VP-UML
Quick Start

Last update: 3rd May, 2011

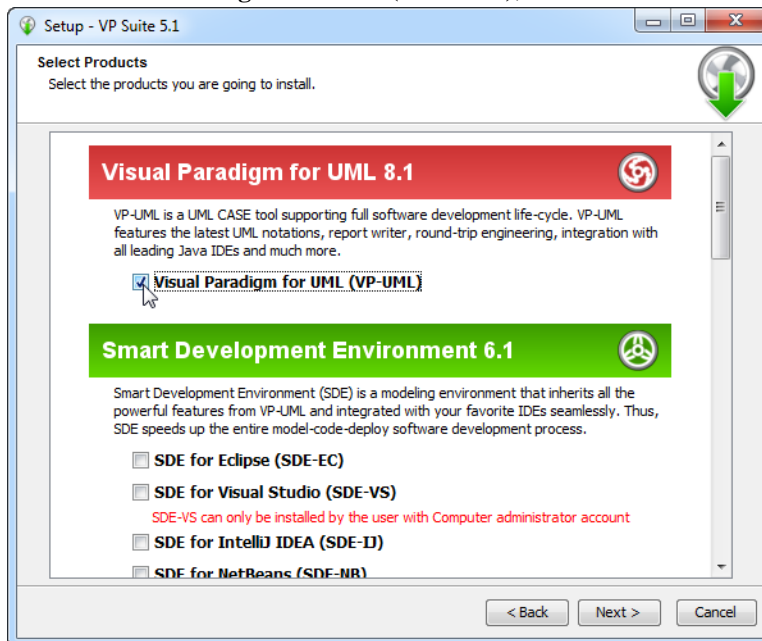# Table of Contents

Getting started

Installing Visual Paradigm for UML (VP-UML)

After you have downloaded Visual Paradigm for UML from our website, the next step you should do is to install it. The following steps are going to teach you how to install it:

1. Double click the downloaded installer file to execute it, then click **Next** button to go to next page.

   VP_Suite_Windows_5_1_20101225....
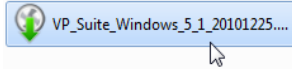
2. Go through the **License Agreement**. Choose **I accept the agreement** after you finish reading the agreement and fully understand and accept the terms, and then click **Next**.

3. Select the destination you want VP Suite to be installed, and then click **Next**.

4. Select the location you want the Start Menu folder to be placed, which is the location where the program's shortcut is, under the Start menu. Click **Next**.

5. Select the file associations that you want to create.

6. Check **Visual Paradigm for UML (VP-UML)**, and then click **Next**.

7. Fill out the **Products Configuration**. Select the edition you have purchased or you want to evaluate, then click **Next**.

8. Fill out the **Products License**. It is optional that you can fill in or just ignore it, then click **Next**.

| Note	If you put the key and installer file in the same folder, it'll automatically fill the key address for you in file path. Wait a few seconds for installing and checking system environment.

9. The installation has been completed. You can choose either to start it now by clicking **Visual Paradigm for UML 8.1** and then click **Finish** or start it later on by clicking **Don't start**.

Starting Visual Paradigm for UML

You can start Visual Paradigm for UML by selecting **Start Menu** > **Visual Paradigm > Visual Paradigm for UML 8.1 [Enterprise] Edition** (the edition you choose during installation).

Importing license key

1. After you enter VP-UML, you will be asked to provide license key in **License Key Manager**.
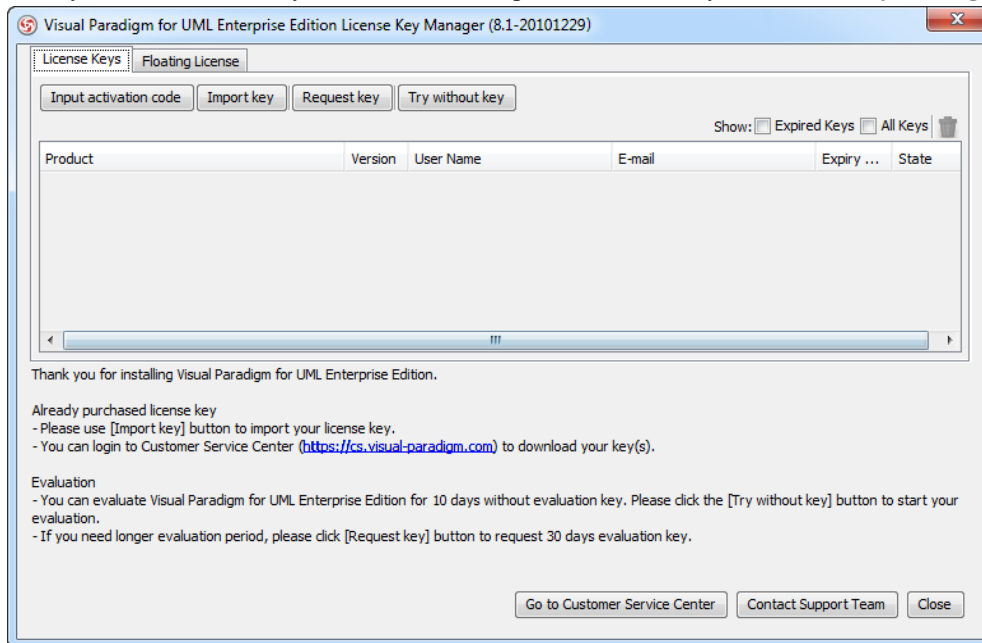
a) If you have purchased VP-UML, you should receive our notification (Email) with an activation code included. The same activation code can also be found from the license key listed in your customer account. Copy the activation code.
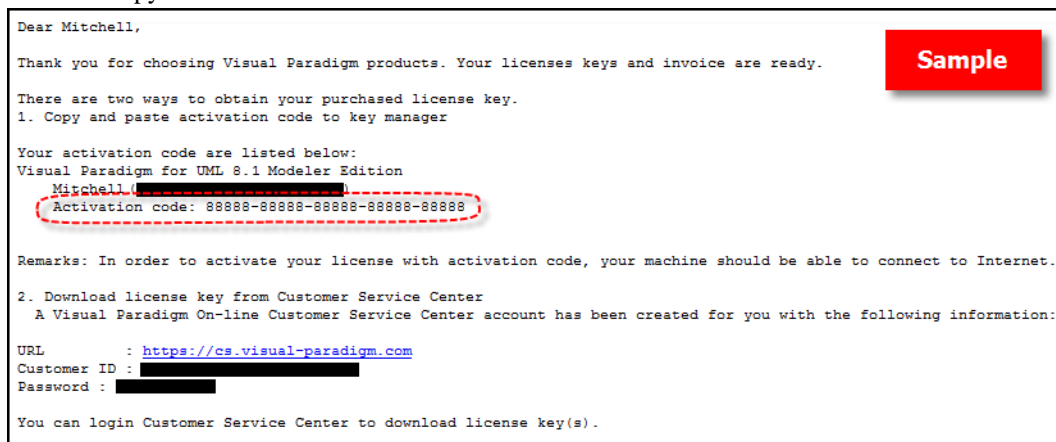
Click on **Input activation code** and paste the copied activation code in the dialog box.
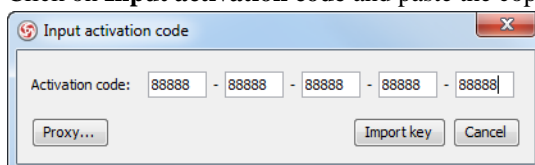
b) For evaluation, please click either **Try without key** or **Request key**. Choosing **Try without key** provides 10 days evaluation period without input your email while choosing **Request key** lets you request for an evaluation key that allows you to evaluate 30 days. If you are our member, you will be asked to sign in after you click **Request key**. If you are not our member, you should register as a member on Internet by filling the Registration form. The key will be sent to your email account automatically after you have signed in.

Note	If you are running the Community Edition, you can either try VP-UML for 1 hour without key, or request for a key that enables you to run permanently.

Selecting workspace

Workspace is a directory that can memorize your own setting and preferences. You will be asked to select a workspace every time you start VP-UML. If you would like to keep the application settings, always start with the same workspace. When you move to a new computer, you just need to copy the workspace, and then all settings will be kept. If you want to have a fresh working environment, start with a new workspace.
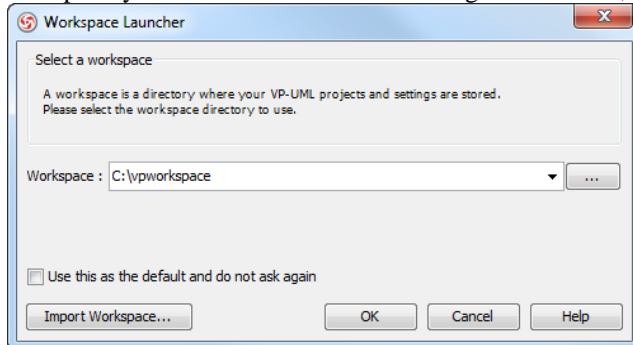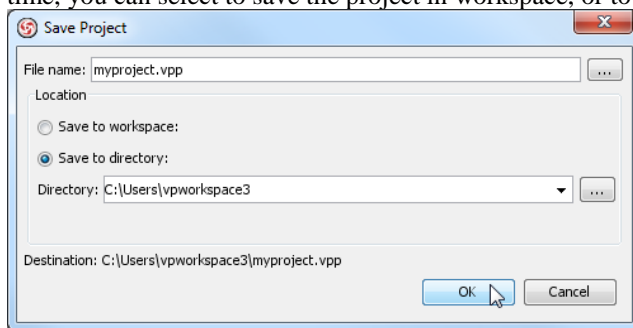

