Visual Paradigm

# Flyweight Pattern Tutorial

Written Date : October 14, 2009

This tutorial is aimed to guide the definition and application of Gang of Four (GoF)flyweight design pattern. By reading this tutorial, you will know how to develop a model for the flyweight 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 *Flyweight*.
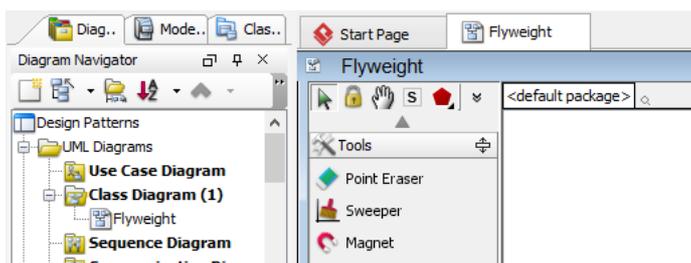
3.   Select **Class** from diagram toolbar. Click on diagram to create a class. Name it as *FlyweightFactory*.
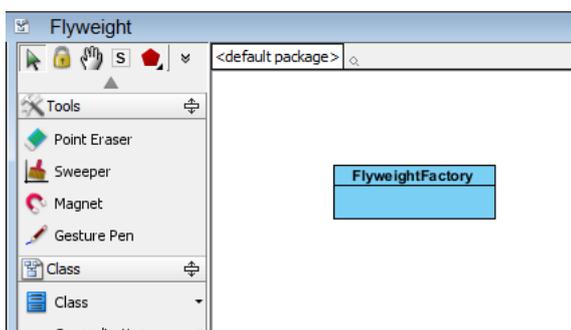
4. Right-click on the *FlyweightFactory* class, and select **Add** > **Operation** from the popup menu.
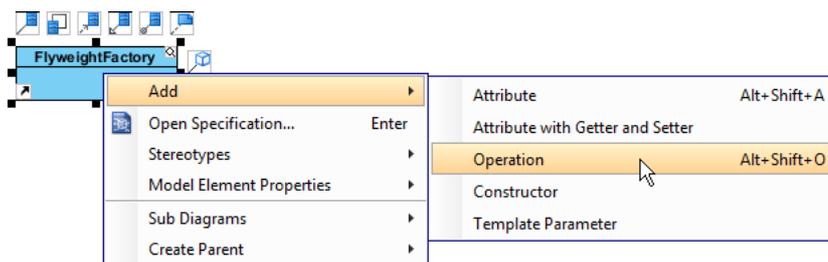
5. Name the operation *GetFlyweight(key)*.

6. Move the mouse cursor over the FlyweightFactory class, and drag out Aggregation > Class to create an aggregated class Flyweight.
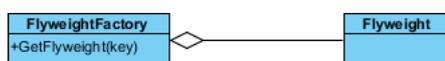
7. Create an operation in Flyweight, name it as Operation and takes an argument extrinsicState.

