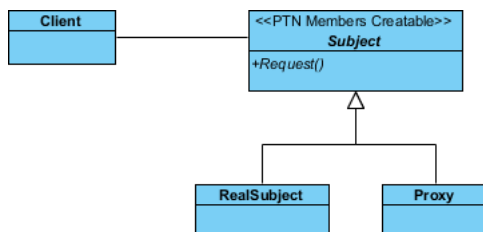
Defining Pattern

1. Select all classes on the class diagram.

2. Right-click on the selection and select **Define Design Pattern...** from the popup menu.
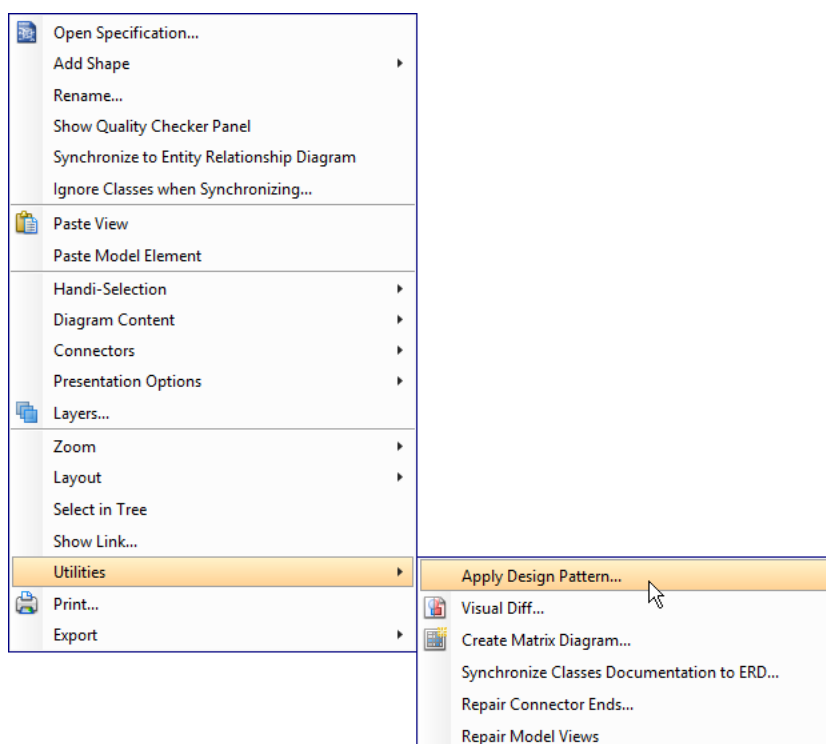
3. In the **Define Design Pattern** dialog box, specify the pattern name *Proxy*. Keep the file name as it. Click **OK** to proceed.


Applying Design Pattern on Class Diagram

In this section, we are going to apply the proxy pattern in modeling a client class that talks to a proxy account class.
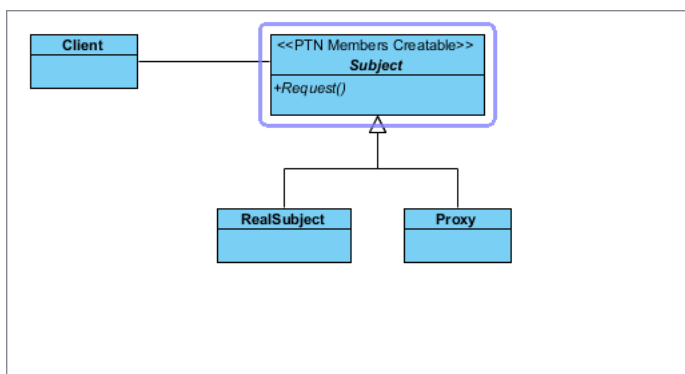
1. Create a new project *Account Management*.