Environment
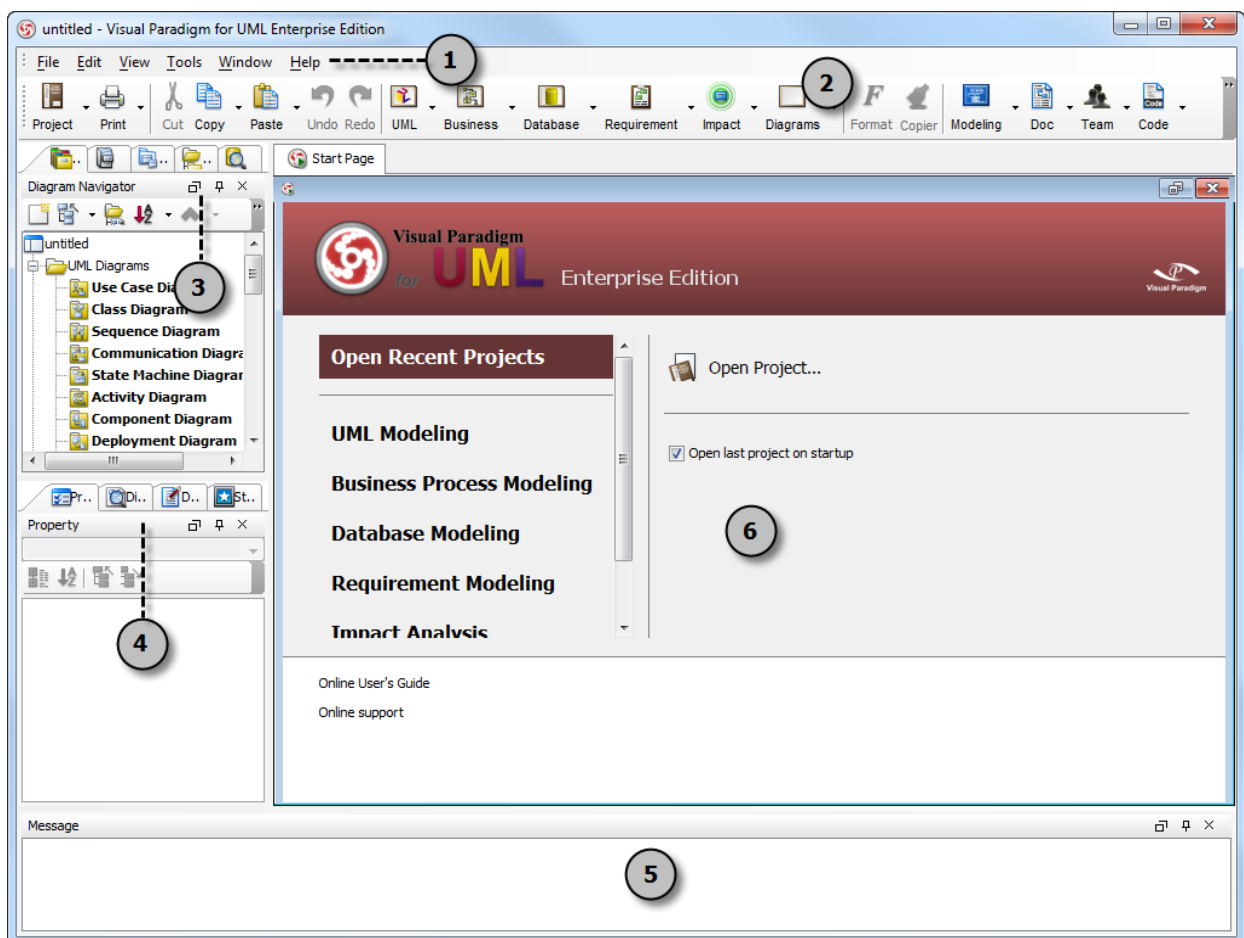
Menu bar, Toolbar, Diagram Navigator, Property pane, Message Pane and Diagram Pane are shown on the environment of VP-UML window. The brief introduction of each function shows in the following table:

1	**Menu bar**	The menu bar at the top of the window allows you to select and perform various operations in Visual Paradigm for UML.
2	**Toolbar**	Toolbar, which is below the menu bar, is the extension of menu. All buttons are presented as groups of icons that handily placed for users.
3	**Diagram Navigator**	A place where diagrams are listed, and where you can create and access diagrams base on their types.
4	**Properties Pane**	The properties of chosen model/ shapes will be shown on properties pane upon selection.
5	**Message Pane**	Possible information or warnings will be shown here.
6	**Diagram Pane**	The diagram will be displayed in diagram pane.

Saving and opening projects

To save your work, select either **File > Save Project** or **File Save Project as…**. When you perform saving the first time, you can select to save the project in workspace, or to another directory you preferred.

To open an existing project, select **File** > **Open Project…** from the main menu and select the project to open.

Diagramming

After you are familiar with the environment of Visual Paradigm, you should get to know how to create a diagram. This chapter is going to teach you not only how to create diagrams, but also how to create and connect diagram elements (shapes), documents their details, how to format them and a general description on the nickname and layers feature.
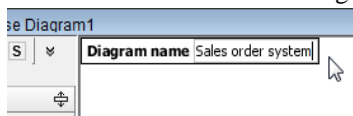
Creating diagram

To create a diagram say, a use case diagram:

1. Right click **Use Case Diagram** on **Diagram Navigator** and select **New Use Case Diagram** from the pop-up menu.
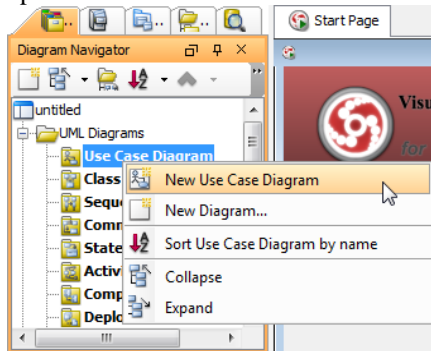
2. Enter the name for the new diagram as *Sales order system* after creating it.

Creating shapes

To create a shape, select the type of shape from the diagram toolbar, click on diagram and start dragging it to create with a preferred size. Taking creating use case as an example:

1. Select **Use Case** from the diagram toolbar, click on the diagram and then start dragging. Release the mouse to confirm creation.
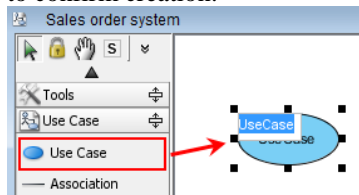
2. Enter the name for the use case as *place order* and press enter to confirm the name.

Connecting shapes

Two shapes can be connected by making use of the resource icons surrounding a shape. Let say if we want to associate an actor with a use case, move the mouse pointer towards the actor, press on its **Association** resource icon and drag it to the use case, finally release the mouse button.
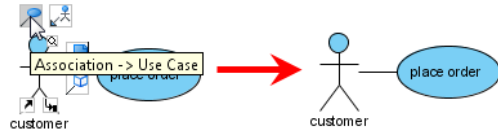
Documenting model element

You can type in the textual description to your shape by opening the **Documentation Pane** at the bottom left of screen and typing in the space provided.
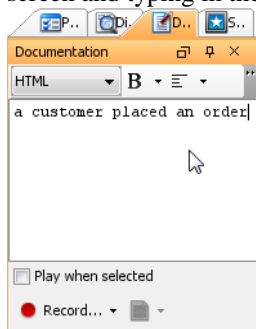
In addition to text description, you can describe by voice through recording. Click the **Record** button at the bottom of **Documentation Pane**. In the **Record Voice** dialog box, start recording by clicking the red circle button and stop recording by clicking **Stop** button. To save your recording, click **OK** button.
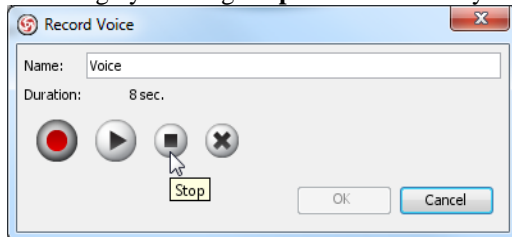
| Note	Make sure your recording device is available in order to apply this feature.

