Welcome to
Software Engineering (IT314)

Lab 7
UML Diagrams Tool Demo

Jash Rathi, Mahir Shah

Course Instructor: Prof. Jayprakash Lalchandani

Dhirubhai Ambani Institute of Information and Communication Technology

DA-IICT

Visual Paradigm

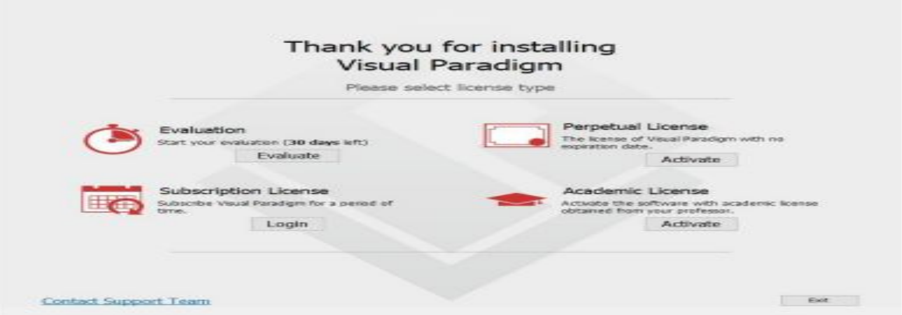
Download Link: https://www.visual-paradigm.com/download/

Getting Started

Starting Visual Paradigm

Selecting License Type

When you start Visual Paradigm the first time, you are asked to select a way to activate Visual Paradigm.



Thank you for installing Visual Paradigm

Please select license type

Evaluation

Start your evaluation (**30 days** left)

Perpetual License

The license of Visual Paradigm with no expiration date.

Subscription License

Subscribe Visual Paradigm for a period of time.

Academic License

Activate the software with academic license obtained from your professor.

Depending on whether you own a purchased copy or an evaluation copy of Visual Paradigm, you can proceed by following the steps below respectively:

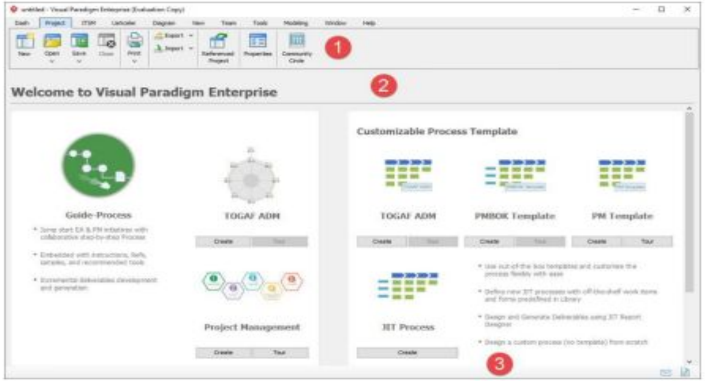
For Customers

You should receive a notification Email with an activation code. The same activation code can also be found from the license key listed in your customer account. Copy the activation code first, click on **Activate** under **Perpetual License**, paste the copied activation code and then click **Activate** to continue.

For Evaluation

If you want to evaluate Visual Paradigm, click **Evaluate**.

The Environment



- Jump start EA & PM initiatives with collaborative step-by-step Process
- Embedded with instructions, refs, samples, and recommended tools
- Incremental deliverables development and generation

- Use out-of-the-box templates and customize the process flexibly with ease
- Define new JIT processes with off-the-shelf work items and forms predefined in Library
- Design and Generate Deliverables using JIT Report Designer
- Design a custom process (no template) from scratch

No.	Name	Description
1	Toolbar	A tabbed toolbar that allows you to perform various operations in Visual Paradigm.
2	Diagram editor	The diagram will be displayed in diagram editor.
3	Status bar	Notifications are shown here. You can also open the message pane and description pane from the bottom right of the status bar.

Saving and Opening Project

To save your work, select either **Project > Save** or **Project > Save as...**. When you are saving a project for the first time, you will be asked to specify its location.

To open an existing project, select **Project > Open** from the toolbar and select the project to open.

Basic Diagramming Techniques

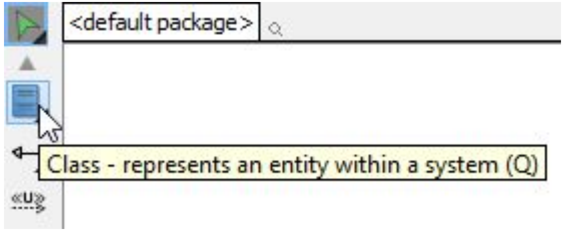
Compatible editions: All

This section will go through the steps of creating diagrams, creating shapes and connecting them. You will also learn how to document model elements and make diagrams more readable by coloring shapes.

How to Draw Class Diagram?

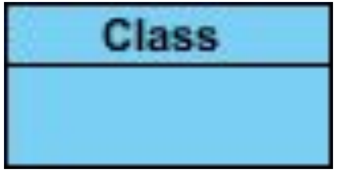
Creating class

1. Select **Diagram > New** from the application toolbar.
2. In the **New Diagram** window, select **Class Diagram**.
3. Click **Next**.
4. Enter the diagram name and description. The **Location** field enables you to select a model to store the diagram.
5. Click **OK**.



Class - represents an entity within a system (Q)

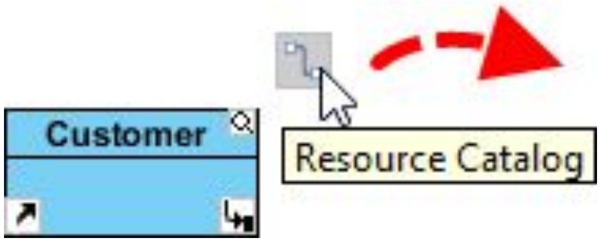
A class will be created.



Creating association

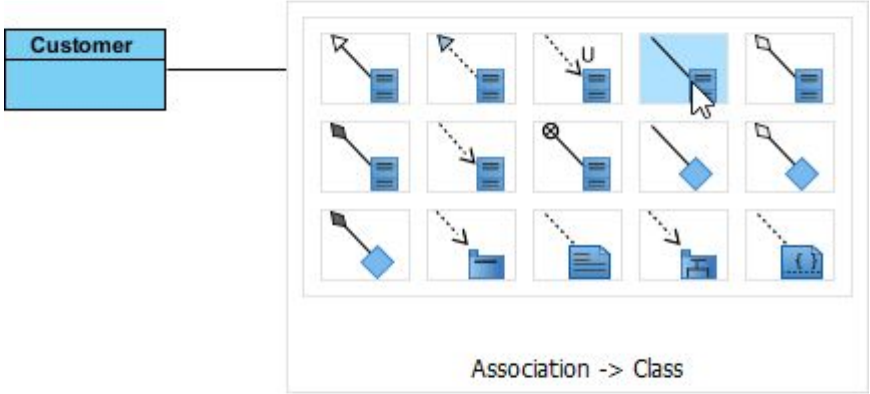
To create an associated class in a class diagram:

1. Move your mouse pointer over the source shape.



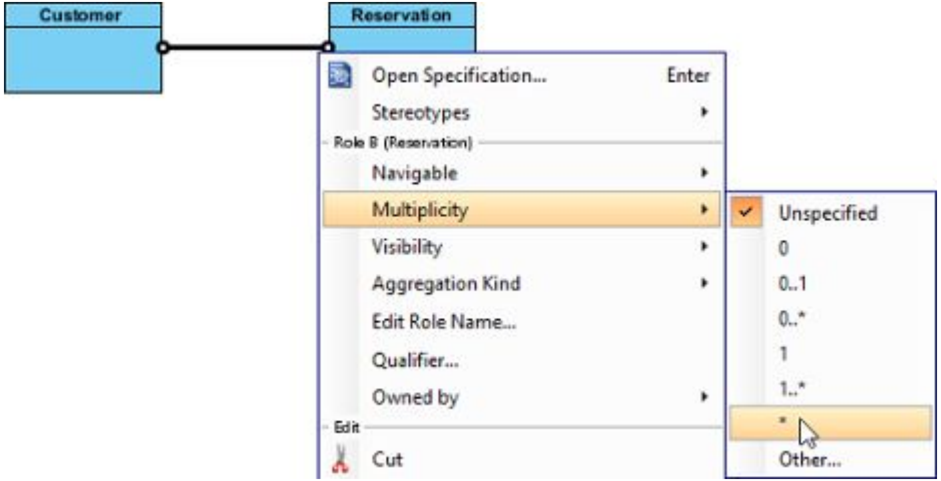
Class Continue.

If you are connecting to an existing class, select **Association** from Resource Catalog. If you are creating a new class, select **Association -> Class** from Resource Catalog. If you want to create an aggregation or composition, select **Aggregation -> Class** or **Composition -> Class** instead.



Add the multiplicity

To edit multiplicity of an association end, right-click near the association end, select **Multiplicity** from the popup menu and then select a multiplicity.