2. Create a class diagram *Domain Model*.

3. Right-click on the class diagram and select **Utilities** > **Apply Design Pattern...** from the popup menu.
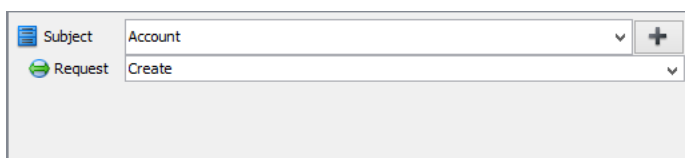
4. In the **Design Pattern** dialog box, select *Proxy* from the list of patterns.
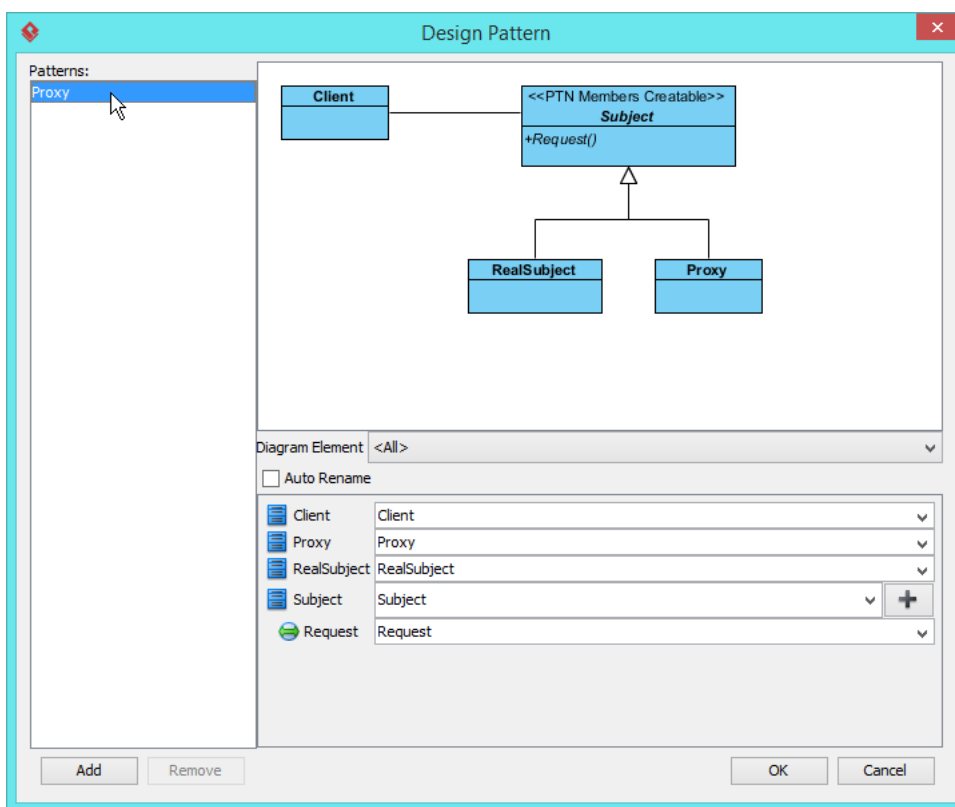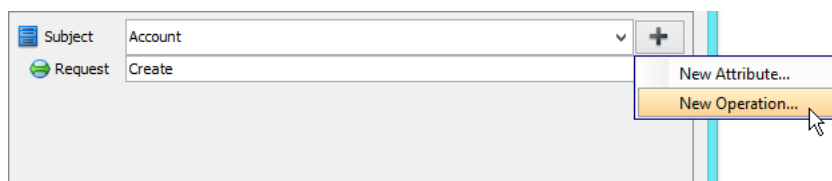