Formatting shapes

You can format a shape with your preference by right clicking on it, selecting **Styles and Formatting** and then **Formats…** from the pop-up menu. Taking changing the use case's background color as an example:

1. Right click use case and select **Styles and Formatting** and then **Formats…** from the pop-up menu.
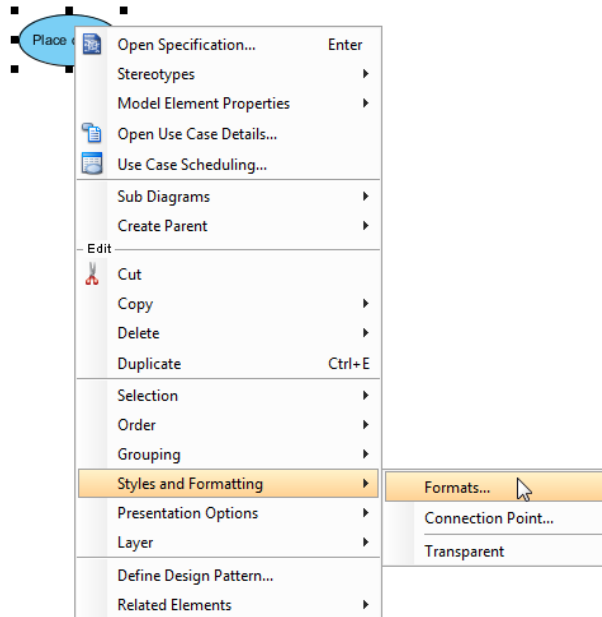
2. In the **Formats** dialog box, select a color such as *green* in the **Background** tab and click **OK** button to confirm the change, the color of use case will turn into green.
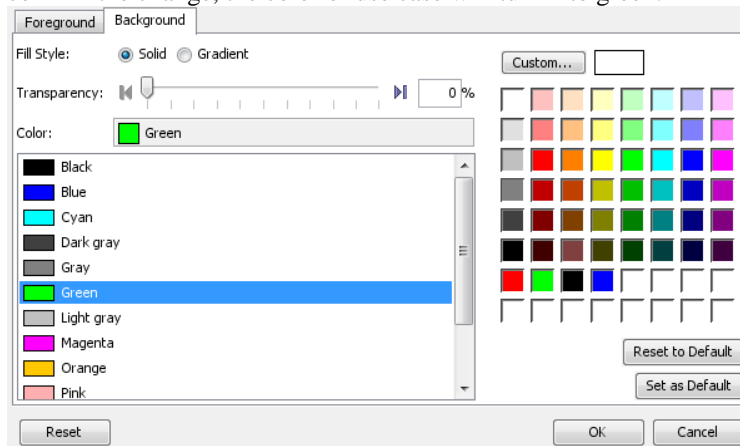
Formatting the connector style as another example:

1. Right click a connector and select **Styles and Formatting** and then **Formats…** from the pop-up menu.

2. In the **Formats** dialog box, select *blue* as line color and click **OK** to confirm. As a side note, the option **Weight** is used to alter the thickness of the connector. The higher the value in **Weight**, the thicker the connector will become.
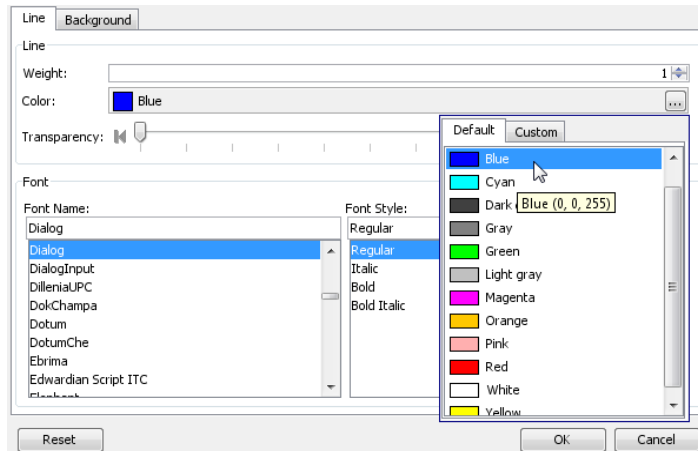

Using Nicknamer

Nicknamer is a feature which helps you to set up and manage multiple language sets of a model. This can apply on those multinational corporations for modifying various versions of a project to local staffs. The original nickname is defaulted as English. You can add a new nickname, for example, Traditional Chinese:

1. Select **View > Nicknames > Configure Nicknames…** from the main menu.

2. In the **Configure Nickname** dialog box, click **Add User Language** and select **Chinese (Traditional)** in the **Add User Language** dialog box and then click **OK**. Finally, click **OK** to go back to the diagram and start working with the Chinese version of model.
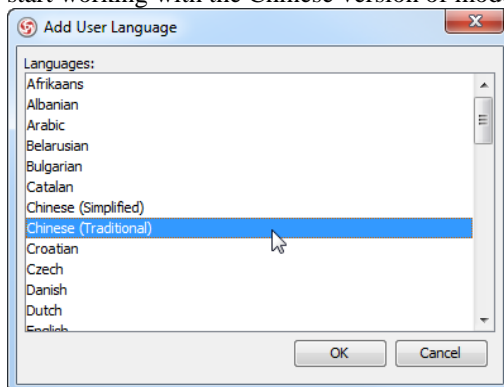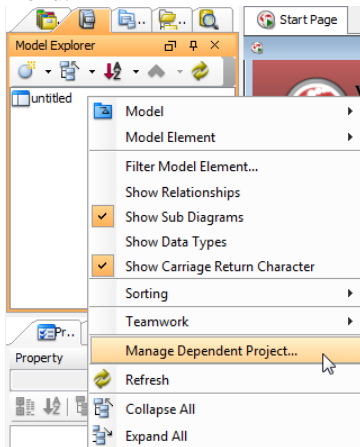

| Note	To change the nickname back to English, select **View > Nicknames > Original** from the main menu.

Add dependent project

Project dependence is a feature enables us to reference model elements in external project, and use its model element(s) in our own project.

1. Right click on **Model Explorer**'s background and select **Manage Dependent Project…** from the pop-up menu.


2. When the **Manage Dependent Projects** dialog box pops out, click **Add** button to select the reference project(s) from your computer.
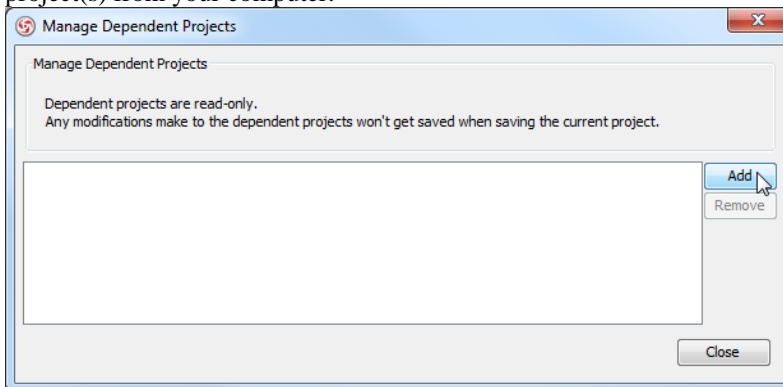

3. After select several projects, press **Close** button in **Manage Dependent Projects** dialog box.

4. The reference project(s) will be shown on **Model Explorer** in combo box. You can switch between your working project and the referenced project through this combo box.
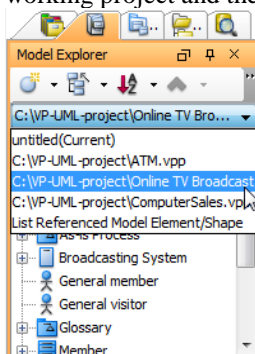
5. You can drag your preferred model from the referenced project on your working diagram and connect it with your current project. For example, drag *Administrator* class on the diagram.
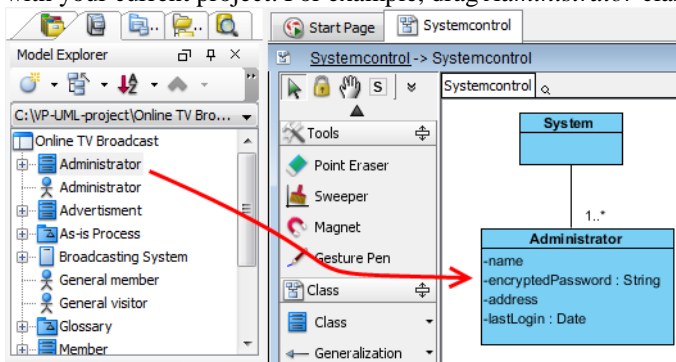
6. The list of referenced models can be found by selecting **List Referenced Model Element/Shape** in the combo box.
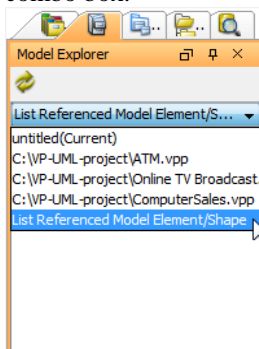
7. In the Diagram Navigator, scroll to the bottom, expand the dependent project tree node and double click on a diagram to open it. You can read dependent project's design with this method.
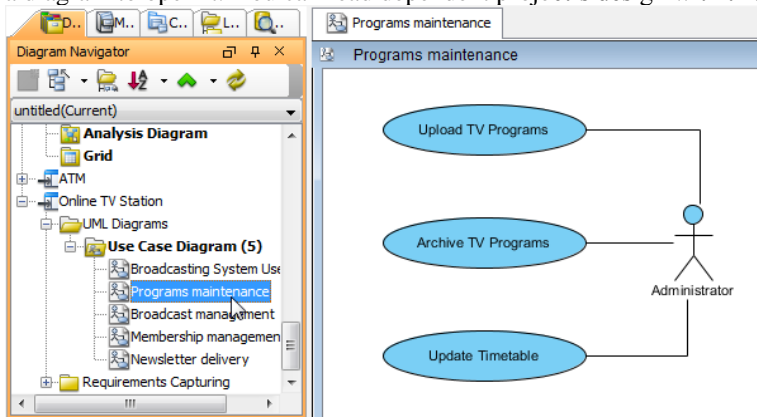

The benefit of using referenced model is to prevent your working project from being oversized because the information of referenced model will not be stored. However, since the referenced model is read-only from its source project, you cannot create a child on it. To deal with this problem, you can make use of **Create Mirror Model Element**. The mirrored model element is also read-only on its properties, but comparing with models in reference project, you can add child model to it.