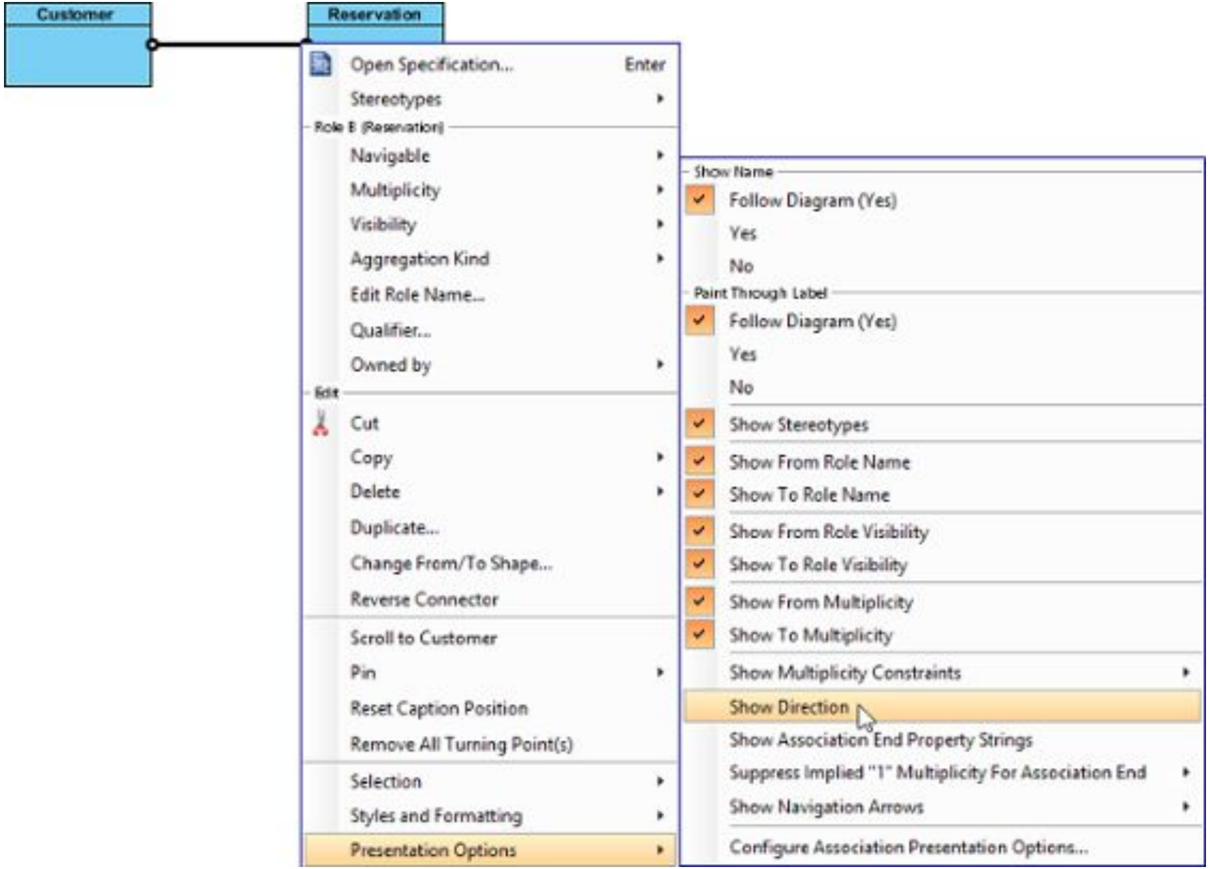


Association

To show the direction of an association,

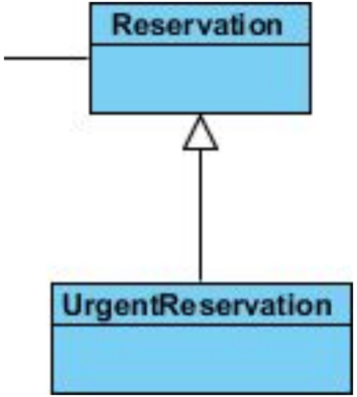
- right click on it and
- select

Presentation Options > Show Direction



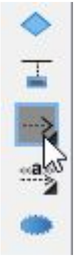
Generalization

If you are creating a new class, you should see the class now and it is connected to the source shape with a generalization. Enter its name and press **Enter** to confirm editing.

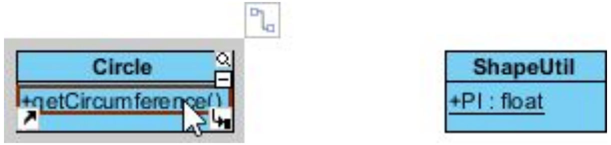


Creating dependency from/to attribute/operation

1. Select **Dependency** from the diagram toolbar.

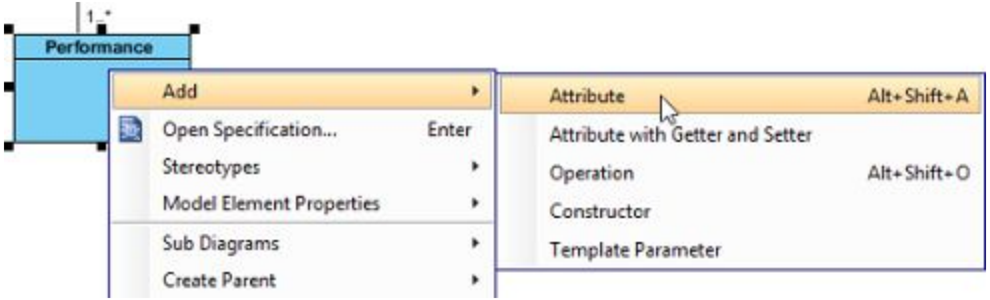


2. Press on the source shape or a class member.

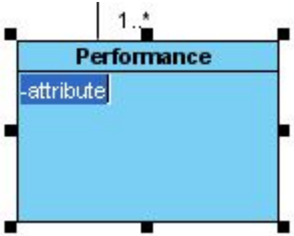


Creating attribute

To create attribute, right click the class and select **Add** > **Attribute** from the pop-up menu.

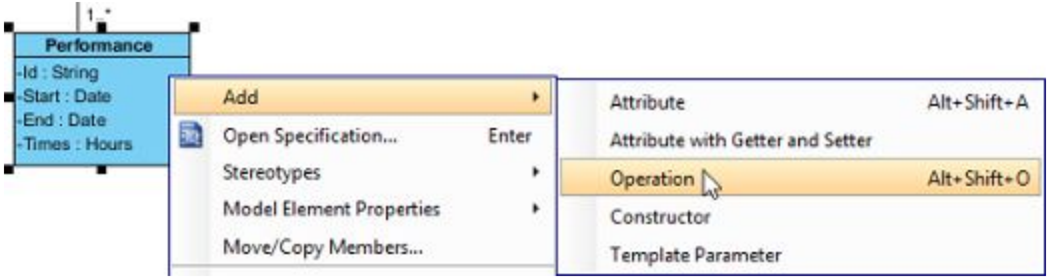


An attribute is created.

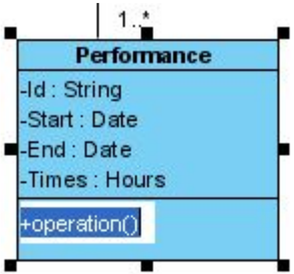


Creating operation

To create operation, right click the class and select **Add** > **Operation** from the pop-up menu.

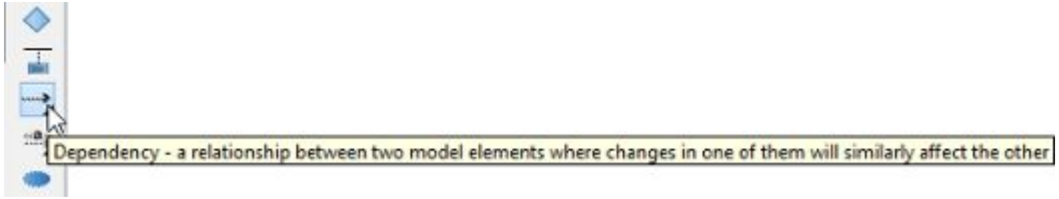


An operation is created.



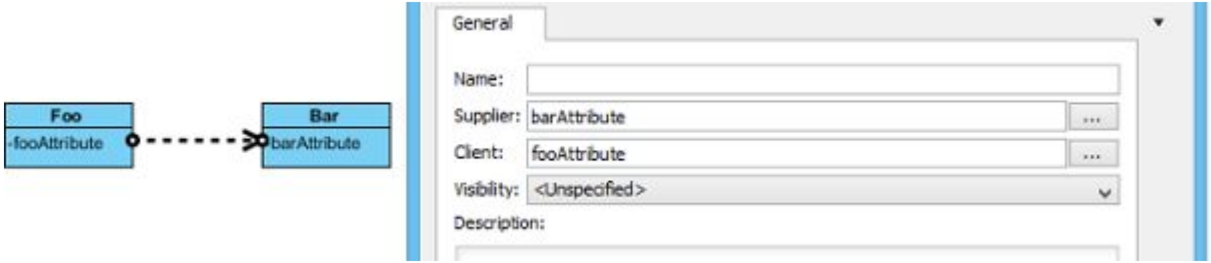
Relation

Select the type of relationship to be created, under the diagram toolbar.



Dependency - a relationship between two model elements where changes in one of them will similarly affect the other

While it looks like the connector is connecting the classes but not the members,

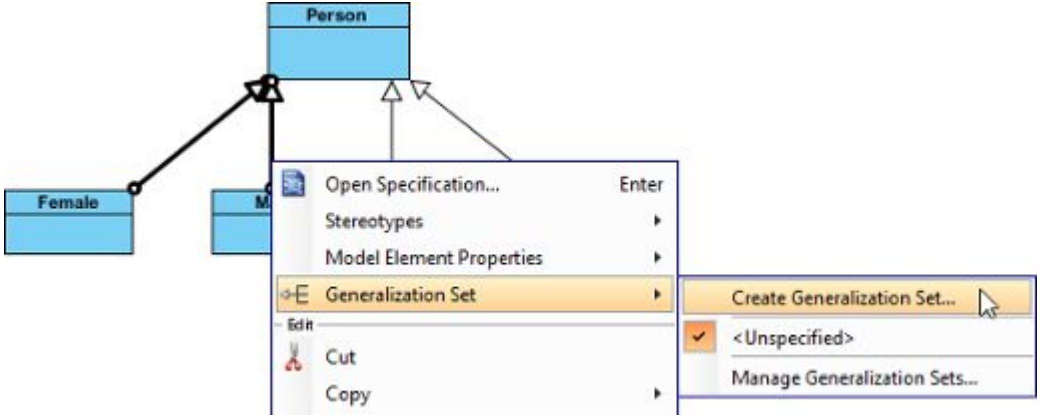


Name:
Supplier: barAttribute
Client: fooAttribute
Visibility: <Unspecified>
Description:

Generalization set

A generalization set defines a particular set of generalization relationships that describe the way

in which a general classifier (or superclass) may be divided using specific subtypes. To define a generalization set, select the generalizations to include, right click and select Generalization set > Create Generalization Set... from the popup menu.



