## **Development in Eclipse**

The preconfigured Eclipse projects for two sample plugins and a batch mode (command-line) tool are provided with a program installation. These projects can be found in <modeling tool installation directory>/openapi/ide/eclipse.zip.

Let's use MagicDraw as an example to describe the following procedures.

To setup the Eclipse environment for the modeling tool development

1. Import java projects from < modeling tool installation directory >/openapi/ide/eclipse.zip into your Eclipse Workspace:

1.1. In the main menu, click File > Import. The Import dialog opens.

1.2. In the Select an import wizard list, select General > Existing Projects into Workspace and click Next.

1.3. Copy the java projects zip location to the **Select archive file** box or click **Browse** and browse to the zip file. The project list appears in the **Pro** jects box.

1.4. Select projects you need and click Finish.

Import					
Import Projects		7			
Select a directory to sear		4			
O Select root directory:			~	B <u>r</u> owse	
Select <u>archive file:</u>	Program Files	MagicDraw\openap	i\ide\eclipse.zip ~	B <u>r</u> owse	
Projects:					
MagicDraw (MagicDraw/)			Select All		
MagicDraw with all plugins (MagicDraw with all plugins/) My Commandline (My Commandline/)		Deselect All			
My Plugin 1 (My Plugin 1/)			R <u>e</u> fresh		
<ul> <li>✓ My Plugin 2 (My Plugin 2/)</li> </ul>					
Options					
Search for nested pro	ojects				
Close newly imported	orkspace d projects upon	completion			
Hide projects that all	ready exist in the	e workspace			
Working sets					
Add project to work	Ne <u>w</u>				
Working sets:			$\sim$	S <u>e</u> lect	
?	< <u>B</u> ack	Next >	<u>F</u> inish	Cancel	1

2. Open the **Package Explorer** view, expand the MagicDraw project and edit the MAGIC\_DRAW\_INSTALL\_DIRECTORY link: 2.1. Right-click the link and select **Properties**.

2.2. On the left side of the opened dialog, select **Resource** and click the **Edit** button on the right side of the dialog. The **Edit Link Location** dialog opens.

2.3. Select the location and click OK.

(!)

Be sure the link points to your MagicDraw (or other according to a modeling tool you are using) installation directory (see the following figure).

📑 = 🗐 🕼 🎄 = O = Q = 9	• • 🗑 G • 🕭 🛷 •	▣ ◙ 월 ▾ බ ▾ 박 랴 수 ▾ 수 ▾ ┏				
😫 Package Explorer 🛛	5					
<ul> <li>✓ I MagicDraw</li> <li>&gt; ▲ JRE System Library [JavaSE</li> <li>G MAGIC_DRAW_INSTALL_D</li> </ul>	-11] IRECTORY					
> B My Commandline	Properties for MAGIC_DRAW_INSTALL_DIRECTORY -					
> B≓ My Plugin 1 > B≓ My Plugin 1 Test > B≓ My Plugin 2	type filter text	Resource	⇔ ▼ ⇔ ▼ 8			
	> Resource Run/Debug Settings	Path: /MagicDraw/MAGIC_DRAW_INSTALL_DIRECTORY Type: Linked Folder				
		Edit Link Location   X	Edit			
		Edit Ink location				
	L	ocation: C\Program Files\MagicDraw Folder Variable esolved Location: C\Program Files\MagicDraw				
		⑦ OK Cancel				
	L	nestore benaults	Apply			
	?	Apply and Close	Cancel			

- **3.** Make sure that precisely Java 11 version is used by the **MagicDraw** project in your Eclipse IDE: 3.1. Right-click the **MagicDraw** project and select **Properties**.
  - 3.2. On the left side of the open dialog, select Java Build Path and then select Libraries tab on the right.
  - 3.3. Double-click JRE System Library [JavaSE-11].
  - 3.4. Click Installed JREs and check if Java 11 is available here.

3.5. Click Apply and Close.

(!)

- 3.6. Select Execution environment check box and in the combo box, select JavaSE-11.
- 3.7. Click Environments button.
- 3.8. In the Execution Environments list, select JavaSE-11 and in the Compatible JREs list, select Java 11.
- 3.9. Click Apply and Close and then Finish buttons.
- 3.10. Select Order and Export tab, click Select All button to select all check boxes.
- 3.11. Click Apply and Close.

Eclipse Workspace is ready for the source code development and running/debugging.

To use one of the prepared launch configurations

• On the main menu, click Run > Run Configurations... (or Debug Configurations...). The Run Configurations dialog opens.

If there is no Java 11, add it. Java 11 version must be added, because later versions (e.g. 17) are not supported by our tool.

Run Configurations	
Create, manage, and run configur Run a Java application	rations
Image: Second state of the second	Name: MagicDraw with all plugins   ImagicDraw with all plugins ImagicDraw with all plugins   MagicDraw with all plugins ImagicDraw with all plugins   Main class: ImagicDraw with all plugins   ImagicDraw with all plugins ImagicDraw with all plugins   Main class: ImagicDraw with all plugins   ImagicDraw with all plugins ImagicDraw with all plugins
Filter matched 12 of 12 items	Sho <u>w</u> Command Line Revert Apply
?	<u>R</u> un Close
<ul> <li>The launch configuration is designed to directory (see step #2) and two plugins f directories), developing plugins are not le</li> <li>MagicDraw with all plugins launches Ma Plugin 2.</li> <li>My Commandline launches batch mode</li> </ul>	load plugins from the MagicDraw (or other according to a modeling tool you are using ) installation from the Eclipse Workspace. Thus, if the <i>md.plugins.dir</i> java system property is not defined (see Plugins oaded. agicDraw (or other modeling tool) with plugins available in the IDE workspace namely <i>My Plugin 1</i> and <i>i</i>
<ul> <li>MyPlugin1JUnit<n>Test launches samp</n></li> </ul>	ole JUnit <n> test.</n>
The libraries (jar files) of the plugin must code depends on that plugin. For example, if the code depends on plu (A, B, C, and D) must be added to the cl For more information, see the Identifying	be added to the development class path throughout the plugin dependency hierarchy if the developing Igin A; plugin A depends on plugins B and C; plugin B depends on plugin D, the libraries of all plugins ass path. Ig plugin dependencies page.
When you launch your own plugin, you r can be found in MAGIC_DRAW_INSTAL STALL_DIRECTORY/plugins and its sub	need to add all jar files that are required by your plugin from appropriate plugins. The MagicDraw jar file LL_DIRECTORY/lib and its sub directories, whereas plugins' jar files can be found in MAGIC_DRAW_lib o directories.

Even if the plugin descriptor file contains information about the runtime plugin .jar file, it is not necessary to build and deploy this .jar file to a plugin directory while the plugin is developed under Eclipse.