8. Right-click on *Flyweight*, and select **Model Element Properties** > **Abstract** to set it as abstract.
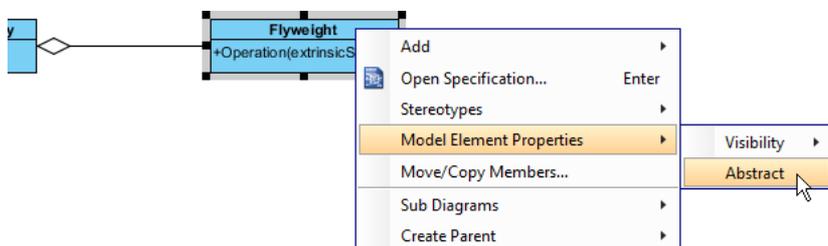
9. Right-click on *Operation* in *Flyweight*, and select **Model Element Properties** > **Abstract** to set it as abstract.
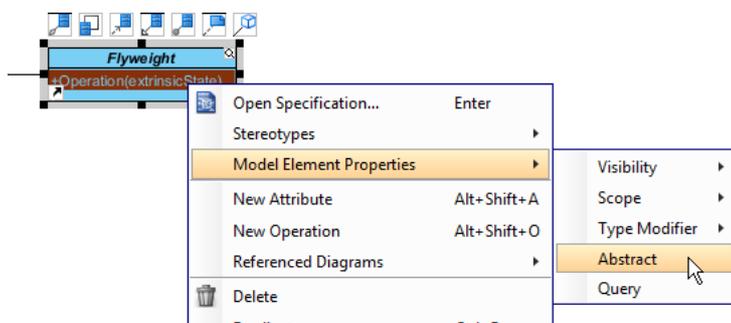
10. Move the mouse cursor over the *Subject* class, and drag out **Generalization** > **Class** to create subclasses *ConcreteFlyweight* and *UnsharedConcreteFlyweight*.
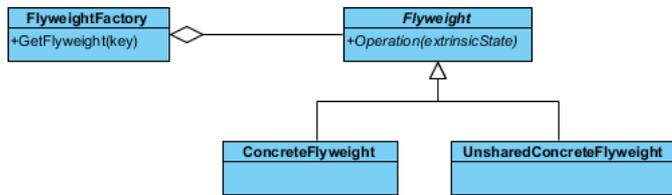
11. Make *ConcreteFlyweight* and *UnsharedConcreteFlyweight* inherit the abstract operations provided from *Flyweight* by right clicking on them, and selecting **Related Elements** > **Realize all Interfaces** from the popup menu.
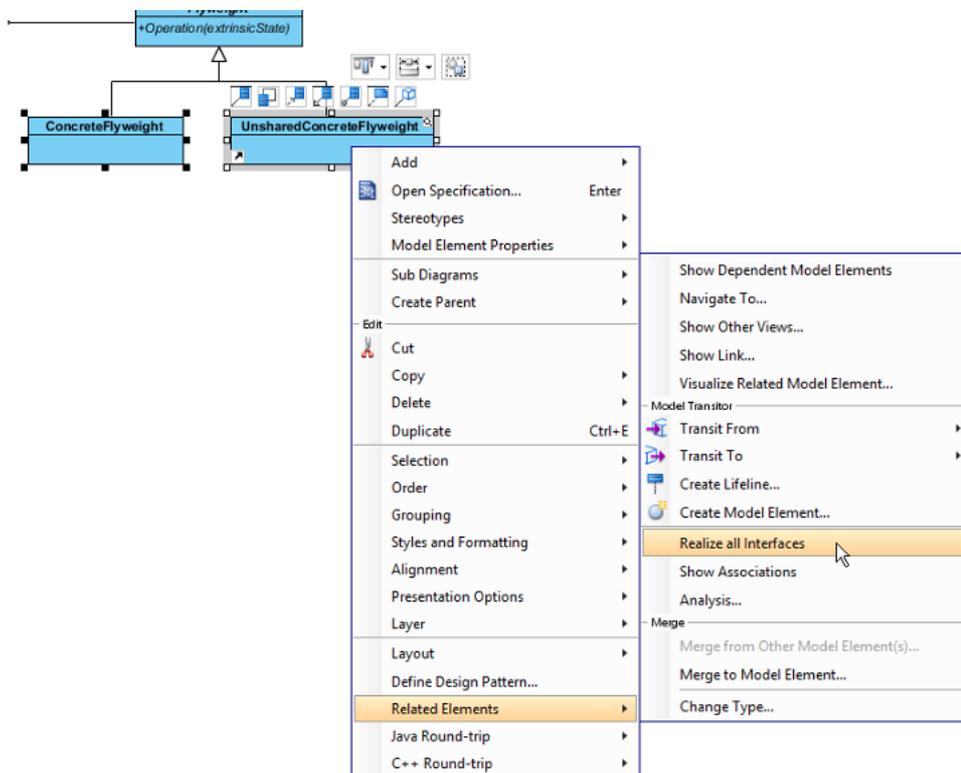
12. Add an operation to *ConcreteFlyweight* by right clicking on *ConcreteFlyweight*, and selecting **Add** > **Attribute** from the popup menu. Name the attribute *intrisicState*.
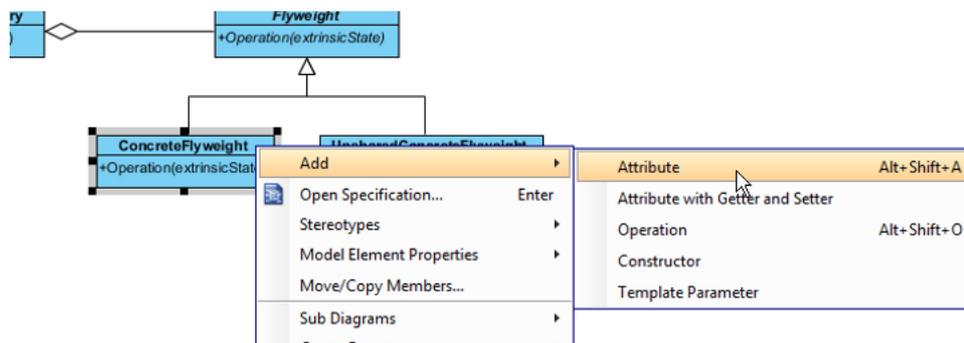