How to Draw Component Diagram?

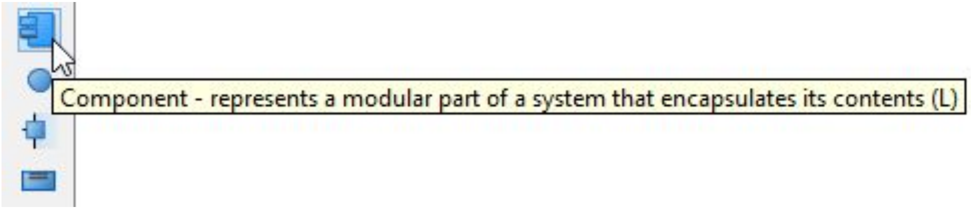
Creating component diagram

Perform the steps below to create a UML component diagram in Visual Paradigm.

1. Select **Diagram > New** from the application toolbar.
2. In the **New Diagram** window, select **Component Diagram**.
3. Click **Next**.
4. Enter the diagram name and description. The **Location** field enables you to select a model to store the diagram.
5. Click **OK**.

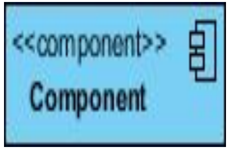
Creating component

To create component in component diagram, click **Component** on the diagram toolbar and then click on the diagram.

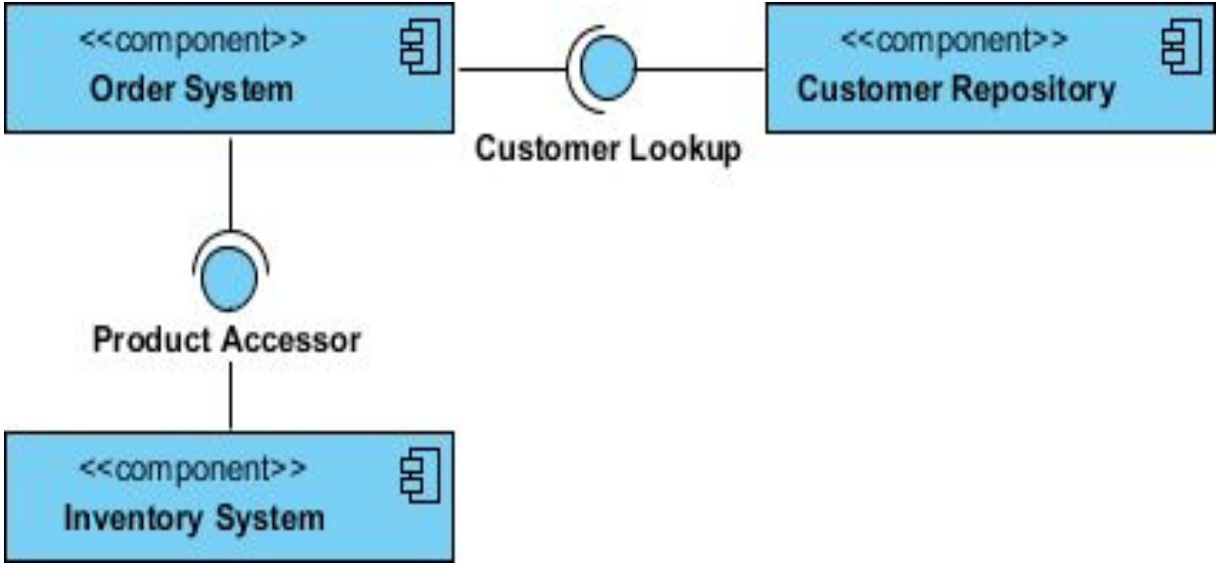


Component - represents a modular part of a system that encapsulates its contents (L)

A component will be created.



Example


<<component>>
Order System
Customer Lookup
<<component>>
Customer Repository
Product Accessor
<<component>>
Inventory System

How to Draw Deployment Diagram?

Creating deployment diagram

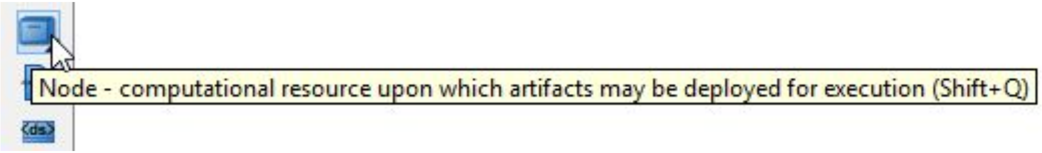
Perform the steps below to create a UML deployment diagram in Visual Paradigm.

1. Select **Diagram > New** from the application toolbar.
2. In the **New Diagram** window, select **Deployment Diagram**.
3. Click **Next**.
4. Enter the diagram name and description. The **Location** field enables you to select a model to store the diagram.
5. Click **OK**.

Deployment Diagram

Creating node

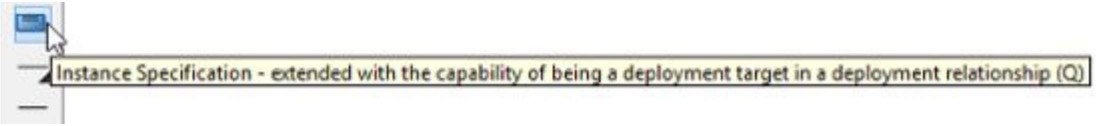
To create node in deployment diagram, click **Node** on the diagram toolbar and then click on the diagram.



Node - computational resource upon which artifacts may be deployed for execution (Shift+Q)

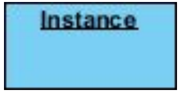
Creating instance of node

To create instance of node, click **Instance Specification** on the diagram toolbar and then click on the diagram.



Instance Specification - extended with the capability of being a deployment target in a deployment relationship (Q)

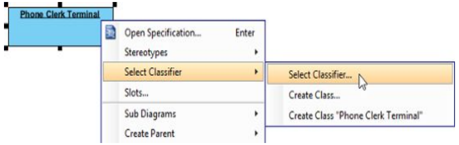
An instance specification will be created.



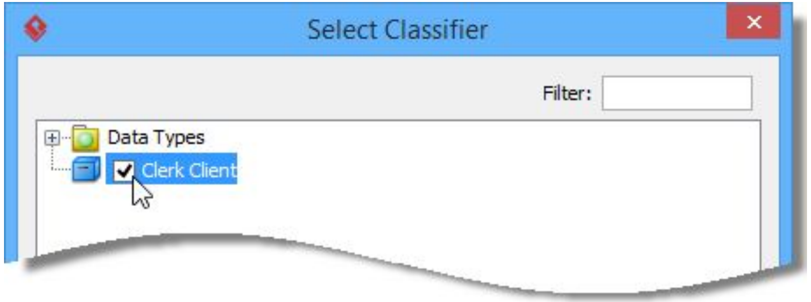
Selecting Classifier

Selecting classifiers

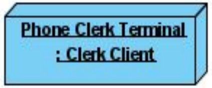
To specify classifiers for an instance specification, right-click it and select **Select Classifier** > **Select Classifier...** from the pop-up menu.



When the **Instance Specification Specification** window pops out, the **Classifiers** tab is opened by default. Click **Add...**. Then, select the classifier(s) in the popup window and click **OK**.

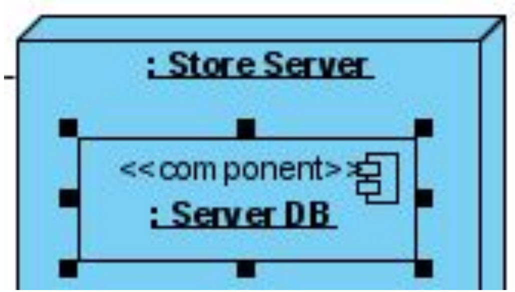


Click **OK** button to close the specification window. The selected classifiers are assigned to the instance specification.



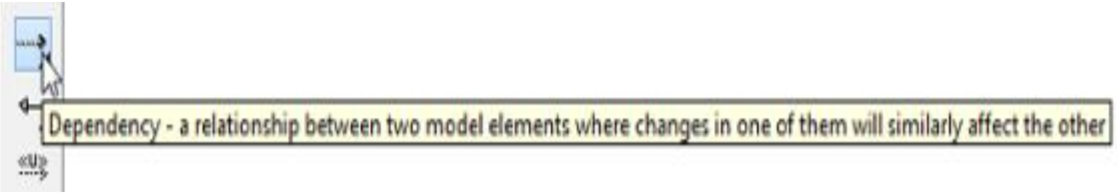
Creating instance of component

Similar to creating instance of node, you first create a component model element and then create an instance specification. However, this time assigns a component to the instance specification as classifier. After that the instance specification will be displayed as a component.



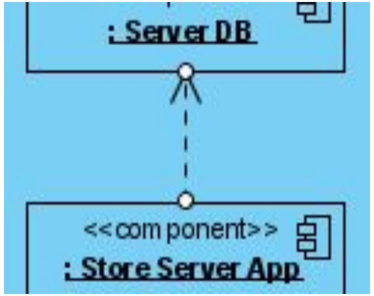
Creating dependency

To create dependency, click **Dependency** on the diagram toolbar.



Dependency - a relationship between two model elements where changes in one of them will similarly affect the other

Drag from the source shape, move the mouse over the target shape and then release the mouse button to create the dependency.



