

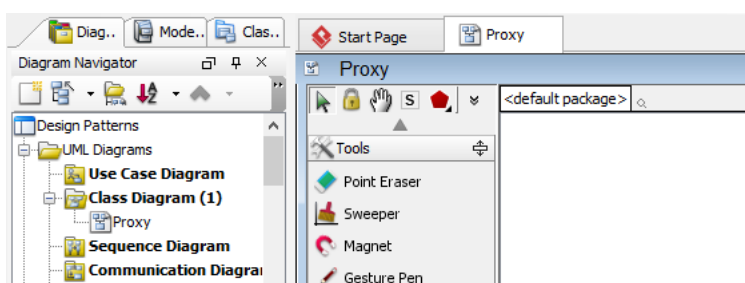


## Proxy Pattern Tutorial

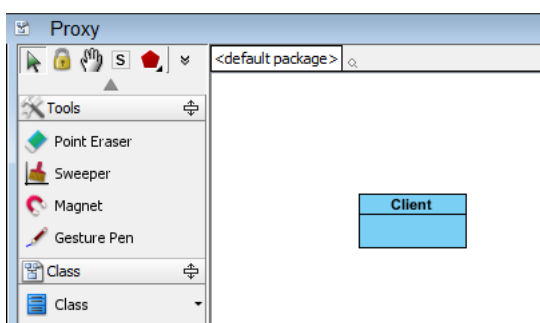
Written Date : October 7, 2009

### Modeling a Design Pattern with a Class Diagram

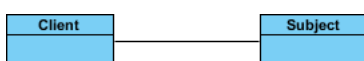
1. Create a new project named *Design Patterns*.
2. Create a class diagram named *Proxy*.



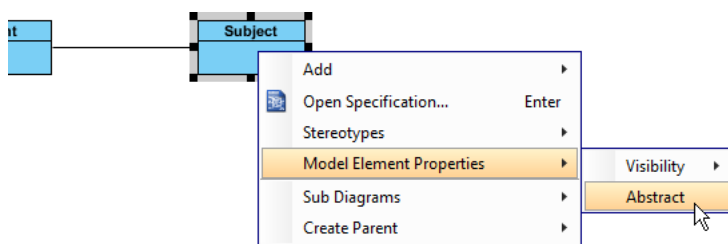
3. Select **Class** from the diagram toolbar. Click on the diagram to create a class. Name it *Client*.



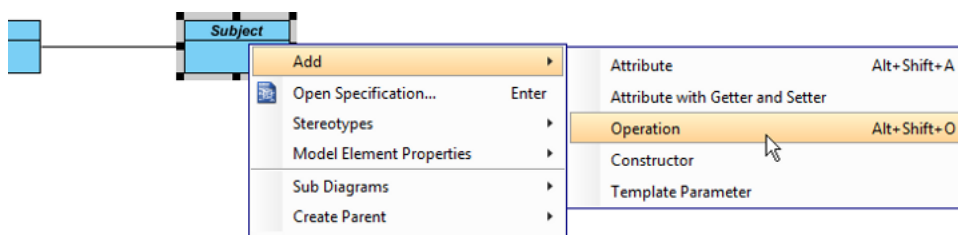
4. Move the mouse cursor over the *Client* class and drag out **Association > Class** to create an associated class *Subject*.



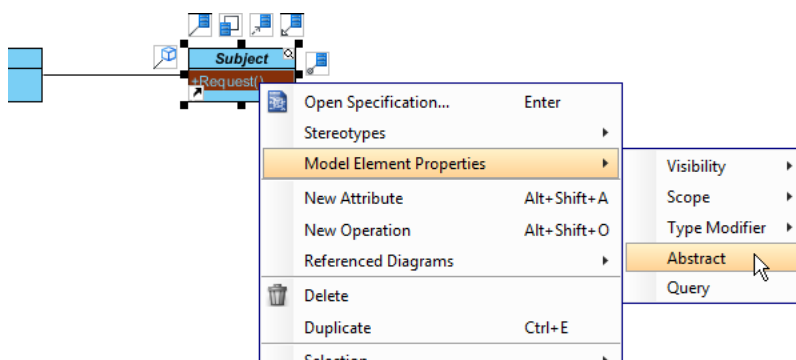
- Right-click on *Subject* and select **Model Element Properties** > **Abstract** to set it as abstract.