Traceability

When you create a new term in glossary grid, you can create a model element simultaneously. Moreover, you can trace the created use case from glossary grid. Let's create *Place order* use case in glossary grid and paste it in *Sales order system*.

1. Click **Requirement** on the toolbar and select **Glossary Grid** to create a glossary grid.

2. In **Glossary Grid** page, click **Create Term (Insert)** button to create a new term and name it as *Place order*.
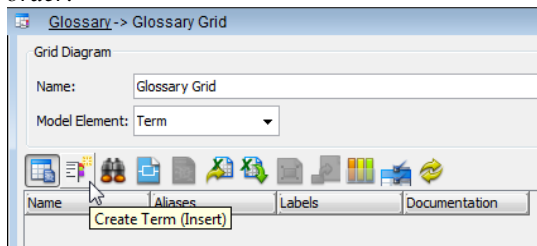

3. Right click on the term and select **Create Use Case** from the pop-up menu.
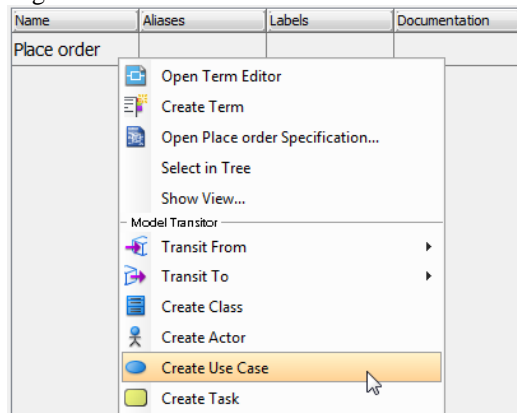

4. When the **Use Case Specification** dialog box pops out, you may define the rank document for the use case. Click **OK** button to proceed.
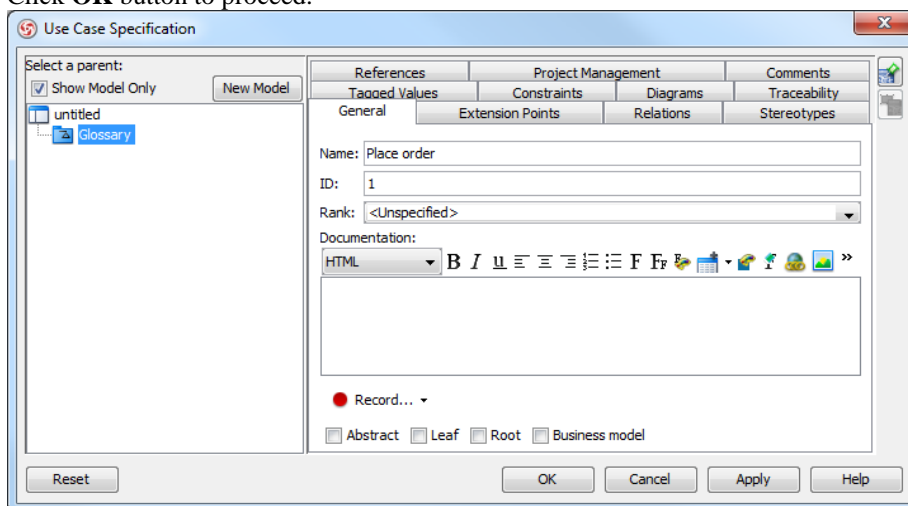
5. Subsequently, the **Visualize Model Element** dialog box pops out. Check either **Create new diagram** if you want to create the use case in a new diagram or **Show in existing diagram** if you want to create the use case in an existing diagram. Click **Show** button to confirm and proceed.
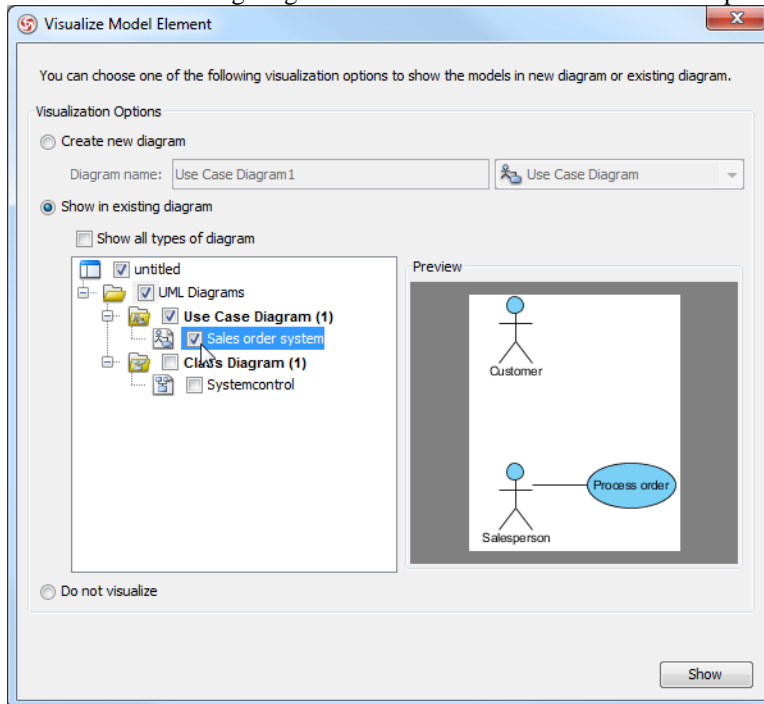
6. As a result, *Place order* use case is shown on the selected diagram.
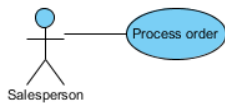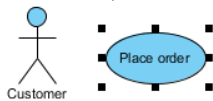
UML modeling

Drawing use case diagram

Use case diagram is designed for modeling the context of a system. In a use case diagram, all stakeholders and system goals are identified to elaborate how a system is formed. The main elements of use case diagram include use case, actor and association (communication link).

The following simple use case diagram illustrates the basic requirements of enrolling in an overseas university. The actor *Student* represents the role of student while *apply for student visa online*, *take a physical examination* and *pay tuition fee* are three goals that the student can achieve. You can create actor *Student* from diagram toolbar, and the three use cases from the resource icon **Association -> Use Case** of actor.
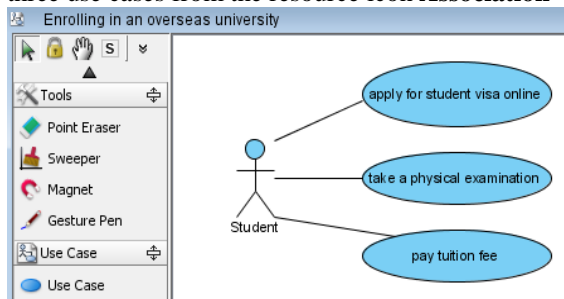

Documenting use case details

Apart from naming a use case, you also can document its detail. Right click on a use case, such as *Book Sport Facility* and select **Open Use Case Details…**. Open the **Info** tab and fill in the **Rank**, which presents the importance of this use case.
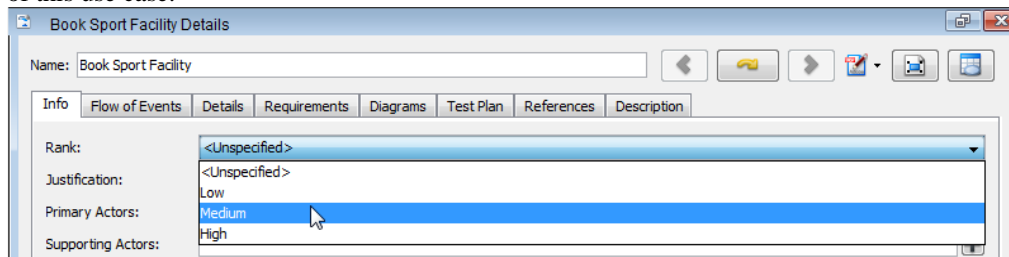