How to Draw Object Diagram?

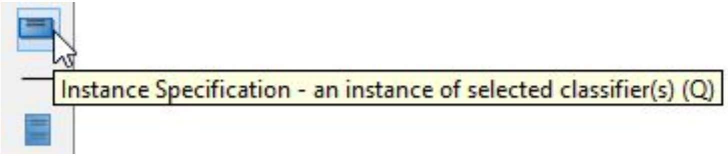
Creating object diagram

Perform the steps below to create a UML object diagram in Visual Paradigm.

1. Select **Diagram > New** from the application toolbar.
2. In the **New Diagram** window, select **Object Diagram**.
3. Click **Next**.
4. Enter the diagram name and description. The **Location** field enables you to select a model to store the diagram.
5. Click **OK**.

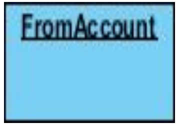
Creating instance specification

To create instance specification in object diagram:



Instance Specification - an instance of selected classifier(s) (Q)

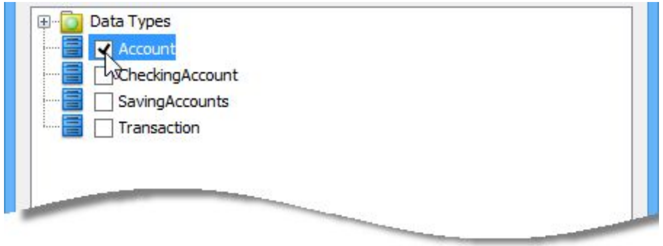
Click on the diagram to create an instance specification shape. Name it.



Selecting classifiers

To specify classifiers for an instance specification:

1. Right-click on the desired instance specification shape and select **Select Classifier** > **Select Classifier...** from the pop-up menu.
2. This opens the **Classifiers** tab. Click **Add...** in it.
3. In the **Select Classifier** window, select the class(es) to be the classifier of the instance specification. If you are referencing another project, you can select its model element to be the classifier. Just change the **from project** selection at the top of the window.



Data Types

☑ Account

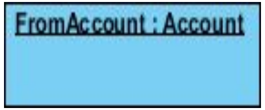
□ CheckingAccount

□ SavingAccounts

□ Transaction

4. Click **OK** to return to the **Instance Specification Specification** window.

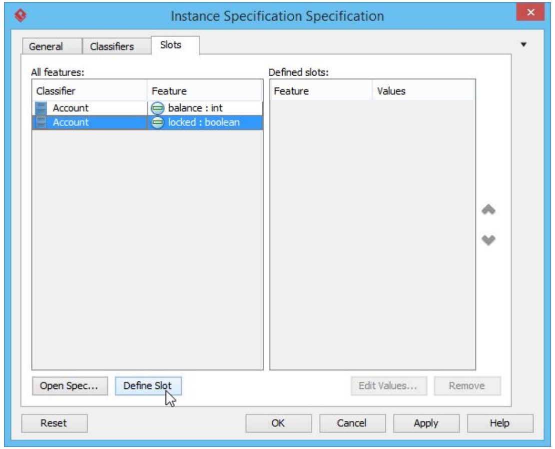
5. Click **OK** to return to the diagram



Defining slots

To define slots for an instance specification:

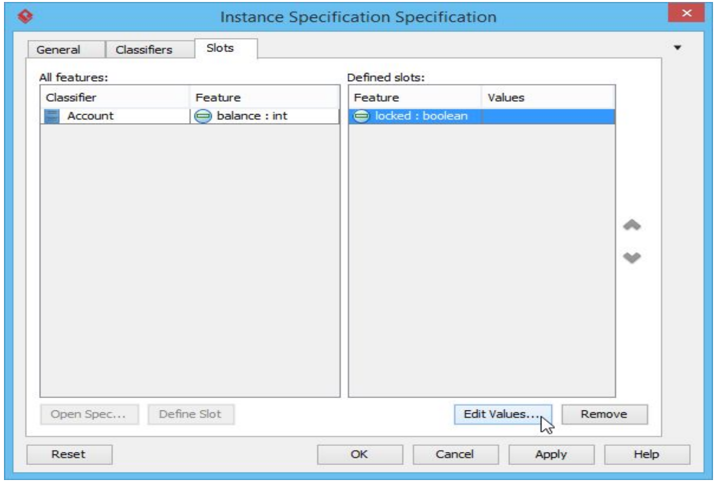
1. Right-click on the desired instance specification shape and select **Slots...** from the pop-up menu.
2. The **Instance Specification Specification** window appears with the **Slots** tab selected. Select the features that you want to define slots on the left and click **Define Slot**.



Classifier	Feature
Account	balance : int
Account	locked : boolean

Feature	Values

3. Select a defined slot and click **Edit Values...** at bottom right



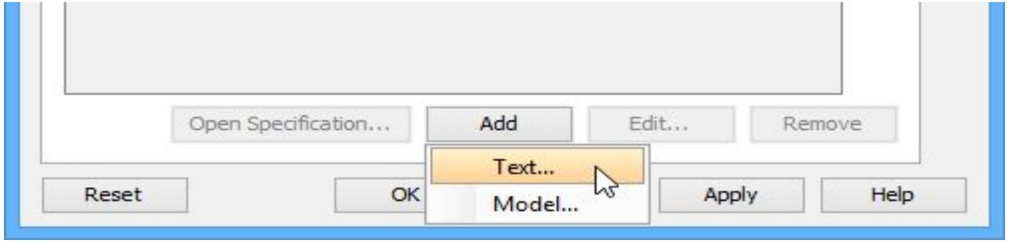
All features:

Classifier	Feature
Account	balance : int

Defined slots:

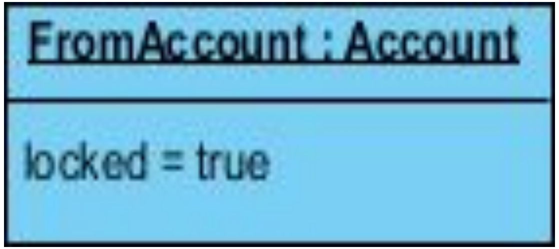
Feature	Values
locked : boolean	

4. The **Slot Specification** window pops out, the **Values** tab is opened by default. Click **Add** button and select **Text** from the pop-up menu.



5. Enter the slot value and click **OK** to confirm.

6. Click **OK** again in the **Instance Specification Specification** window to return to the diagram



How to Draw package Diagram?

Create Packages for your diagrams

First, we need to have our packages ready. To create packages:

1. To create a Package Diagram, select **Diagram > New** from the toolbar.
2. In the **New Diagram** window, select **Package Diagram** and click **Next**.
3. Enter *Racing Game Packages* as diagram name and click **OK** to confirm.

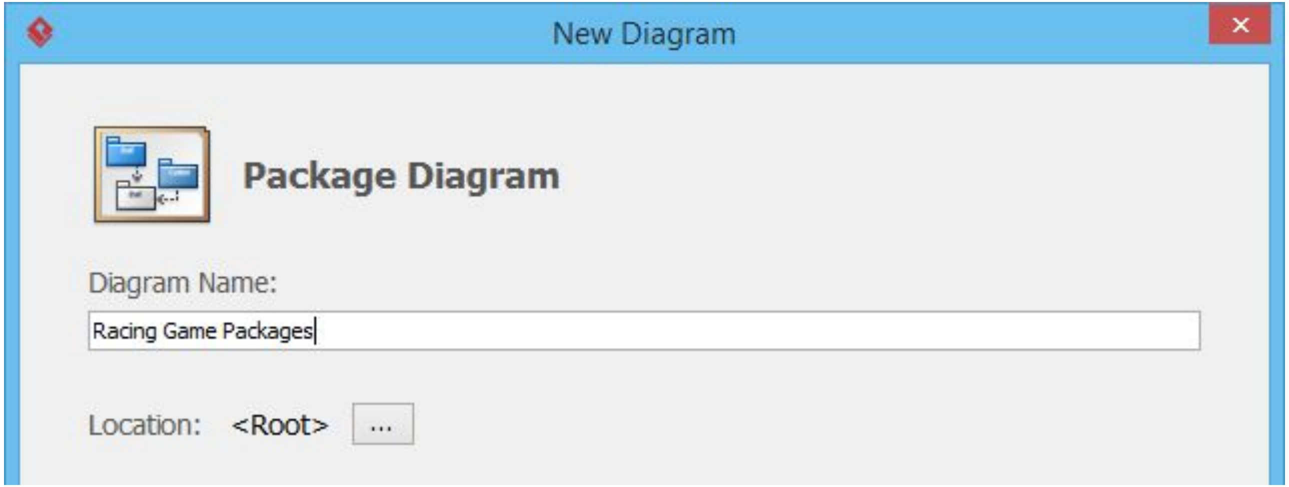
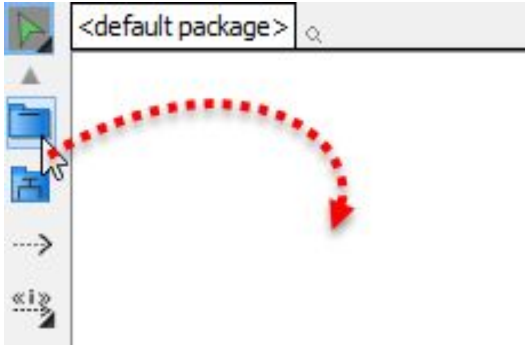