5. Click on *Subject* in the overview.

6. Rename *Subject* to *Account*, and operation *Request* to *Create* at the bottom pane.

7. Besides the operation *Create*, we also need two more operations for *Suspend* and *Delete*. Keep *Subject* selected, click on the + button at the bottom pane, and select **New Operation...** from the popup menu.
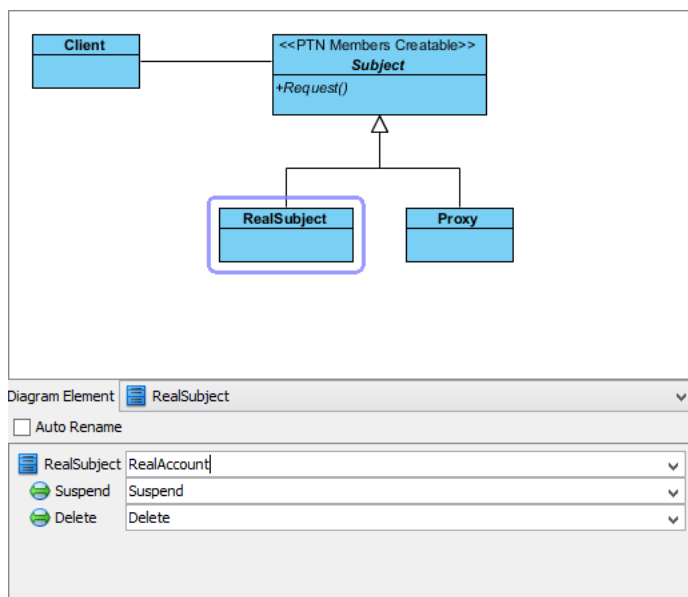

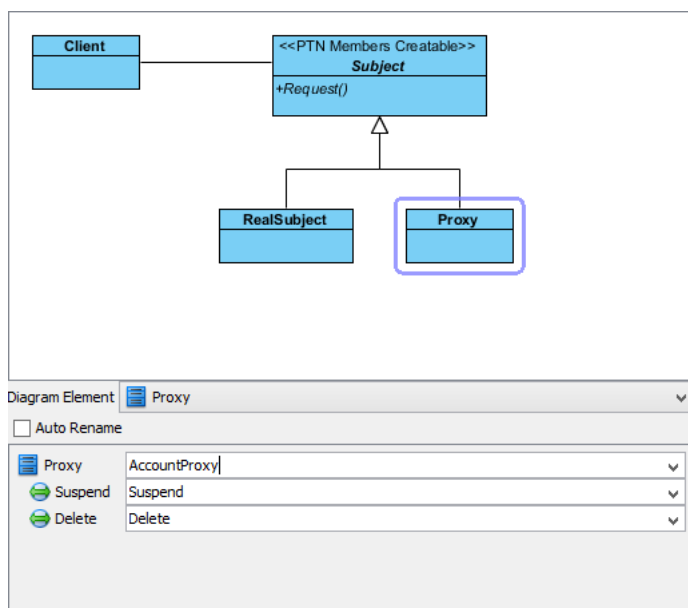8. In the **Operation Specification** dialog box, name the operation *Suspend*.
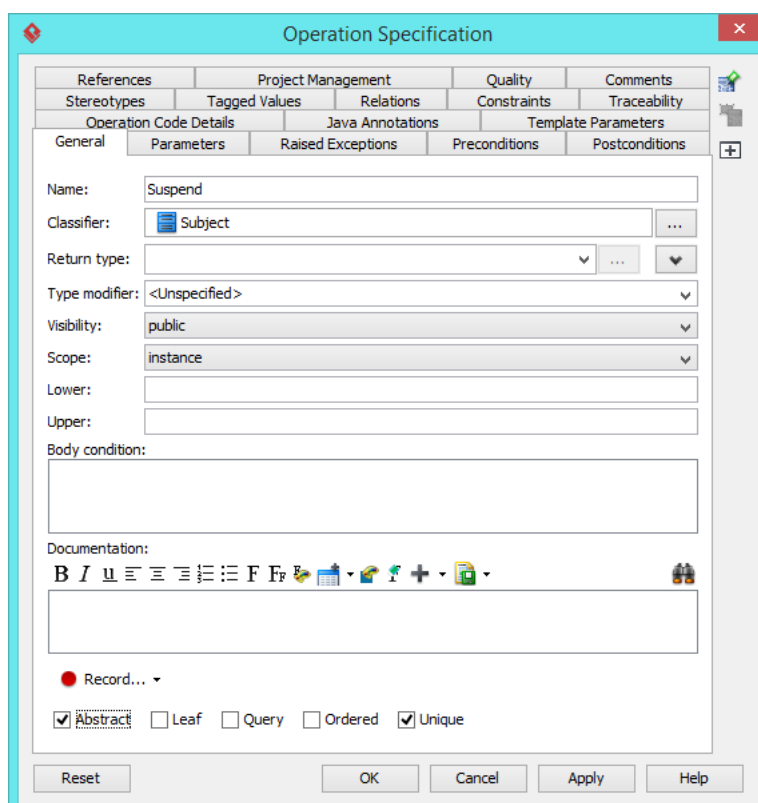
9. Repeat steps 7 and 8 to create operation *Delete*.