Furthermore, you can provide the procedure of attaining the goal of the use case in **Flow of Events**. Open the **Flow of Events** tab and enter the procedure step by step. Below are the procedures of booking sport facility online. Firstly, *Login*; secondly, *Select a sport facility* and finally, *Click [Confirm]*. The first step is followed by three sub-steps: *Fill in name*, *Fill in password* and *Click [OK]* respectively. Press **Tab** to add a sub-step after a step. Enter them one by one in the flow of events editor as the image shown below.
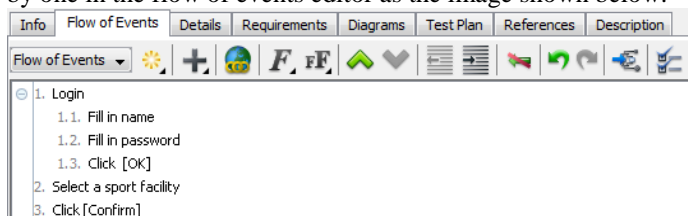
You can insert an extension for a step to perform different situation. For example, you can insert an extension for *Login*. As the login system is only for members, those non-memebers are redirected to another page instead of login page. To insert an extension for *Login*, click **Login** and then select the **Extension** menu (or press **Shift-Enter**). After that, start editing the extension.
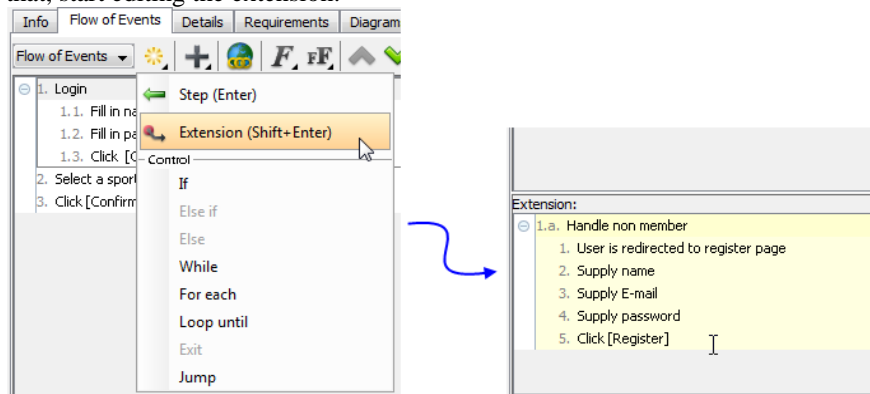
Building glossary

Glossary is a place where domain-specific vocabularies are stored and managed. Instead of adding terms (vocabularies) from your imagination, you can develop a glossary by identifying terms from flow of events.

1. Suppose *online system homepage* in a step of flow of events is a key phrase. Select it in flow of events, right click and select **Add "online system homepage" to Glossary** from the pop-up menu to extract this as a term.
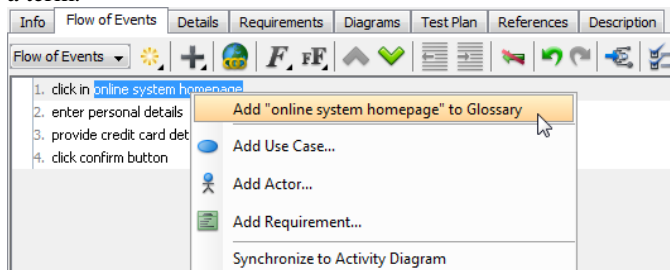
2. This opens the glossary with term *online system homepage* added. Right click on the term and select **Open Term Editor** from the pop-up menu.
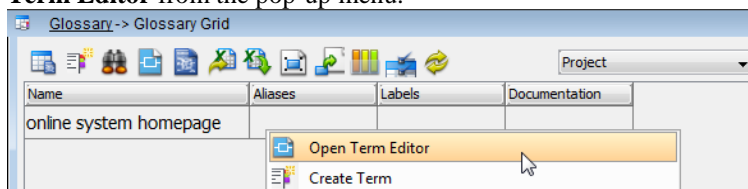
3. Try to specify its alias. In the **Term Editor**, click **Add** and enter *official site* as the first alias. Repeat to add alias *main page*. Enter definition of the term in **Definition**.

| Note	Move your mouse pointer to the underlined term *online system homepage* in **Flow of Events**, the documentation of the term will reveal automatically.

Drawing sequence diagram

A sequence diagram is used primarily to show the interactions between objects that are represented as lifelines in a sequential order. More importantly, lifelines show all their points of interaction with other objects in events. A sequence diagram can be created by right clicking **Sequence Diagram** on **Diagram Navigator** and then selecting **New Sequence Diagram** from pop-up menu.

Taking the use case of *apply for student visa online* as an example:

1. Select **Actor** from the diagram toolbar to create an actor. Name it as *Student*.
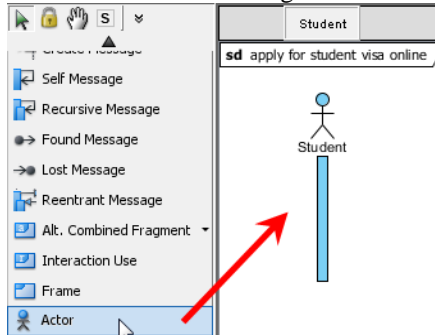
2. Press the actor's resource icon **Message -> LifeLine** and drag it to your desired place to create a lifeline with interaction to actor *Student*. Name *online system* as the lifeline, and *enter personal details* as the message between them.
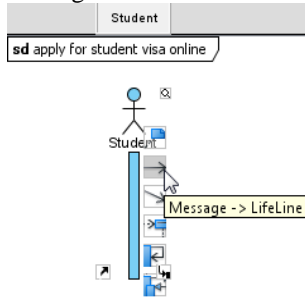
3. Repeat the previous steps to create a lifeline *database* from *online system*.
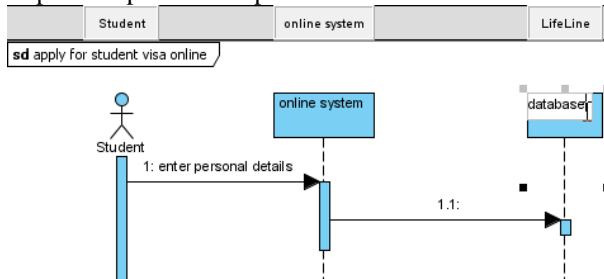
Drawing activity diagram

An activity diagram is essentially a flowchart, showing flow of control from one activity to another. Unlike a traditional flowchart, it can model the dynamic aspects of a system because it involves modeling the sequential steps in a computational process. A new activity diagram can be created by right clicking **Activity Diagram** on **Diagram Navigator** and then selecting **New Activity Diagram**.

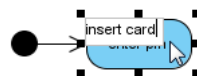Taking *using an ATM* as an example:

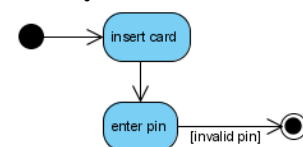1. Select **Initial Node** on diagram toolbar and drag it on the diagram pane.
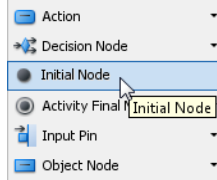
2. Press initial node's resource icon **Control Flow -> Action** and drag it to the target position to create an action.

3. Name the action as *insert card*.

4. Terminate the activity by creating an activity final node through the resource icon **Control Flow -> Activity Final Node** of the final action.

5. You can use swimlane to group actions based on the participants involved. Select **Horizontal Swimlane** from the diagram and drag it on the diagram pane to create it. Double click the head of the partitions and name them as *user* and *ATM system* respectively. You can then move shapes into the appropriate partition.
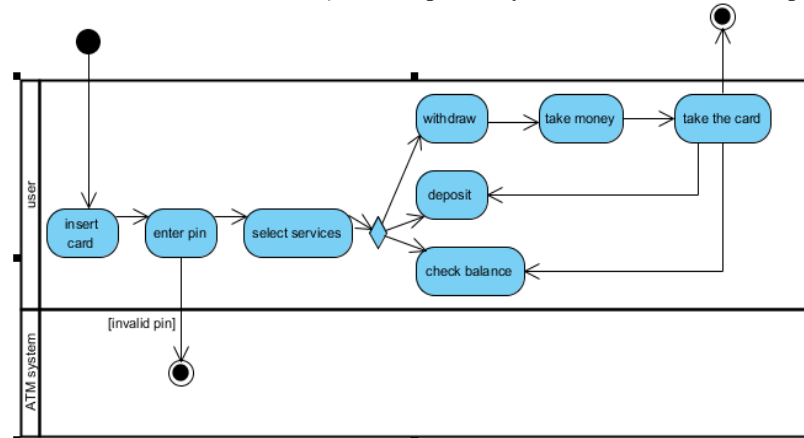
Drawing class diagram

A class diagram shows the blueprints of objects required by a system and the relationships between them. A new class diagram can be created by right clicking on **Class Diagram** on **Diagram Navigator** and then selecting **New Class Diagram** from the pop-up menu.

Taking *bank account* as an example:

1. Select **Class** on the diagram toolbar and drag it on the diagram pane.
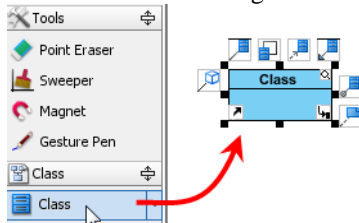

2. Name the class as *bank account*. To create an attribute, right click the class and select **Add > Attribute** from the pop-up menu. You can create as many as attributes as you need by pressing **enter** after adding a new one. In the class of bank account, create *name* and *balance* as two attributes respectively.
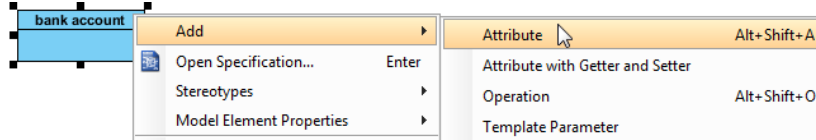

3. Similarly, operation can be created by right clicking the class and select **Add > Operation** from the pop-up menu. Create *debit* and *credit* as two operations respectively.
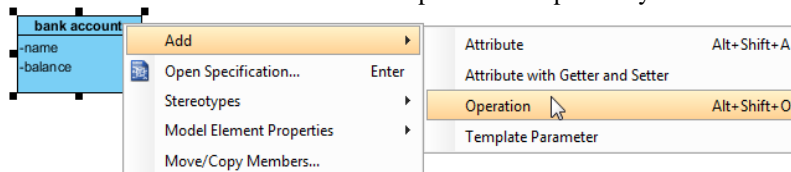
4. Generalization is needed when you want to show the subclass out of the super class. Move the mouse over the super class, select its resource icon **Generalization -> Class** and then drag it to your preferred place.
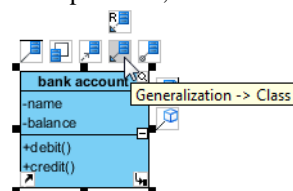