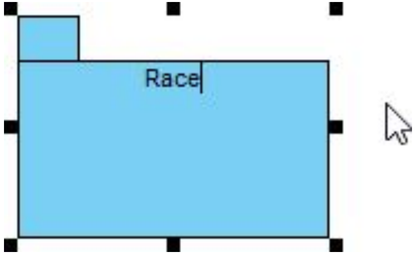
Diagram Name: Racing Game Packages

Location: <Root>

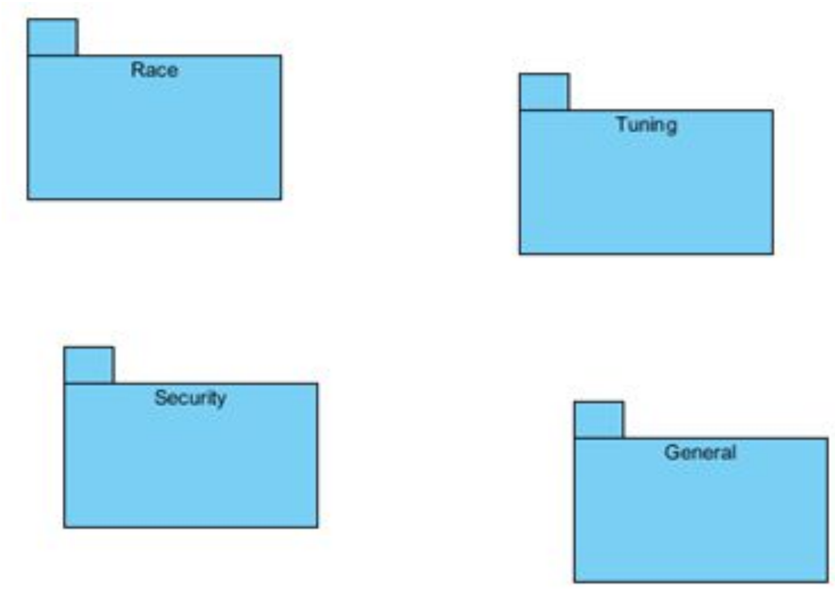
4. Click the **Package** button in diagram tool bar, then click on the blank area of the diagram to create the package.



5.Name the package.

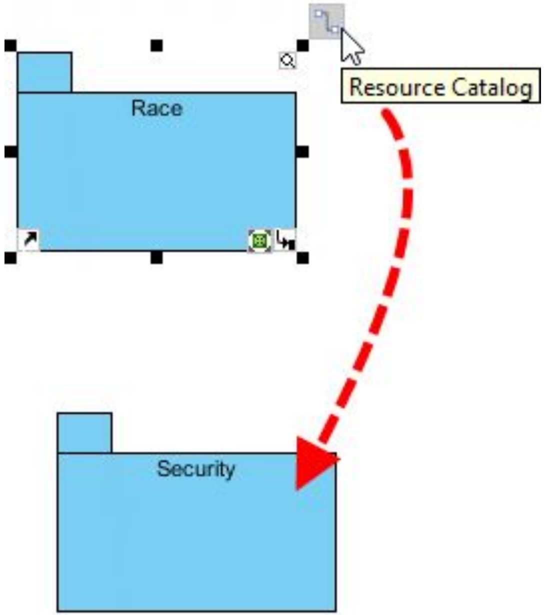


6. Repeat the steps to create other packages, namely, *Tuning*, *Security* and *General* respectively.

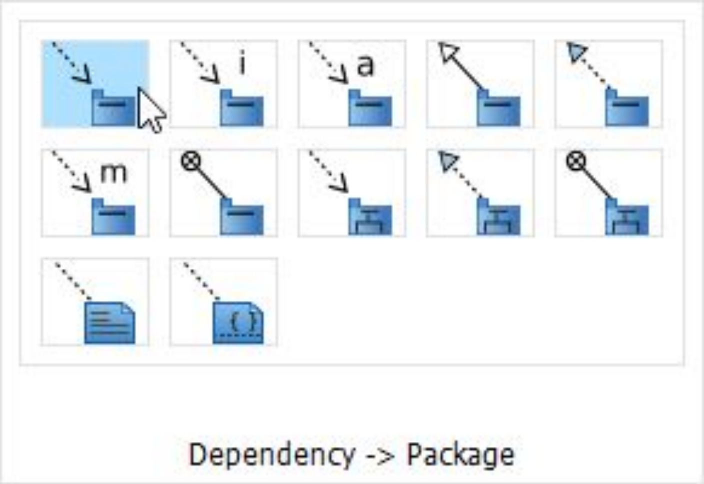


Define relationships between package

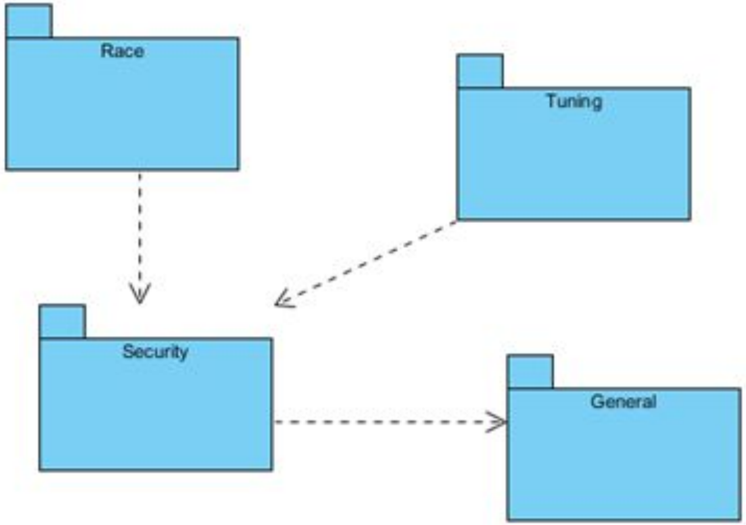
1. Click on the *Race* package to show up its resource icon.
2. Drag out the **Resource Catalog** button and drop it at the *Security* package.



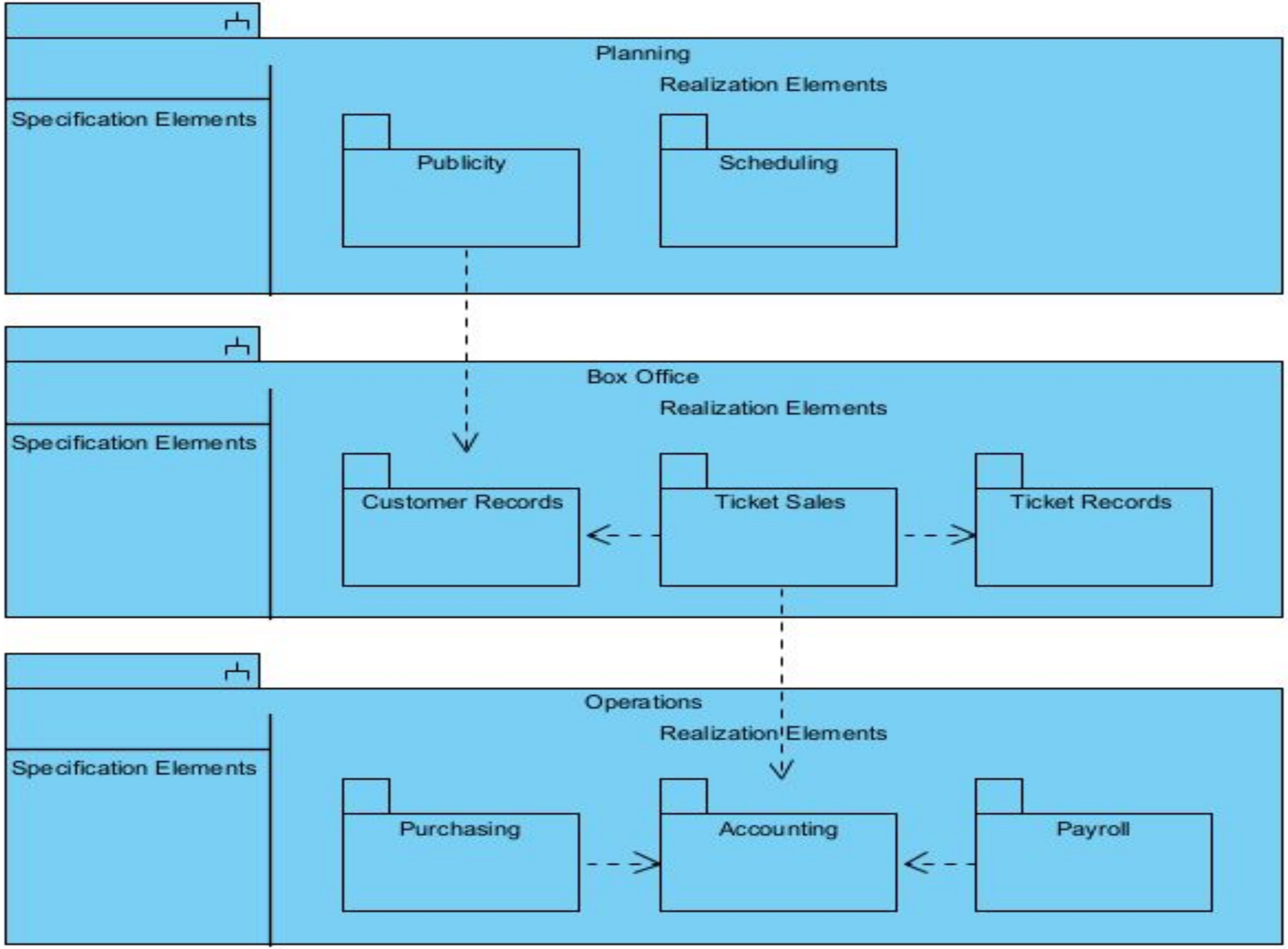
3. Select **Dependency -> Package** in Resource Catalog.