13. Repeat the previous step to add attribute *allState* to *UnsharedConcreteFlyweight*.
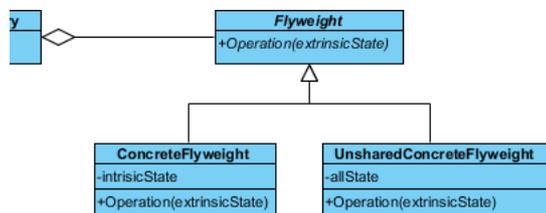

14. Create a class *Client* at the empty region of the diagram.
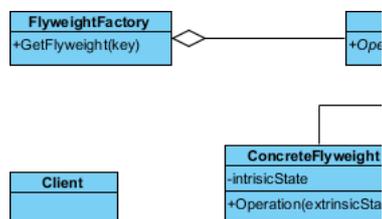
15. Make use of resource centric interface to associate *Client* and *FlyweightFactory*.
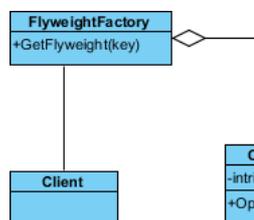
16.   Associate *Client* with *ConcreteFlyweight* and *UnsharedConcreteFlyweight*.
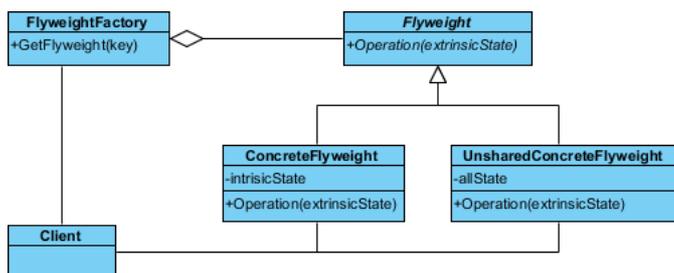