5. Create two subclasses and name them as *saving account* and *term deposit account*. Add *InterestRate* and *depositMonthlyInterest* as attribute and operation of class *saving account*, and add *PeriodInterestRage* and *initialDeposit* as attribute and operation of *term deposit account*. The completed class diagram is shown as below:
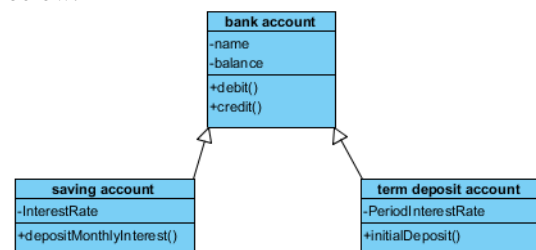
A class can be associated through the use of resource centric interface. Move the mouse over the class *student*. Press its resource icon **Association -> Class** and drag to another class *school*.
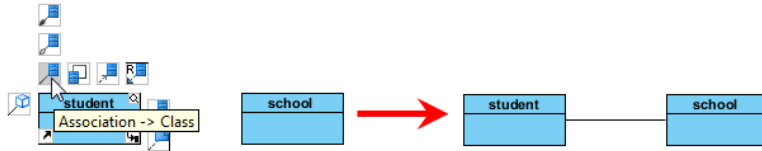
To edit an association or its ends, double click on it or either of its ends to open the **Association Editor**. Name the role of association end by typing it in the middle text box and adjust the properties like multiplicity and navigability.
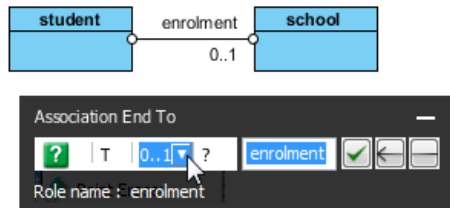
Enterprise Architecture

There are tools that help you define en enterprise by modeling the vision, goals and missions of business activities.

Zachman Framework [Enterprise Edition]

Zachman Framework is matrix which consists of six fundamental questions (What, how, where, who, when, why) as columns with six different perspectives of a business as rows. It lets you define and view an enterprise from a formal and structured way. To create a Zachman Framework:
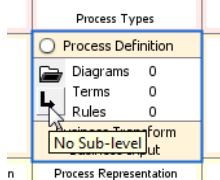
1. Select **Enterprise** on the toolbar and select **Zachman Framework** from the drop down menu.
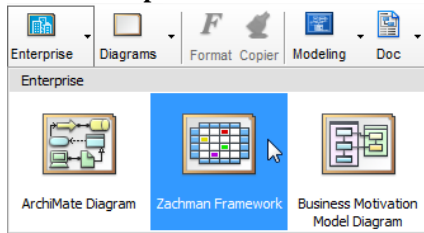

2. Click on a cell to edit it.
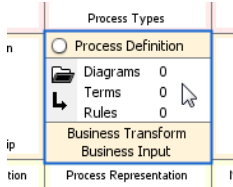
3. You can add diagrams, terms and define business rules for this cell by clicking on the corresponding text link. You can also add a sub-level of Zachman Framework. Let's try. Click on the arrow button.

4. In the **Sub-Level** dialog box, select New Zachman framework. Click **OK** to continue.
5. In the sub-level Zachman framework, you may click the navigation link to go back to the parent level.
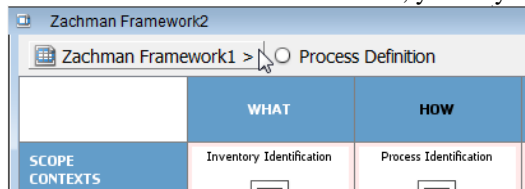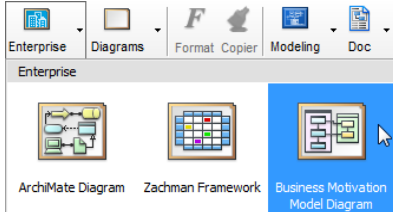
Business Motivation Model (BMM)

Identify, capture and design organization motivation, vision, mission and strategy visually. To create a BMM diagram:

1. Select **Enterprise** on the toolbar and select **Business Motivation Model Diagram** from the drop down menu.
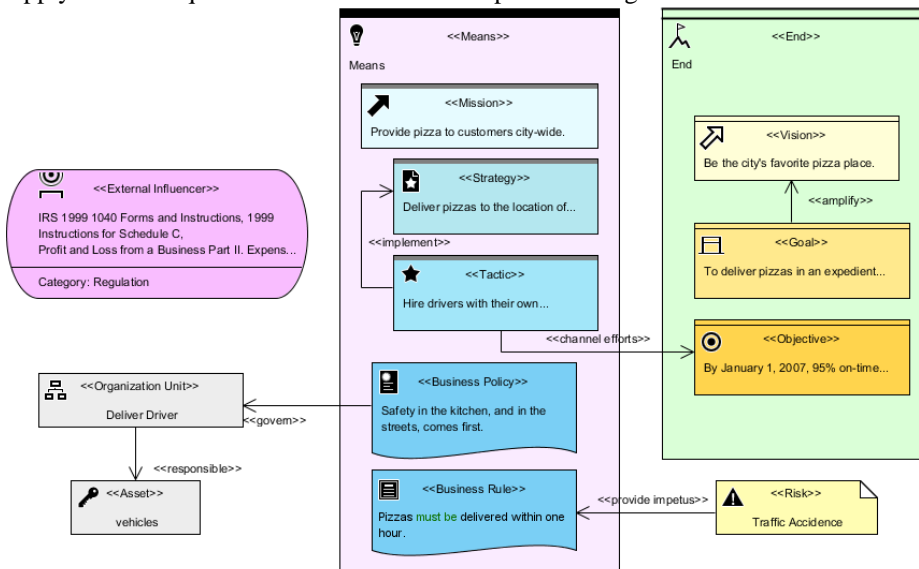

2. Select **Means** from diagram toolbar.
3. Press and drag on the diagram to form a rectangle. Release the mouse button and enter *Means* as name.

4. Select **Mission** from diagram toolbar. Click inside the *Mission* shape to create a mission. Enter text: *Provide pizza to customers city-wide*.
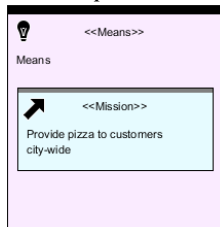
5. Apply the techniques described before to complete the diagram. Make it look like this:

Code generation

Code generation is the process by which code generator generates source code from UML class model. By editing the generated code by filling in the code logic, a fully completed application can thereby be set up.
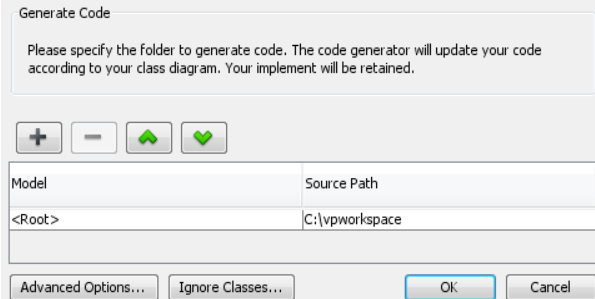
Java round-trip

Round-trip engineering enables you to keep class model and source code in-sync. With Java round-trip, you can reverse a code-base to VP-UML as class model, analyze, and make changes such as to add missing classes, and then update the changes to code, or vice versa.

- To generate Java source code from class model, select **Tools** > **Java Round-trip > Generate Code…** from the main menu. Enter the output path in the **Generate code** dialog box and click **OK** to generate.

- To reverse class model from code, select **Tools** > **Java Round-trip > Reverse Code…** from the main menu. The **Reverse code** dialog box will be pop up and ask you to select the source file path. Click **OK** to reverse.
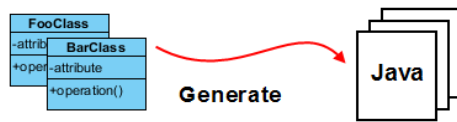
C++ round-trip

- To generate C++ for the whole project, click **Tools** from the menu bar, and select **C++ Round-trip >** **Generate Code…**. **Generate code** dialog box will be pop up and ask you to select a path, click **+** button to add a path. After selecting, click **OK** to generate.
- To reverse class model from code, select **Tools** > **C++ Round-trip > Reverse Code…** from the main menu. Select source file path in the **Reverse code** dialog box and click **OK** to reverse.

Instant generator [Standard Edition or above]

Instant generator produces source code from your model at a particular instant. Unlike the code generation support in round-trip engineering, instant generator is a one-off. To generate code, select **Tools** > **Instant Generator** from the main menu, then select the programming language to generate.

Instant reverse [Standard Edition or above]

Instant Reverse allows you to reverse different types of source into UML class models, such as Java source, Java classes, C++ source etc. To reverse, select **Tools** > **Instant Reverse** from the main menu, then select the appropriate programming language. Select the source files and proceed.