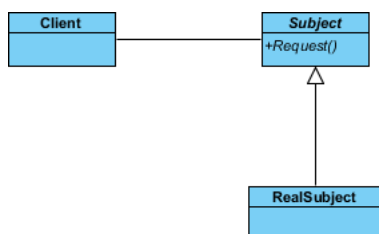
- Right-click on the *Subject* class and select **Add** > **Operation** from the popup menu.



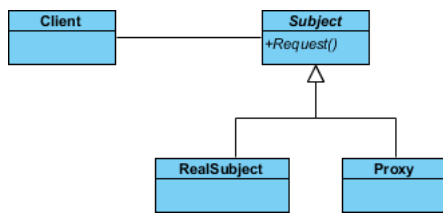
- Name the operation *Request()*.
- Right-click on *Request* and select **Model Element Properties** > **Abstract** to set it as abstract.



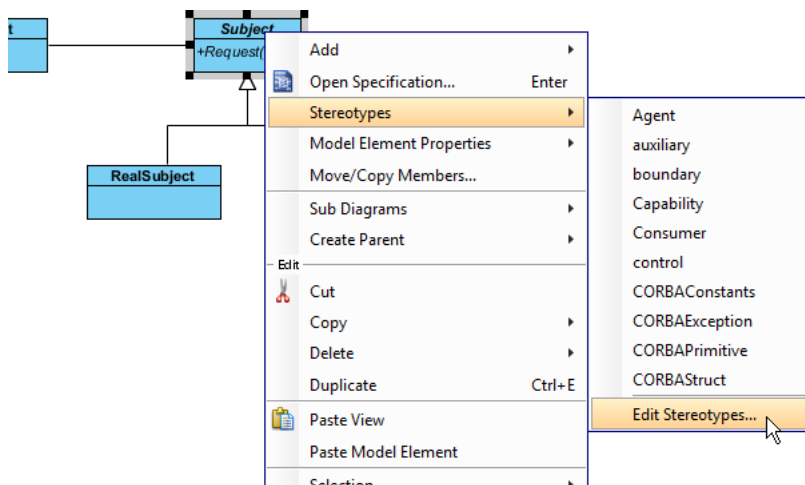
- Move the mouse cursor over the *Subject* class and drag out **Generalization** > **Class** to create a subclass *RealSubject*.



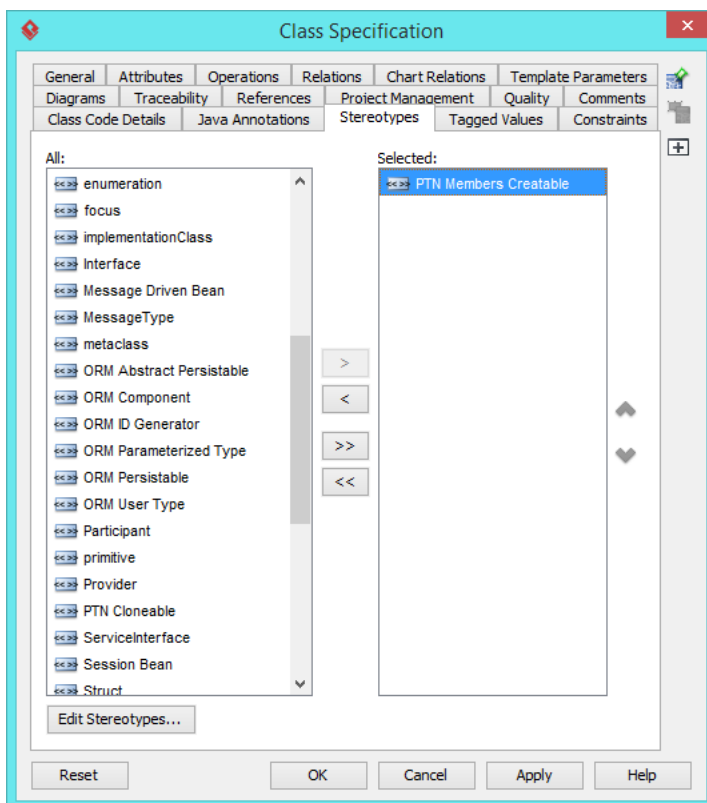
10. Repeat the previous step to create another subclass from *Subject*, namely *Proxy*.