17.   In practice, there may be multiple *ConcreteFlyweight* classes. To represent this, stereotype the *ConcreteFlyweight* class as **PTN Cloneable**. Right-click on *ConcreteFlyweight* class and select **Stereotypes** > **Stereotypes...** from the popup menu.
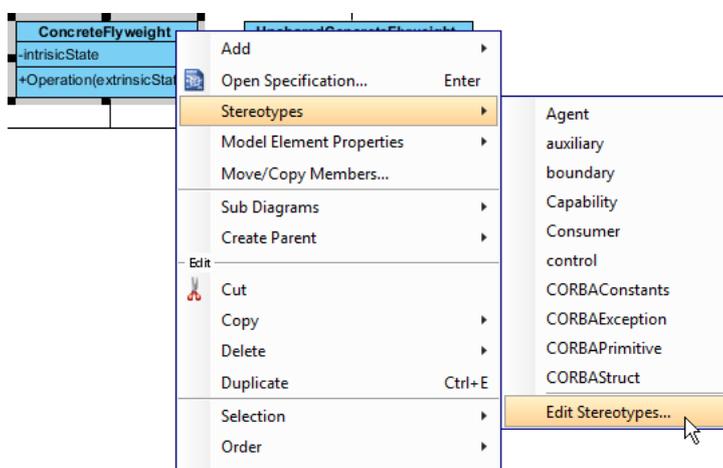
18. In the **Stereotypes** tab of class specification, select **PTN Cloneable** and click > to assign it to the class. Click **OK** to confirm.
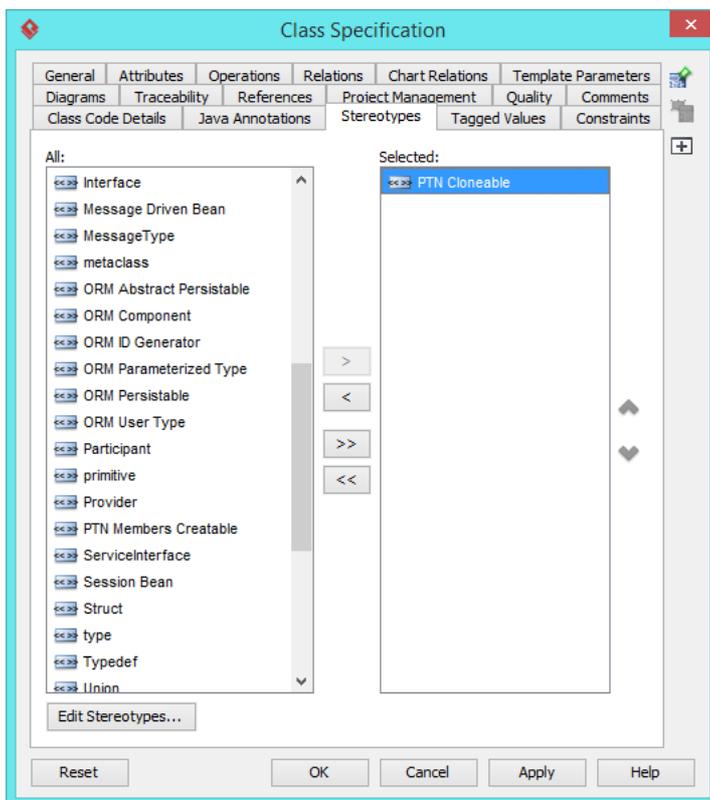
19. Repeat step 17 and 18 on *UnsharedConcreteFlyweight*.
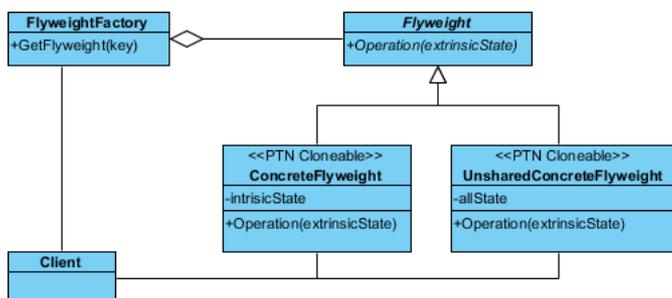
20. There maybe multiple operations in *Flyweight*. To represent this, stereotype the *Flyweight* class as **PTN Members Creatable**. Repeat steps 17 and 18 to stereotype *Flyweight* as **PTN Members Creatable**. Up to now, the pattern should look like this:
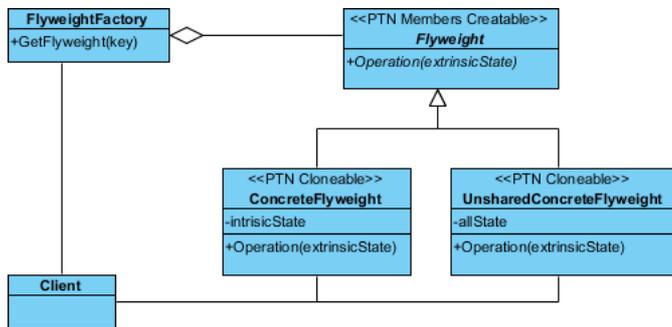