Instant reverse Java code to sequence diagram [Professional Edition or above]

Sequence diagram can help to represent the interaction between objects in runtime. VP- UML enables you to reverse your Java source code into sequence diagram. You can gain a better understanding of Java source code by reading diagram instead of looking to a possibly thousand lines of source code. To reverse Java code to sequence diagram, select **Tools > Instant Reverse > Java to Sequence Diagram…** from the main menu. Add the folder that contain the source code, continue and select the source file. Finally, visualize the code in a new diagram.
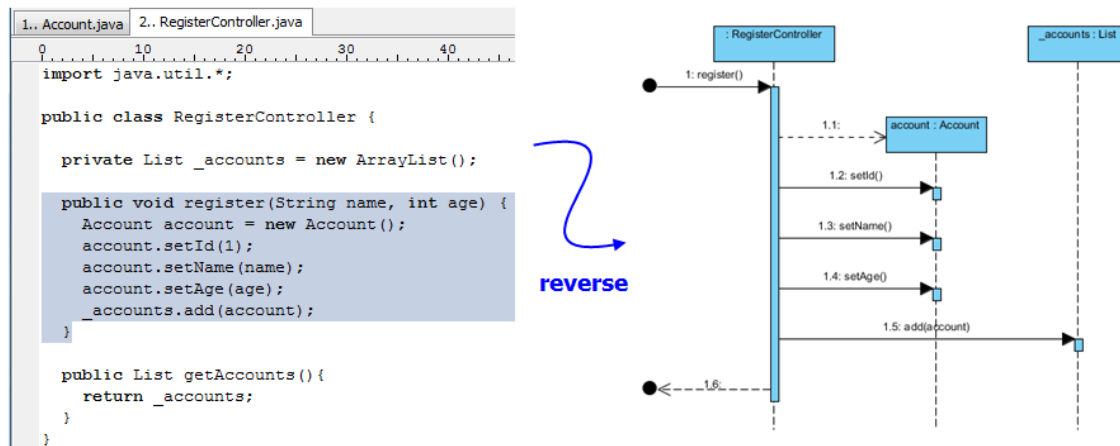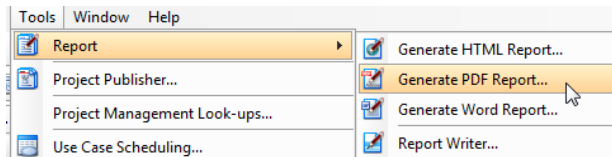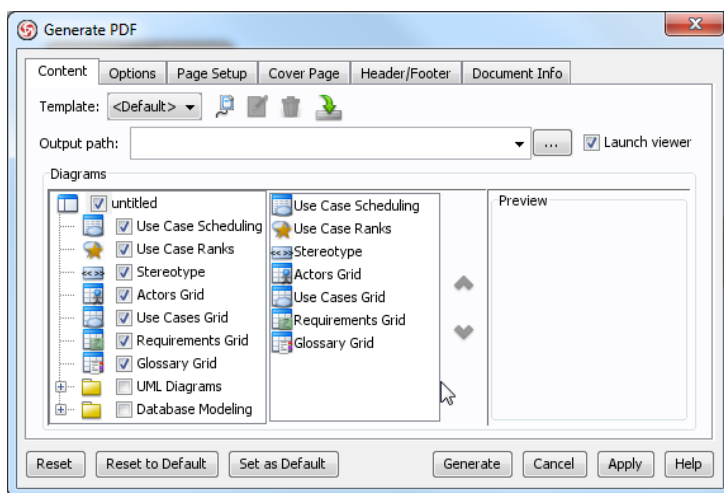
Report Generation

Generating report

You can generate a report in form of HTML, PDF or Word from your project by clicking **Tools** and selecting **Generate HTML/PDF/Word Report…**. For example, select **Generate PDF Report**… if you would like to generate a PDF report.

When the **Generate PDF** dialog box pops out, select the diagram(s) to be included in report. Fill in **Output Path and** click **Generate** to proceed with generation.
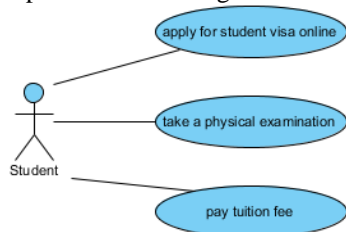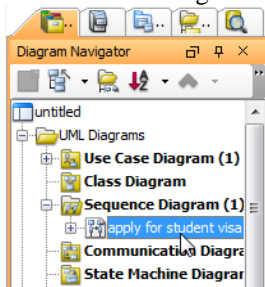

Prepare report by report composer

You can develop professional design documentation using report composer. Further to the diagram based report generation function, you can customize a report by adding elements to the report.
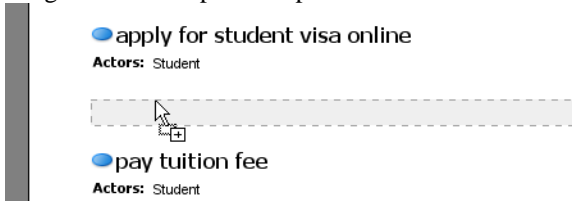
1. Open a use case diagram. Create one if not exist.

2. Right click on the background of diagram and select **Utilities > Generate Use Case Report** from the popup menu. This produces a new report diagram.
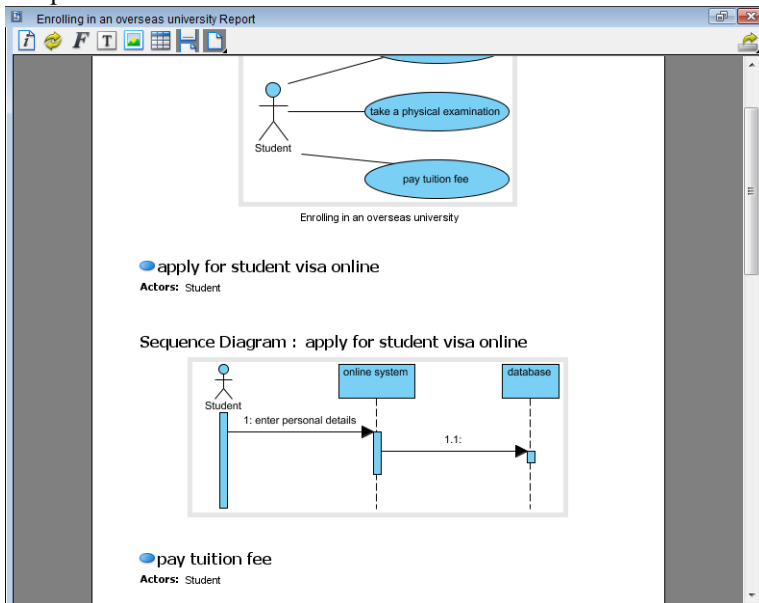
3. Select another diagram in **Diagram Navigator**.
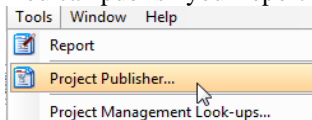

4. Drag it onto the report composer.

The result should look like this. You can click on the **Export** button at the top right to export the report composer to a HTML/PDF/Word file.

Project publisher   [Standard Edition or above]

1. You can publish your report to the website by selecting **Tools** > **Project Publisher…** from the main menu.

2. Enter the output path in the **Project Publisher** dialog box and click **OK** to publish.

3. It publishes the project, and opens the published content in web browser. You should see four tabs: Diagram Navigator, Model Explorer, Class Repository and Logical View on the left and your diagram on

the right. Click the diagram on the left, it will be shown on the right screen.

4. You can click on a shape on an image to enter its detail page, for reading its properties.

Actor - Student

Properties

Name	Value
Visibility	public
Abstract	false
Leaf	false
Root	false
Author	
Create Date Time	Jan 12, 2010 3:40:43 PM
Business Model	false
Actor ID	1

Relationships Summary

Name	Begin	End
— : Association	Student : Actor	take physical examination : Use Case
— : Association	Student : Actor	apply for student visa : Use Case
— : Association	Student : Actor	pay tuition fee : Use Case

Relationships Detail

Name	Value	
	Name	Value
	Role	
From	Element	Student : Actor
	Multiplicity	Unspecified
	Navigable	true

Teamwork Server [Modeler Edition or above]

Teamwork collaboration

SVN/CVS/Perforce [Standard Edition or above]

Your team can work together on the same project effectively through the teamwork collaboration feature. VP-UML supports the integration with version control systems such as SVN, CVS, Perforce and Teamwork Server. In this section, we will cover basic setup of Teamwork Server, where SVN, CVS and Perforce users can skip, and basic teamwork operations with Teamwork Server. The techniques to be taught can be applied to the integration with SVN, CVS and Perforce, too.

Login to server

You can start working by logging into the **Teamwork Client** in VP-UML, manage, checkout and open the project. Manage project is to tell the client products that you are involved in a particular project.