4. Repeat the steps to create dependency from *Tuning* to *Security* and also from *Security* to *General*.



Domain - Ticket selling system

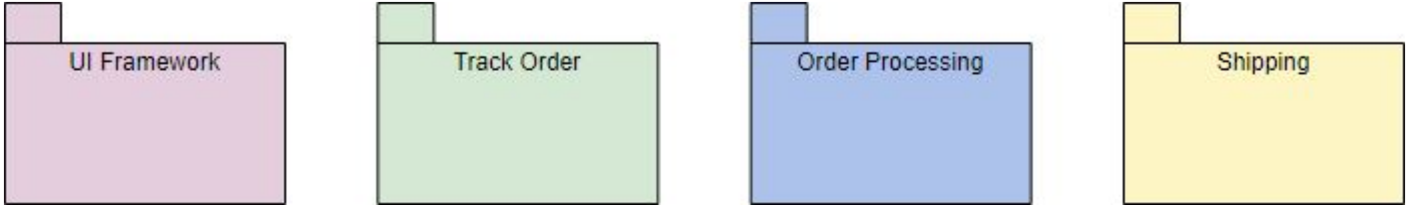

Planning
Specification Elements
Realization Elements
Publicity
Scheduling
Box Office
Specification Elements
Realization Elements
Customer Records
Ticket Sales
Ticket Records
Operations
Specification Elements
Realization Elements
Purchasing
Accounting
Payroll

Track Order Service for an online shopping store.

- Track Order Service is responsible for providing tracking information for the products ordered by customers.
- Customer types in the tracking serial number, Track Order Service refers the system and updates the current shipping status to the customer.

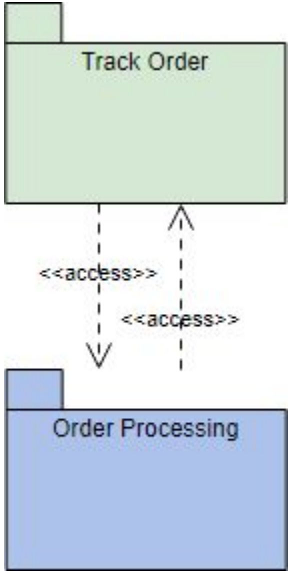
Step 1 - Identify the packages present in the system

1. There is a **"track order"** service, it has to talk with other module to know about the order details, let us call it **"Order Processing"**.
2. Next after fetching Order Details it has to know about shipping details, let us call that as **"Shipping"**.
3. Finally if knows the status of the order it has to update the information to the user, let us call this module as **"UI Framework"**.

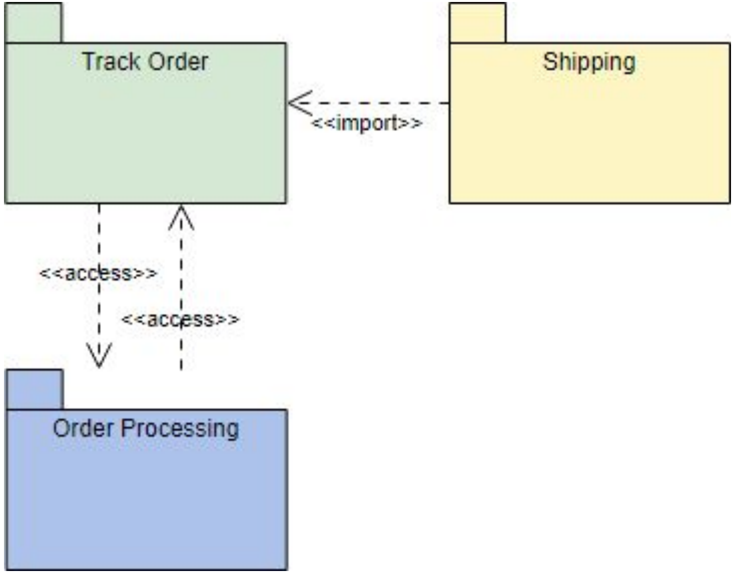


Step 2 - Identify the dependencies

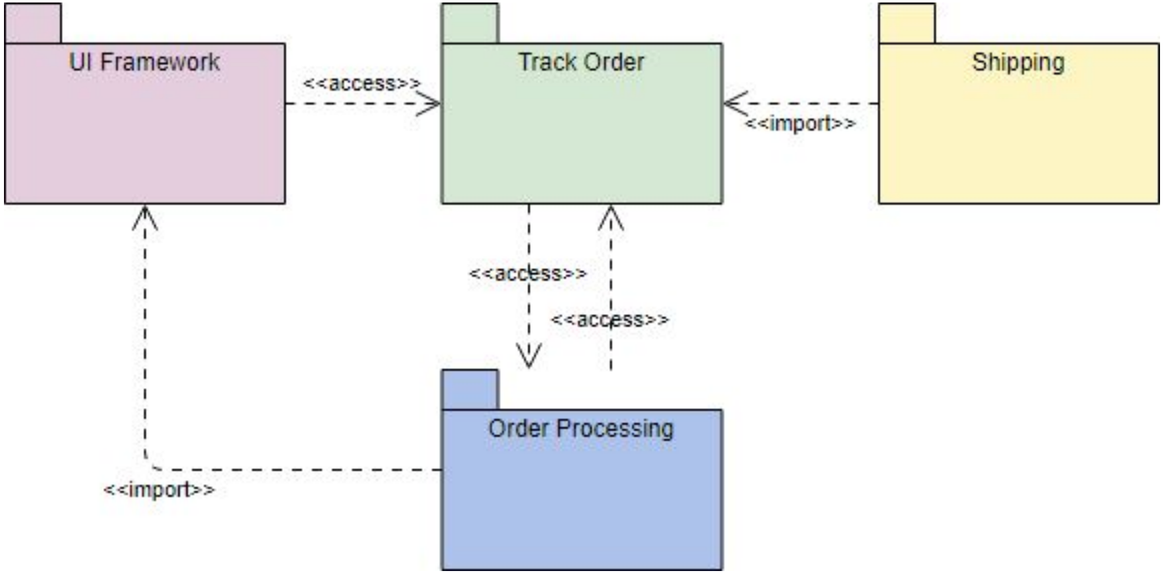
1. **"Track order"** package should get order details from **"Order Processing"** and on the other hand, "Order Processing" also requires the tracking information from the "Track Order" package, thus, the two modules are accessing each other which suffices <<access>> dual dependency.



To know shipping information, "Shipping" requires to import "Track Order" to complete the shipping process.



Step 3 - Finally, Track Order dependency to UI Framework is also mapped in to the diagram which completes the Package Diagram for Track Order subsystem.



Java Code Example

```
package Shipping;

public class shippinng {

public static void main(String[] args)
{
                // TODO
Auto-generated method stub

        }

}
```

```
package Orderprocessing;
import UIFramework.*;
import TrackOrder.*;
public class orderprocessing {

public void msg()
{
System.out.println("Orderprocessing ");
}

public static void main(String[] args) {
trackOrder obj = new trackOrder();
       obj.msg();

        }

}
```

Java Code Sample

```
package UIFramework;
import TrackOrder.*;

public class uiframework {

public void msg()
{
System.out.println("uiframework");
}
public static void main(String[] args)
{

trackOrder obj = new trackOrder();
obj.msg();
    }


          }
```

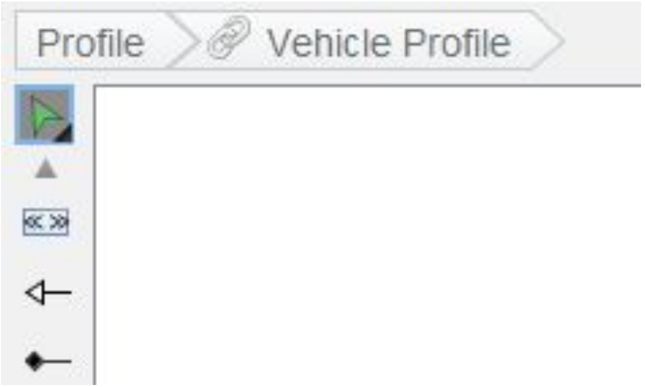
```
package TrackOrder;

public class trackOrder {

          public void msg() {
                    System.out.println("Trac
kOrder");
    }

          public static void main(String[]
args) {

          }

}
```

How to Draw profile Diagram?

Creating a profile diagram

To create a profile diagram:

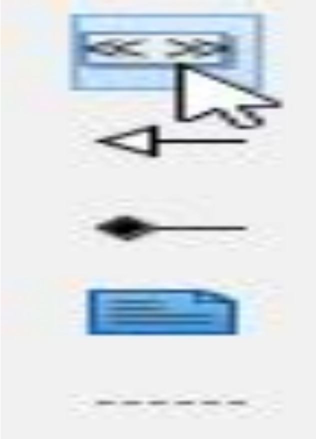
1. Select **Modeling > Profile > New Profile** from the toolbar
2. Name the diagram and press **Enter** to confirm.



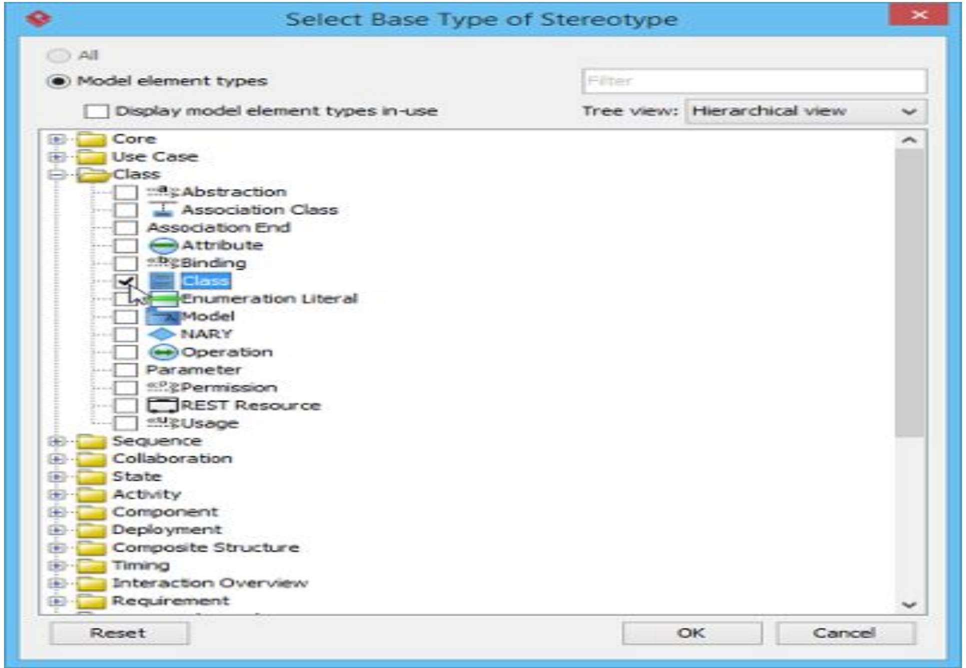
Drawing a stereotype

To draw a stereotype in profile diagram:

1. Select **Stereotype** in diagram toolbar



1. Click on the diagram to create a stereotype.
2. In the **Select Base Type of Stereotype** window,
3. select the base type of stereotype from the model type tree.
4. A base type is the type of model element that the stereotype will extend.
5. Click **OK**. Name the stereotype and press **Enter** to confirm creation.