Defining Pattern

1. Select all classes on the class diagram.
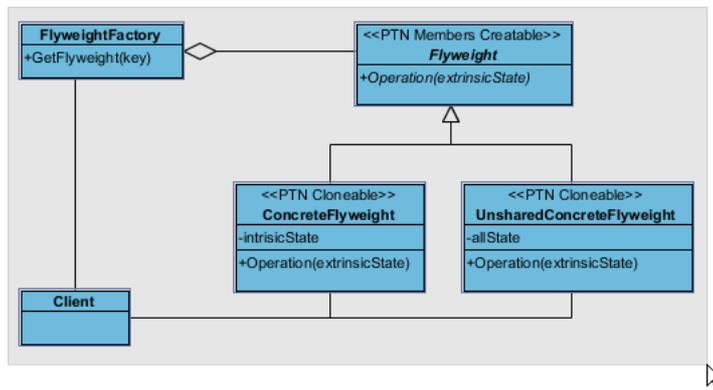
2.   Right-click on the selection and select **Define Design Pattern...** from the popup menu.
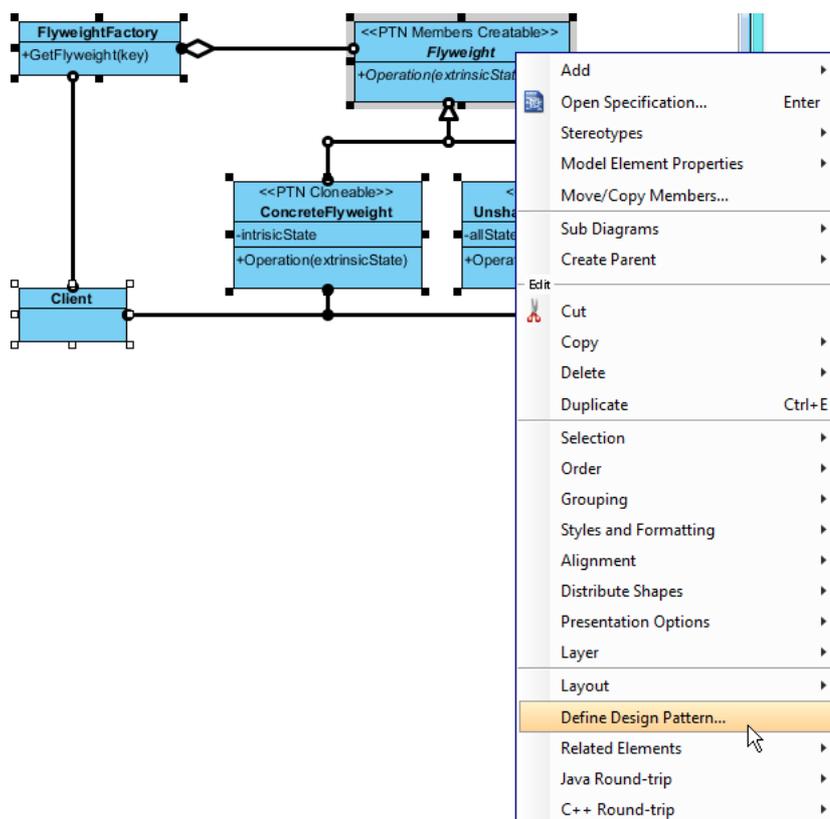
3.   In the **Define Design Pattern** dialog box, specify the pattern name *Flyweight*. Keep the file name as it. Click **OK** to procced.
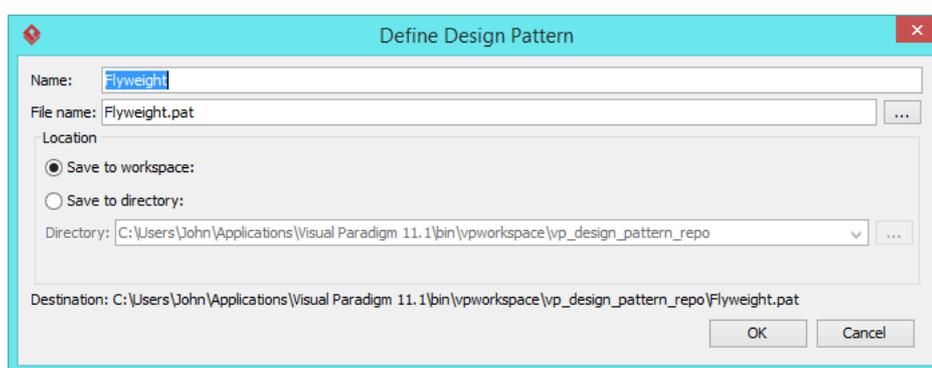
Applying Design Pattern on Class Diagram