1. In VP-UML, select **Tools > Teamwork > Open Teamwork Client...** from the main menu.

2. In the **Login** dialog box, enter *127.0.0.1* as server host, *john* as name and password (follow that defined in the last section). Click **OK**.
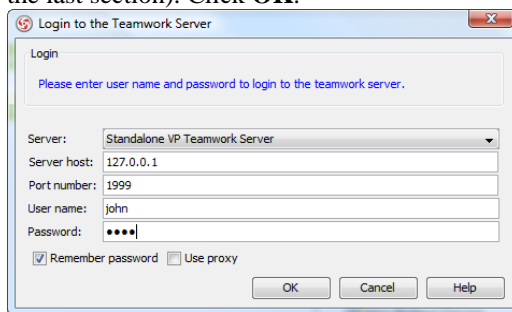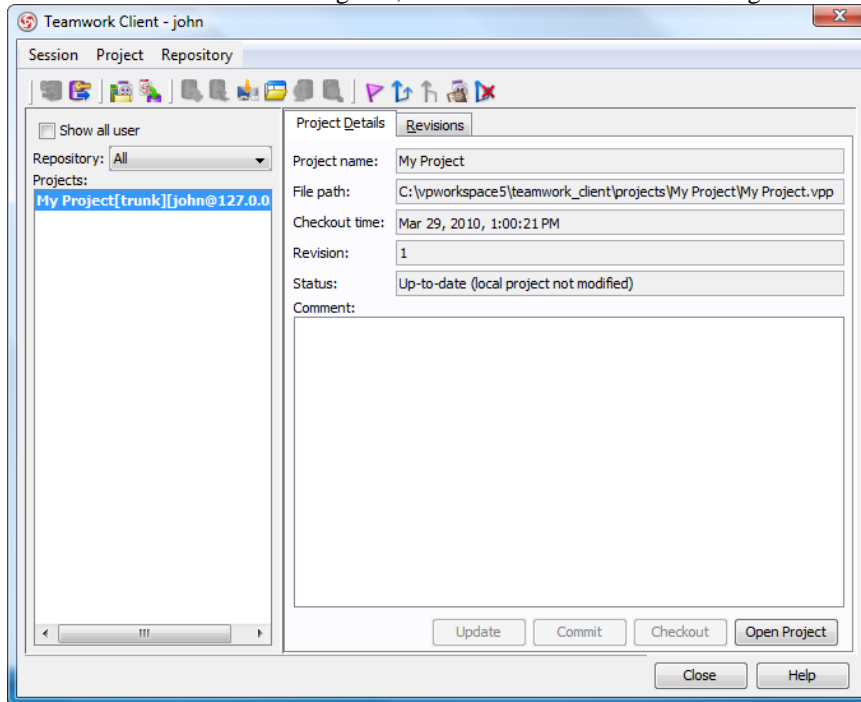
3. In the Manage Project dialog box, select **My Project** and click **>** to manage it. Click **OK** to proceed.
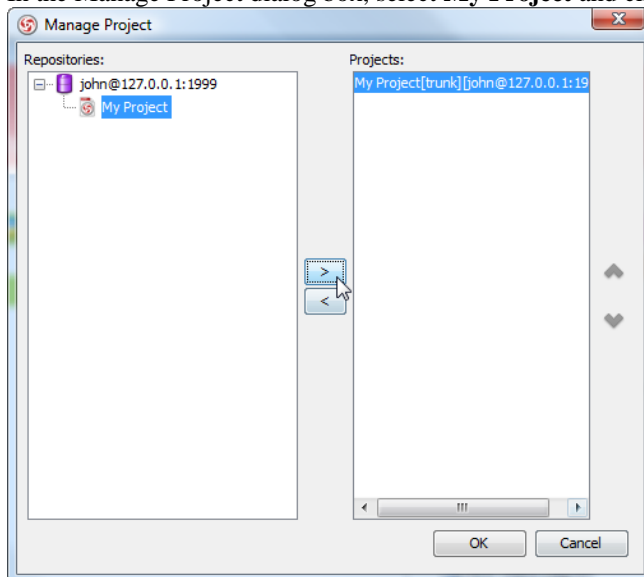
Checkout and open project

Checkout project is to download a managed project from server to your computer. Open project is to open the downloaded project in VP-UML.
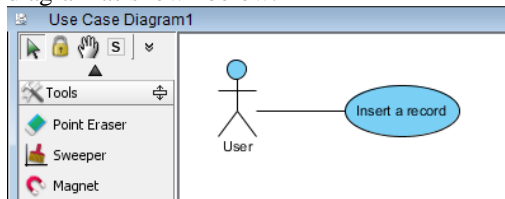
1. In the **Teamwork Client** dialog box, click **Checkout** at the bottom right corner of dialog box.
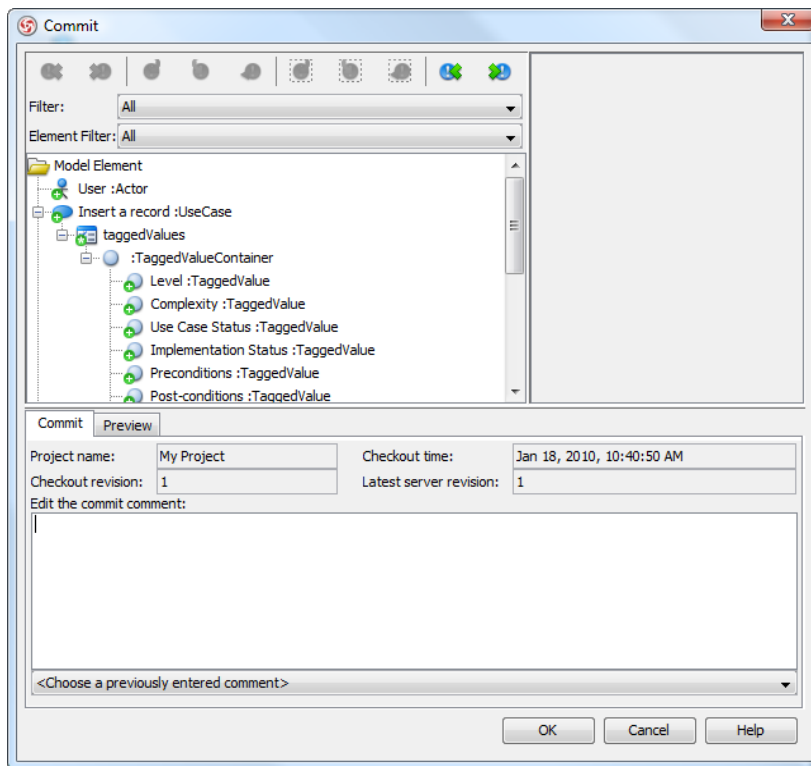

2. Click **Open Project**.

Commit local modification

Commit local modification refers to the process of uploading local modifications to server. Create a simple use case diagram as shown below:

1. Select **Tools > Teamwork > Commit...** from the main menu to commit your changes to server.

2. The **Commit** dialog box displays the changes to be committed to server. Click **OK** to proceed.

Update server changes to local

Update server changes to local refers to the process of getting or downloading changes others have committed to server.

1. Ask another team member to start VP-UML in his/her computer.

2. Follow the steps as listed in the **Login to server** section to login to server as user *peter*.

3. Checkout and open project *My Project*.

4. Open the only use case diagram, and rename the use case.

5. Follow the steps as described in the **Commit** section to commit the change to server.

6. Now, go back to *john's* environment.

7. The **Update** dialog box displays the changes to be updated from server. Select **Diagram > Use Case Diagram1 : UseCaseDiagram > Add a record : Use Case**. Click **Preview**. This is to foresee the changes before actually updating it. In case the change is not desired, you may click **Cancel** to abort the process. Click **OK** button.
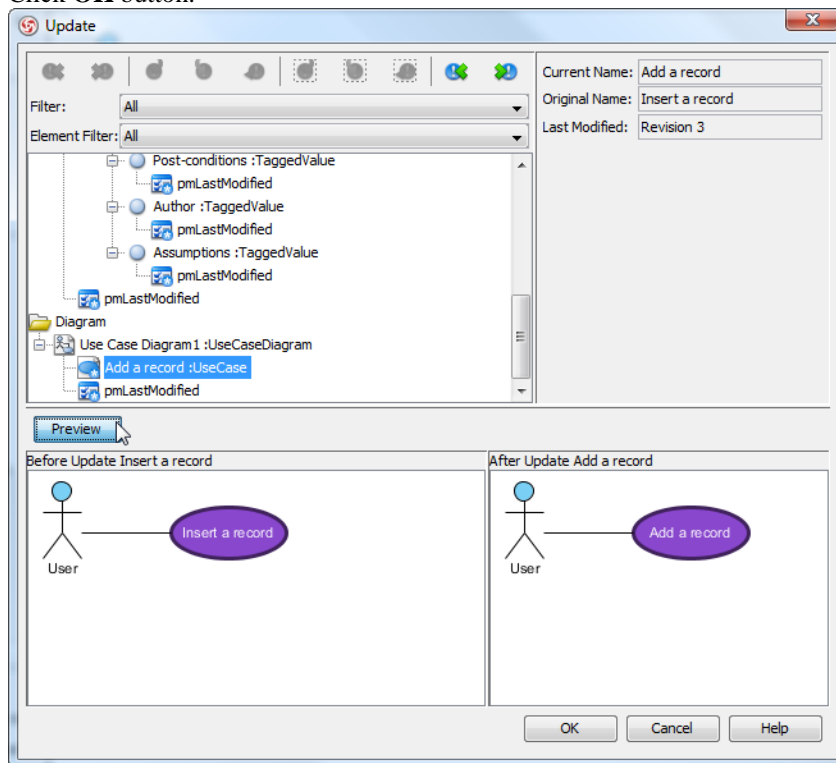
8. Check the use case. Its name has been changed to *Add a record*.