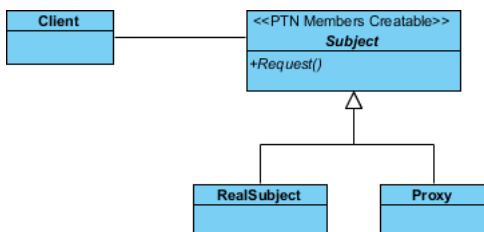
11. In practice, there may be multiple requests. To represent this, stereotype the class *Subject* as **PTN Members Creatable**. Right-click on *Subject* and select **Stereotypes > Stereotypes...** from the popup menu.



12. In the **Stereotypes** tab of the **Class Specification** dialog box, select **PTN Members Creatable** and click > to assign it to the *Subject* class. Click **OK** to confirm.

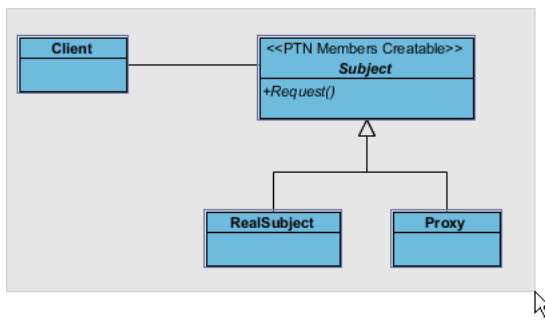


Up to now, the diagram should look like this:

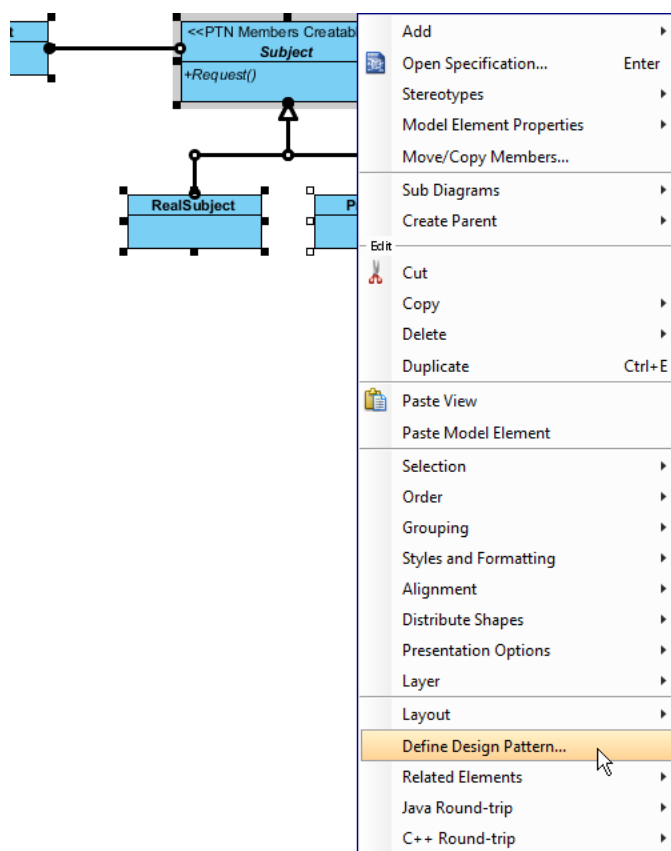


## Defining a Pattern

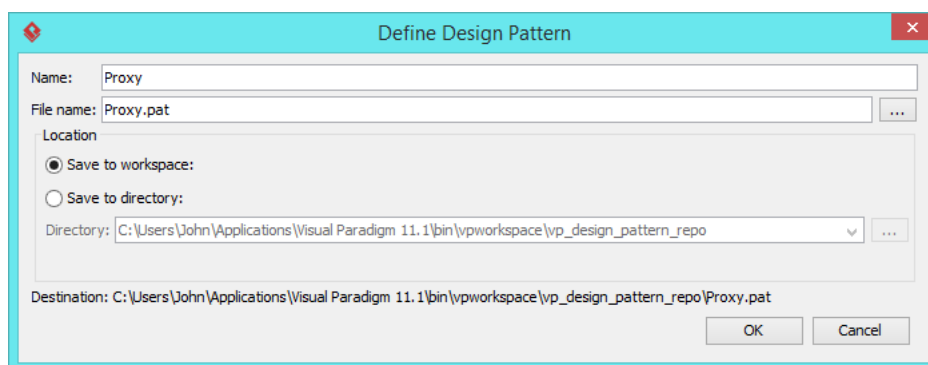
1. Select all classes on the class diagram.



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



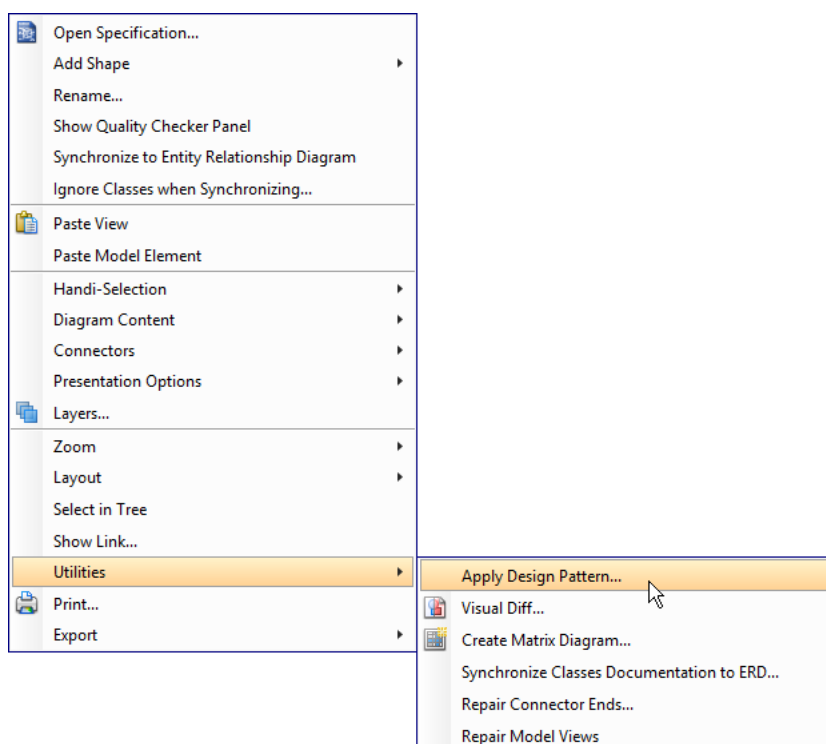
3. In the **Define Design Pattern** dialog box, specify the pattern name *Proxy*. Keep the file name as is. Click **OK** to proceed.