In this section, we will try to make use of the flyweight pattern to model a part of a diagram editor.

1.   Create a new project *My Diagram Tool*.

2.   Create a class diagram *Domain Model*.

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

Open Specification...	
Add Shape	▸
Rename...	
Show Quality Checker Panel	
Synchronize to Entity Relationship Diagram	
Ignore Classes when Synchronizing...	
Paste View	
Paste Model Element	
Handi-Selection	▸
Diagram Content	▸
Connectors	▸
Presentation Options	▸
Layers...	
Zoom	▸
Layout	▸
Select in Tree	
Show Link...	
Utilities	▸
Print...	
Export	▸

Utilities submenu:
- Apply Design Pattern...
- Visual Diff...
- Create Matrix Diagram...
- Synchronize Classes Documentation to ERD...
- Repair Connector Ends...
- Repair Model Views

4.   In the **Design Pattern** dialog box, select *Flyweight* from the list of patterns.
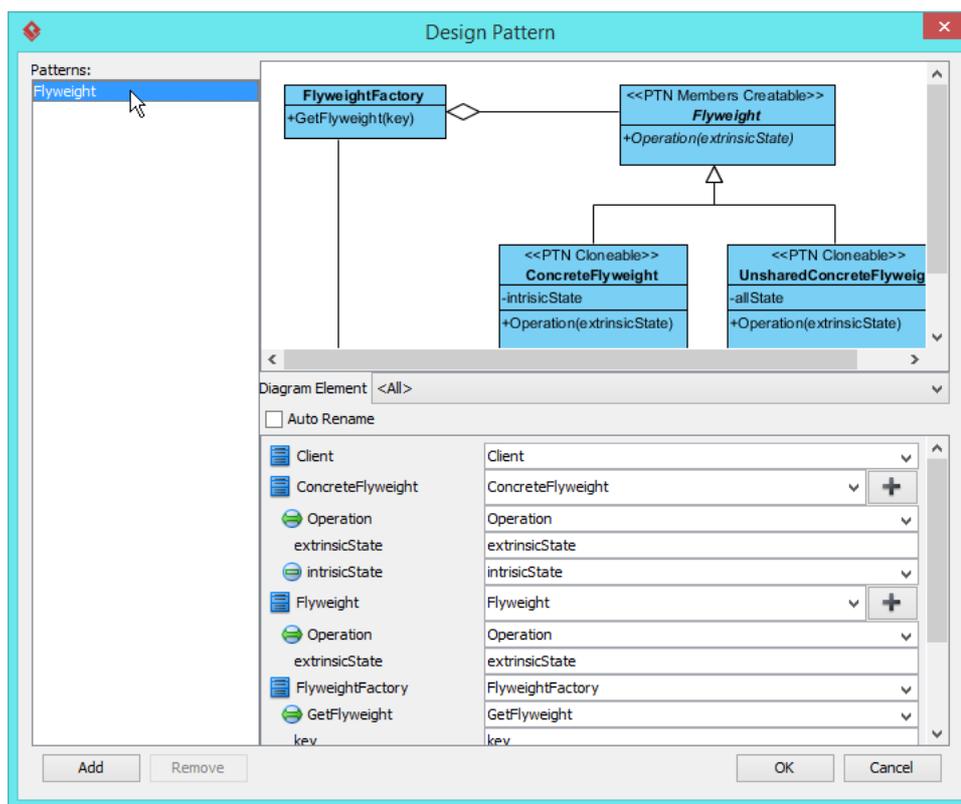
5.   Click on *Flyweight* in the overview.
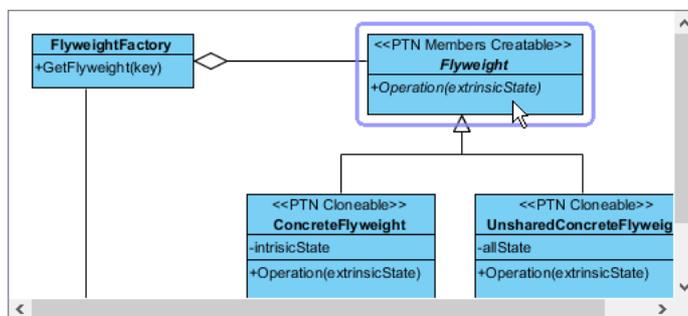

6.   Rename it to *Glyph* at the bottom pane. Rename operation *Operation* to *Draw*, and parameter *extrinsicState* to *gContext*.
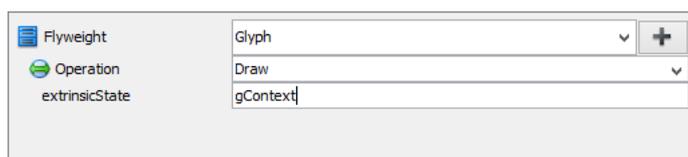
7. Select *ConcreteFlyweight* in overview. At the bottom pane, rename it to *Character*. Rename operation *Operation* to *Draw*, parameter *extrinsicState* to *gContext* and attribute *intrinsicState* to *char*.
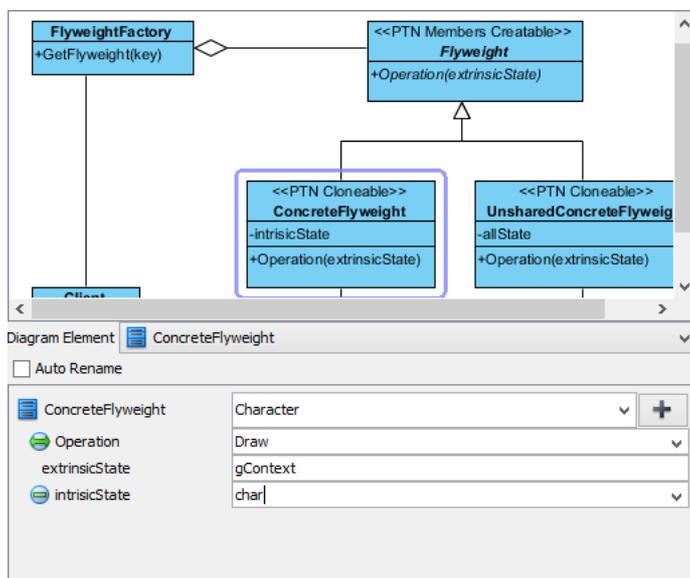