Select Base Type of Stereotype

○ All

● Model element types

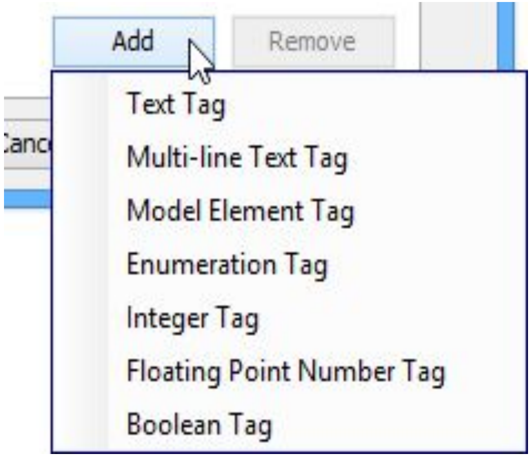
□ Display model element types in-use

Tree view: Hierarchical view

☑ Class

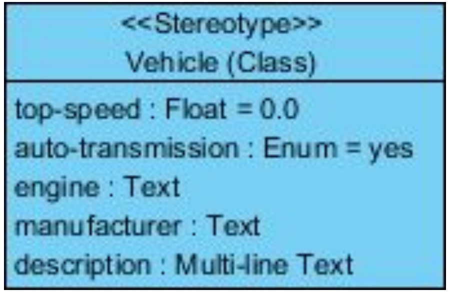
Defining tagged values for stereotypes

1. Right click on a stereotype shape and select **Open Specification...** from the popup menu.
2. In the **Stereotype Specification** window, open the **Tagged Value Definitions** tab.
3. Click **Add**. Select the type of tagged value to define. The type of tagged value limits the type of content user can enter for a tag



4. Double click the name cell and enter the name of tag. Repeat step 3 and 4 to add all tagged values for this stereotype.

5. You can assign a default value to a tag by editing the **Default Value** cell. Usually, you give a tag a default value when the value is true in most cases. For example, a tag "in-door-temperature" can have "25" as default value.

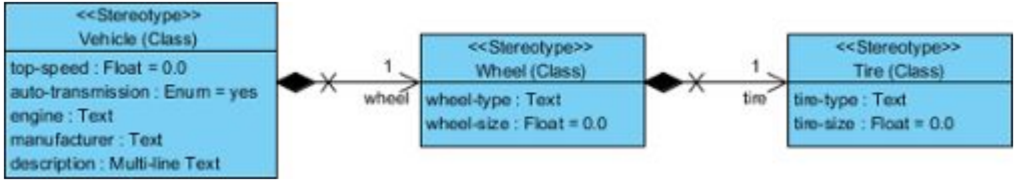


<<Stereotype>>
Vehicle (Class)

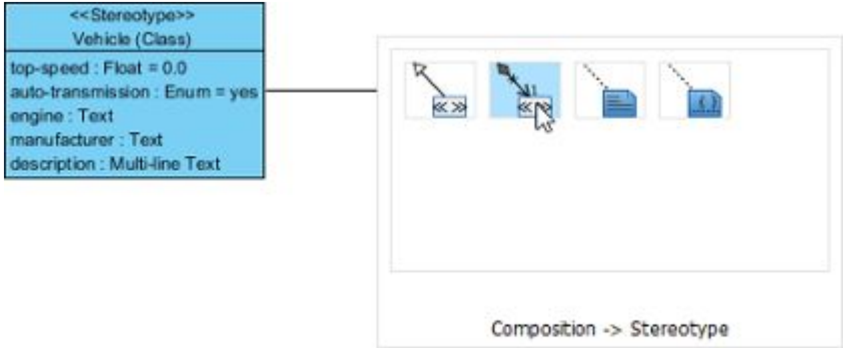
top-speed : Float = 0.0
auto-transmission : Enum = yes
engine : Text
manufacturer : Text
description : Multi-line Text

Composition

A composition relationship shows a "part of" relationship between stereotypes. The composite stereotype has responsibility for the existence and storage of the composed stereotype.

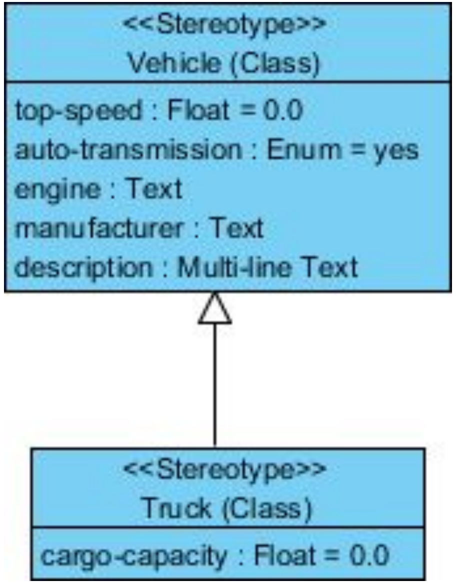


Select **Composition -> Stereotype** from Resource Catalog.

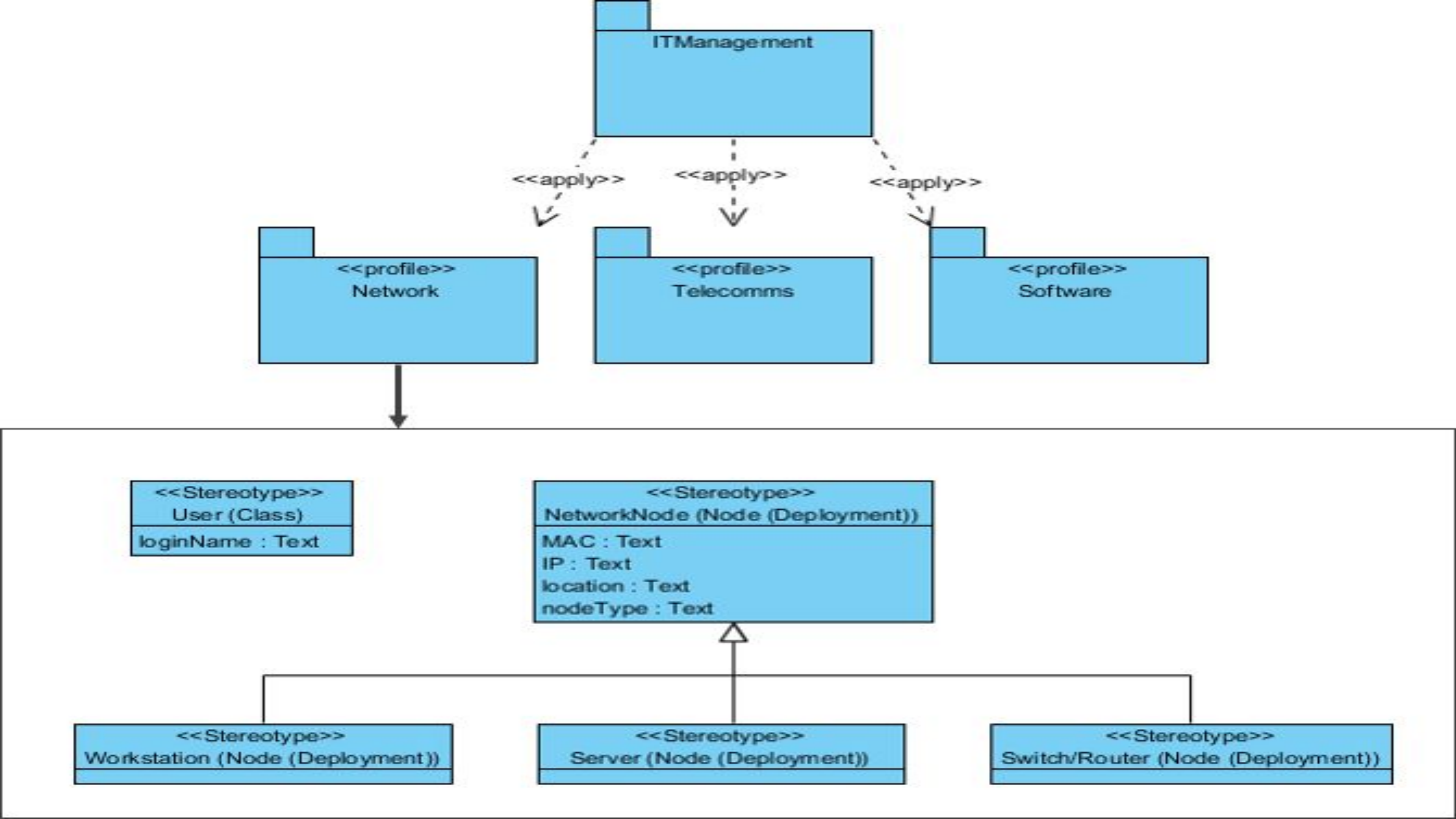

<<Stereotype>>
Vehicle (Class)
top-speed : Float = 0.0
auto-transmission : Enum = yes
engine : Text
manufacturer : Text
description : Multi-line Text
Composition -> Stereotype

Generalization

A generalization relationship shows a "kind of" relationship between stereotypes.