10. Select *RealSubject* in overview, and rename it as *RealAccount* at the bottom pane.

11. Select *Proxy* in overview, and rename it as *AccountProxy* at the bottom pane. Click **OK** to apply the pattern to diagram.

12. We need to make the real object and the proxy class inherit operations from the subject class. Right-click on *RealAccount* and *AccountProxy*, and select **Related Elements** > **Realize all Interfaces** from the popup menu.
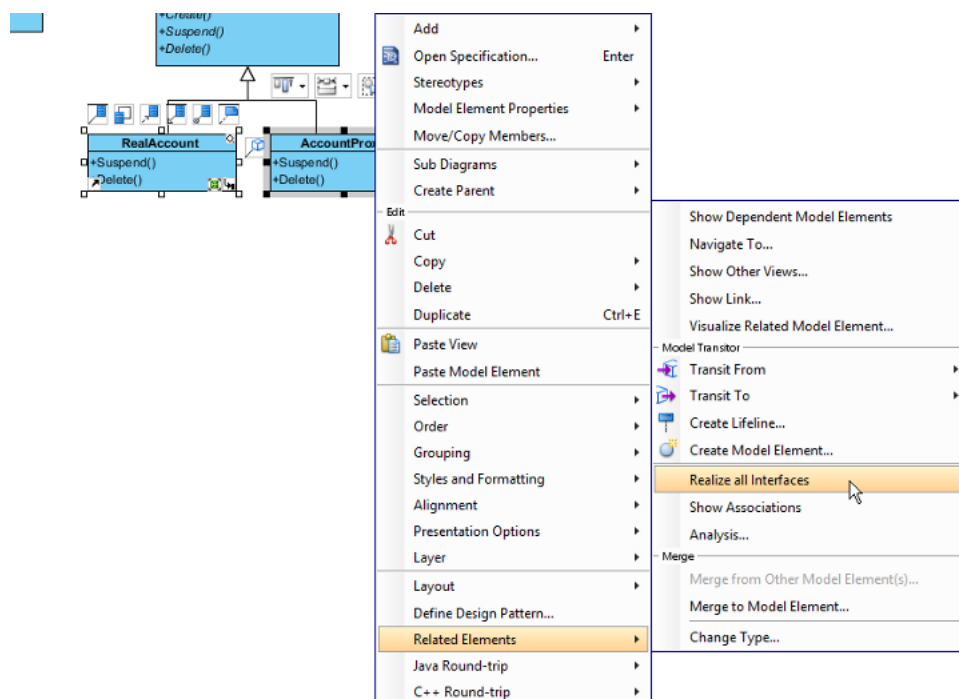
The diagram should look like:
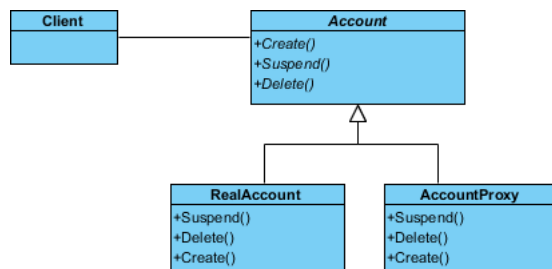
Resources
1.   Proxy.pat

2.   Design Patterns.vpp

Related Links
•   Full set of UML tools and UML diagrams

Visual Paradigm

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

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