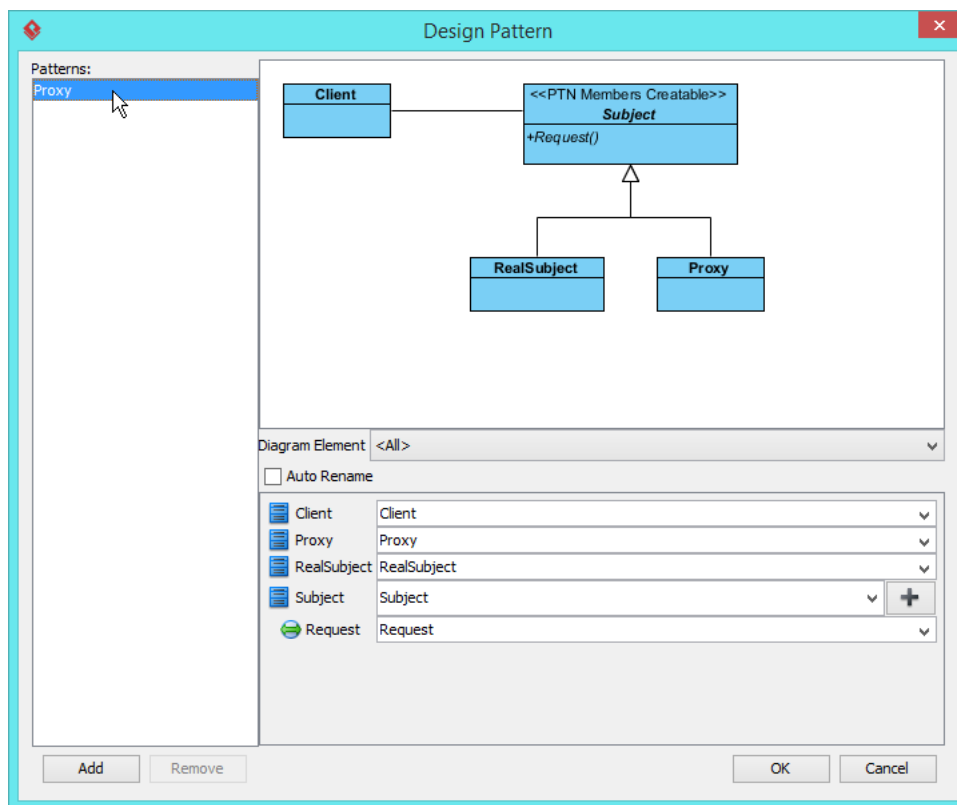
### Applying a Design Pattern on a Class Diagram

In this section, we are going to apply the proxy pattern in modeling a client class that talks to a proxy account class.

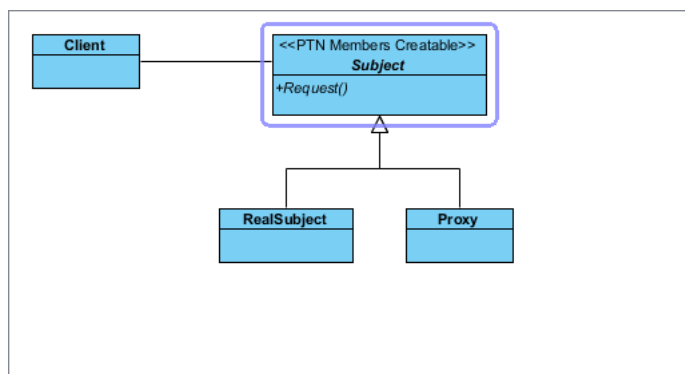
1. Create a new project *Account Management*.
2. Create a class diagram *Domain Model*.
3. Right-click on the class diagram and select **Utilities > Apply Design Pattern...** from the popup menu.



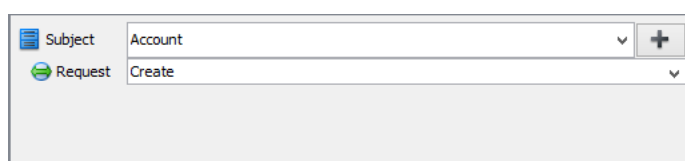
- In the **Design Pattern** dialog box, select *Proxy* from the list of patterns.



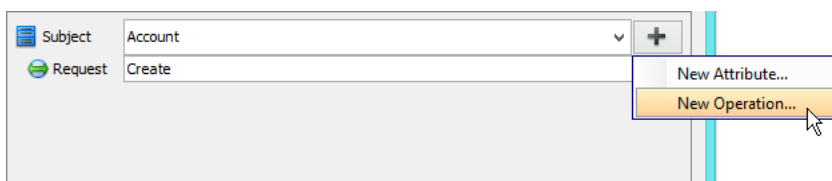
- Click on *Subject* in the overview.