Example - Profile Diagram (IT Management)


ITManagement
<<apply>>
<<apply>>
<<apply>>
<<profile>>
Network
<<profile>>
Telecomms
<<profile>>
Software
<<Stereotype>>
User (Class)
loginName : Text
<<Stereotype>>
NetworkNode (Node (Deployment))
MAC : Text
IP : Text
location : Text
nodeType : Text
<<Stereotype>>
Workstation (Node (Deployment))
<<Stereotype>>
Server (Node (Deployment))
<<Stereotype>>
Switch/Router (Node (Deployment))

How to Draw Composite Structure Diagram?

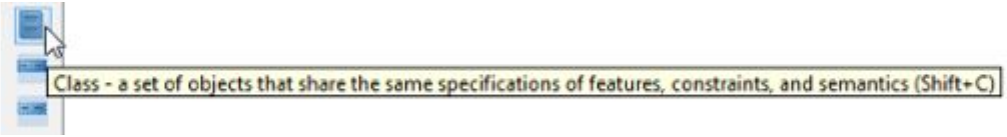
Creating composite structure diagram

Perform the following steps to create a UML composite structure diagram.

1. Select **Diagram > New** from the application toolbar.
2. In the **New Diagram** window, select **Composite Structure Diagram**.
3. Click **Next**.
4. Enter the diagram name and description. The **Location** field enables you to select a model to store the diagram.
5. Click **OK**.

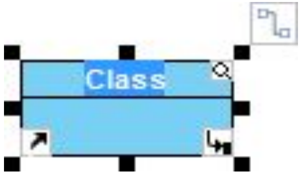
Creating class

To create a class in composite structure, click **Class** on the diagram toolbar and then click on the diagram.



Class - a set of objects that share the same specifications of features, constraints, and semantics (Shift+C)

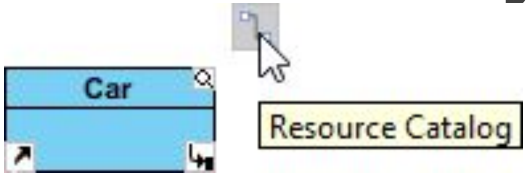
A class will be created.



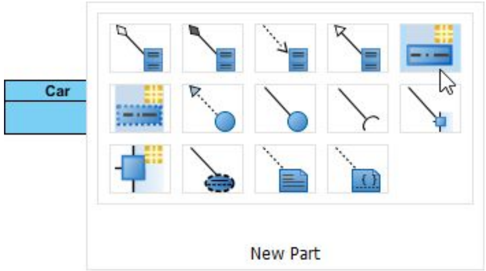
Creating part

To create a part inside a class:

1. Move your mouse pointer over the class.
2. Click on the **Resource Catalog** button.



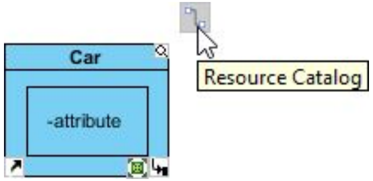
3. Select **New Part** from Resource Catalog.



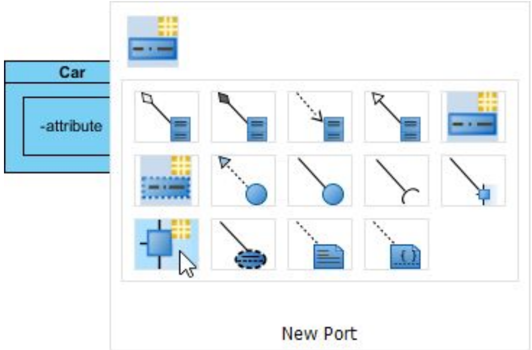
Creating port

To create a port that attaches to a class:

1. Move your mouse pointer over the class.
2. Click on the **Resource Catalog** button.

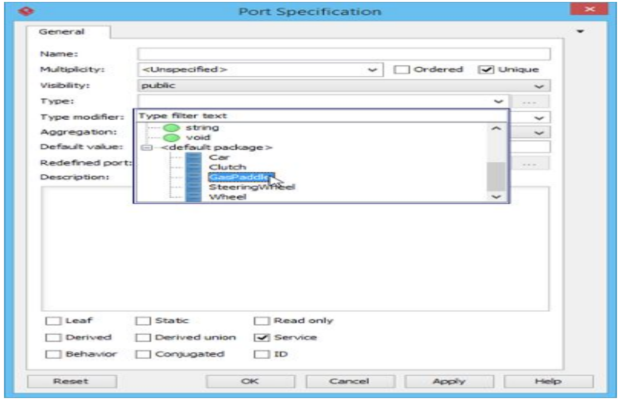


3. Select **New Port** from Resourcce Catalog.



Specifying type of port

Right-click the port and select **Open Specification...** from the pop-up menu. The **Port Specification** window appears.
Click the combo box of **Type** and select a class.



Port Specification

General

Name:

Multiplicity: <Unspecified> □ Ordered ☑ Unique

Visibility: public

Type:

Type filter text

- string
- void
- <default package>
  - Car
  - Clutch
  - GasPaddle
  - SteeringWheel
  - Wheel

Type modifier:

Aggregation:

Default value:

Redefined port:

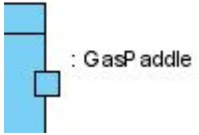
Description:

□ Leaf □ Static □ Read only

□ Derived □ Derived union ☑ Service

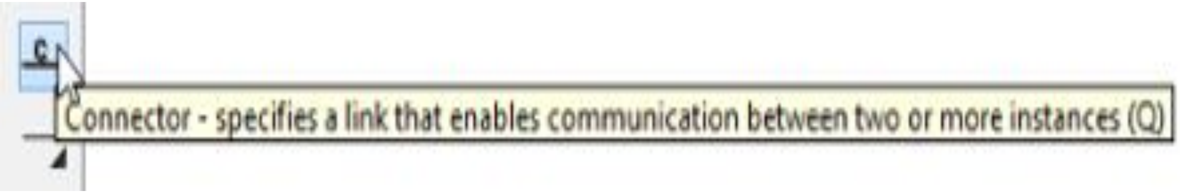
□ Behavior □ Conjugated □ ID

Click **OK** button to apply the changes. Type will be shown on the caption of the port.



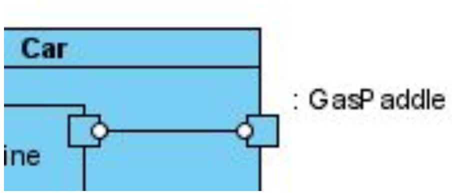
Creating connector

To create connector, click **Connector** on the diagram toolbar.



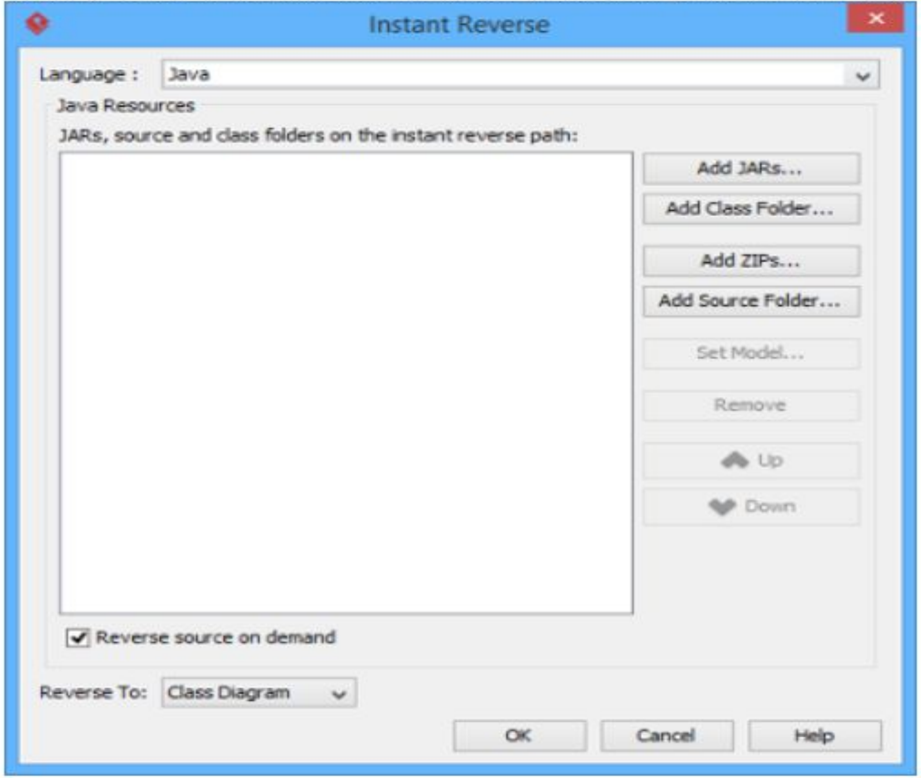
Connector - specifies a link that enables communication between two or more instances (Q)

Drag from the source shape, move the mouse over the target shape and then release the mouse button to create the connector.



Java Code to Package Diagram

1. Select **Tools > Code > Instant Reverse...** from the toolbar.
2. In the **Instant Reverse** window, select **Java** as the **Language**.
3. Specify the path of the source file or the folder that contains those files.



Language : Java

☑ Reverse source on demand

Reverse To: Class Diagram

The instant reverse window

4. Select **Package Diagram** for **Reverse To:**.
5. Click **OK** to start reversing.