8. Select *UnsharedConcreteFlyweight* in overview. At the bottom pane, rename it to *Row*. Rename operation *Operation* to *Draw*, parameter *extrinsicState* to *gContext*.
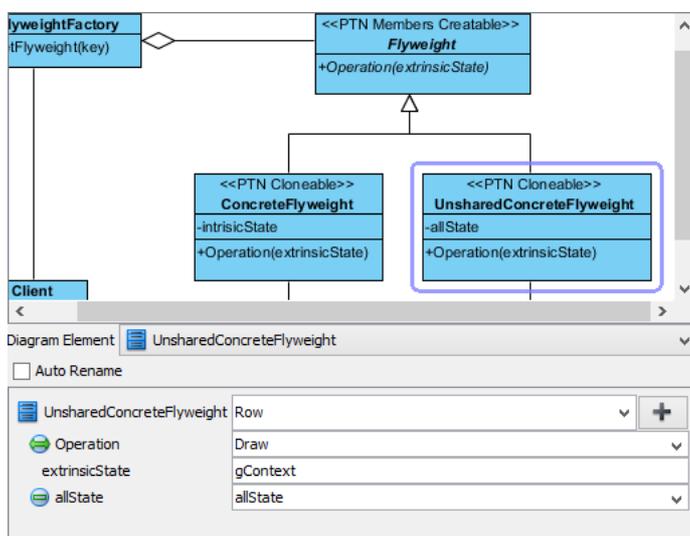
9. Select *FlyweightFactory* in overview. At the bottom pane, rename it to *GlyphFactory* and operation *GetFlyweight* to *GetGlyph*.
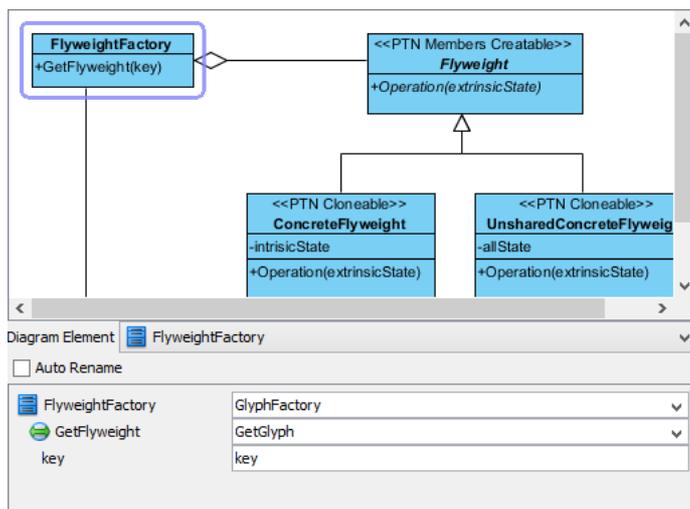
10. Click **OK** to confirm editing and apply the pattern to diagram.

11. Tidy up the diagram. It should become:
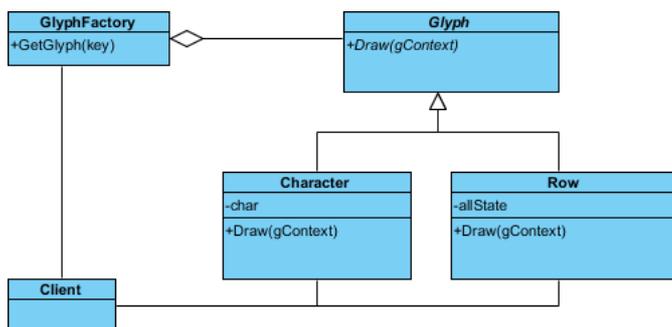
Resources
1. [Design Patterns.vpp](#)

2. [Flyweight.pat](#)

Related Links
*   [Full set of UML tools and UML diagrams](#)

Visual Paradigm

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

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