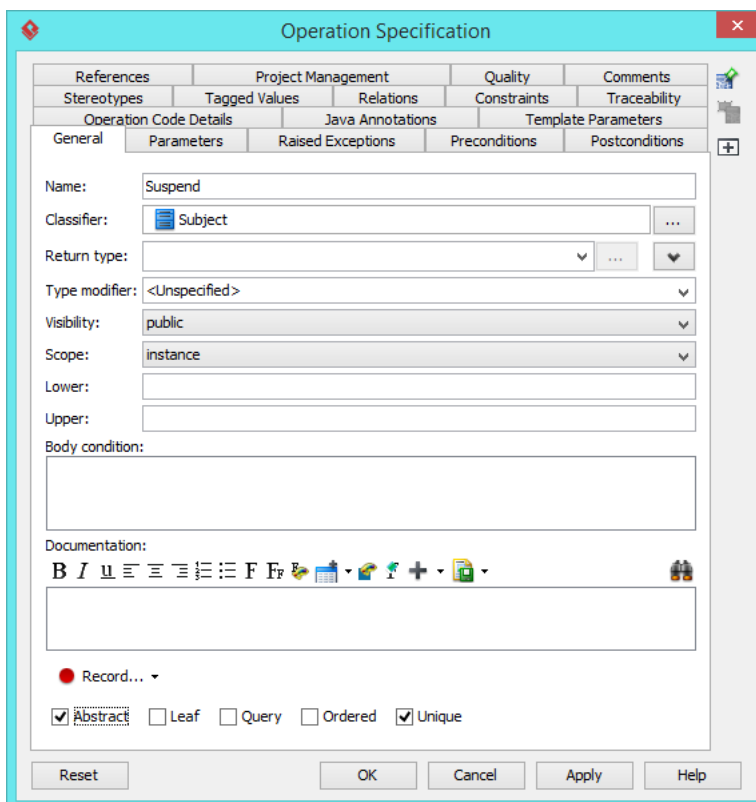
- Rename *Subject* to *Account*, and operation *Request* to *Create* at the bottom pane.



- 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.



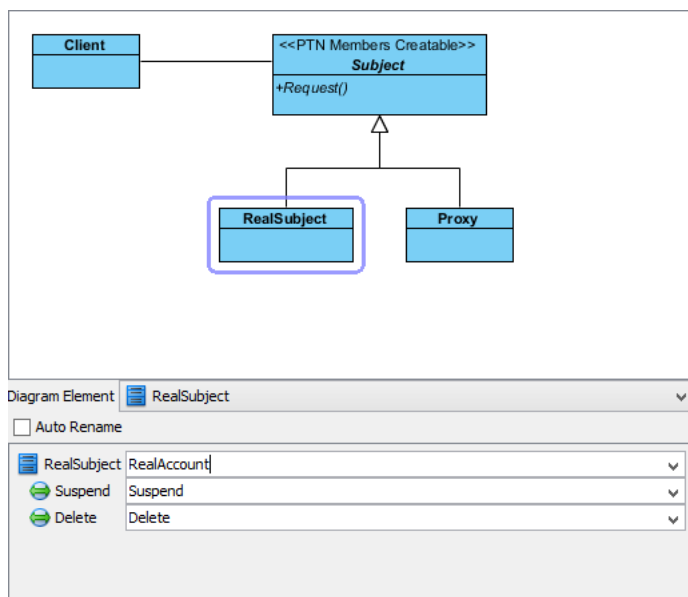
- In the **Operation Specification** dialog box, name the operation *Suspend*.



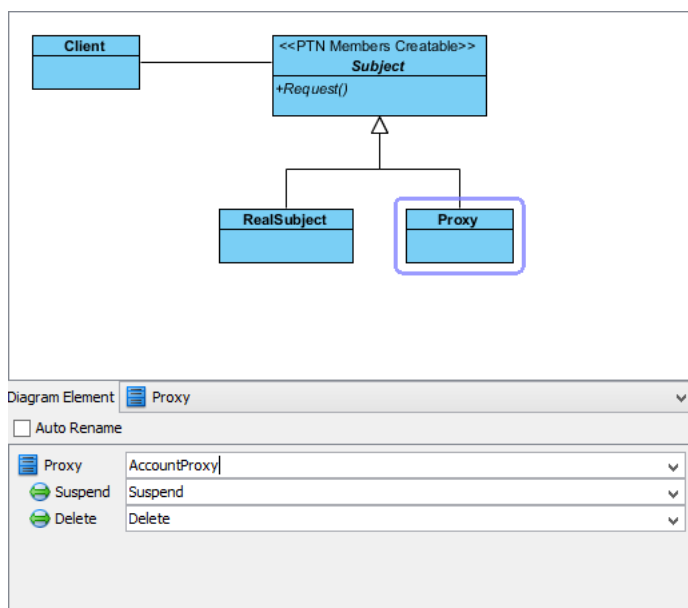
- Repeat steps 7 and 8 to create operation *Delete*.



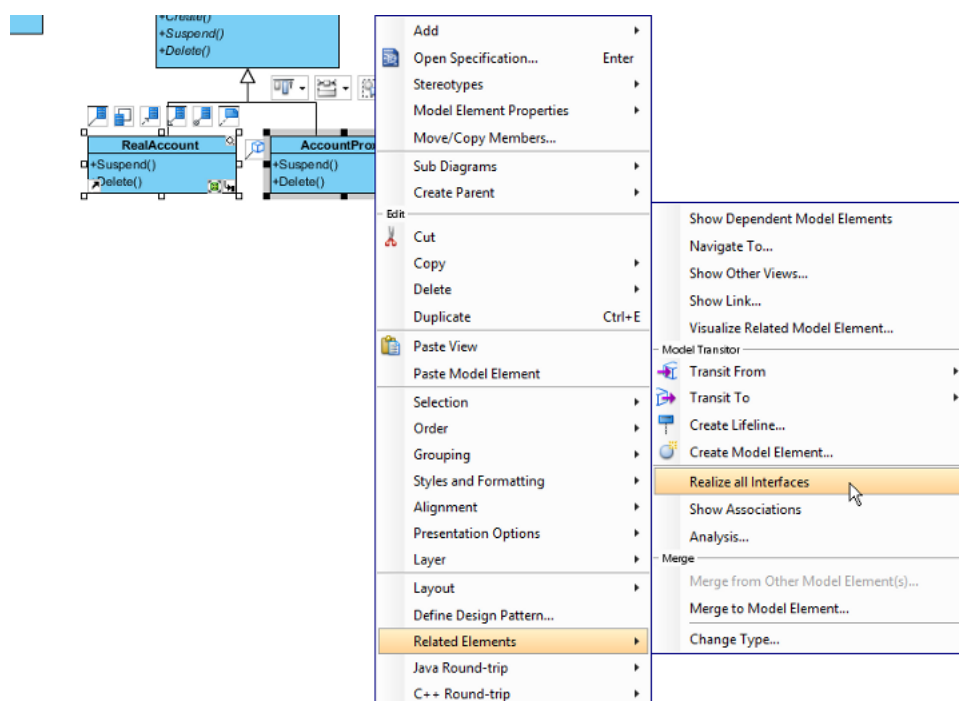
10. Select *RealSubject* in the overview, and rename it as *RealAccount* at the bottom pane.



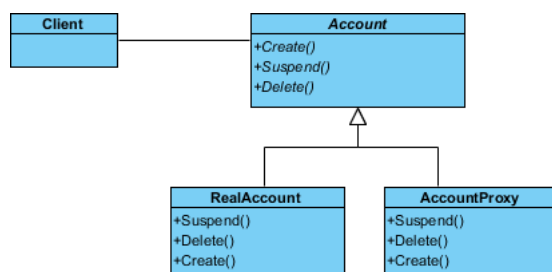
11. Select *Proxy* in the overview, and rename it as *AccountProxy* at the bottom pane. Click **OK** to apply the pattern to the diagram.



- 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

- [Proxy.pat](#)
- [Design Patterns.vpp](#)

#### Related Links

- [Full set of UML tools and UML diagrams](#)



[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/>)