This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

Composer Deployment Guide

Installation

12/17/2025

Installation

Contents

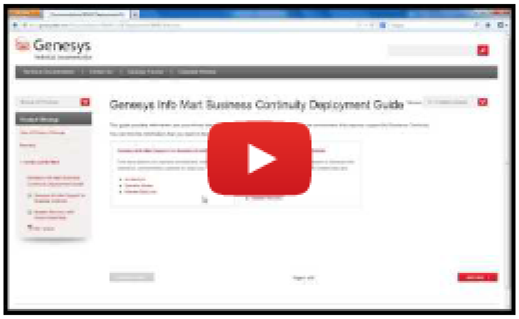
This section describes the Composer software requirements, and installation and launching procedures.

Composer Installation Video

Below is a video tutorial on installing Composer 8.1.3 on Windows in an Eclipse 4.2 environment. Note: When Eclipse 3.7 is installed, there is a small extra step described in the Manual Plugin Installation section below.

Notes:

1. Silent installation of Composer is not supported.
2. Download the correct Eclipse for your computer's processor, i.e., download 64-bit Eclipse download or 32-bit Eclipse based on the target computer.
3. Java Development Kit and Eclipse must match, i.e., both 32-bit or both 64-bit.



Installing Composer as an Eclipse Plugin

Composer 8.1.3 is installed as a plugin into an existing Eclipse 3.7 (Indigo) or 4.2 (Juno) environment as well as Java Development Kit 1.7.0_0 or higher.

Previously, Composer installation consisted of:

1. The "basic" Eclipse application download.
2. Eclipse plugin dependencies.
3. The Genesys Composer plugins.
4. A bundled Tomcat for running generated applications.

Starting with 8.1.3, the Composer installer supplies (3) and (4). The "basic" Eclipse (1) is up to you to download.

Update Site

The Composer plugins are distributed as an *update site*, which is contained in *com.genesyslab.composer.updateSite.zip* . In Eclipse terminology, “update site” refers to a location from which Eclipse can pull plugins. The installer gives the option to pull the plugins from the update site zip file and install them into an existing Eclipse (1) environment.

If that option is enabled, then the Installer runs a script that automatically finds and downloads the plugin dependencies (2) and installs them along with the Composer plugins (3) and the bundled Tomcat (4). This is optional because you may not have downloaded a basic Eclipse yet, in which case you can install (2) and (3) later from within Eclipse using **Help** > **Install New Software** .

Note: Eclipse is not required if you only want to extract the update site without installing Composer plugins into the Eclipse. This is possible, for example, if you just want the update site zip file to copy to multiple machines where the actual Composer installation will take place.

User-Facing Changes as a Result of Plug-in Installation

Change	Before 8.1.3	813 Behavior
Change in file encoding used by Eclipse-based text editors to save files.	Composer based editors such as the VoiceXML Editor and SCXML Editor uses UTF-8 Encoding. Other Eclipse-based generic test editors uses default CP1252 encoding.	All the files and editors within the IDE will use UTF-8 encoding if the General > Workspace > Text File Encoding preference is default while launching the IDE. Otherwise, user-specified encoding takes precedence. However, Composer recommends using UTF-8 encoding.
The Composer splash screen is not displayed anymore.	Composer branded splash screen was displayed showing the Genesys logo, version, and so on.	Eclipse's default screen is displayed. No Genesys specific information shown.
Help > About	The Help About dialog is integrated into Eclipse and displays as part of the Eclipse workbench with information pulled out of the product definition.	The About Composer dialog exists is now a custom implementation. There is now a Help > About Composer in addition to a Help >> About.

Operating Systems Supported

For information on supported operating systems, see the Composer section in the Genesys Supported Operating Environment Reference Guide. Also see the note on Windows 7 and Windows 8 Server under Installing Composer on Wndows.

Application Server Requirements

- Note: For more detailed information on deploying Composer applications to an application server, see Deploying Composer Applications.

Genesys does not certify Composer with specific Web application server vendors and versions. This applies to developing both:

- VXML applications and related resources that will be executed on the Genesys Voice platform and
- SCXML applications and related resources that will be executed on the Orchestration platform.

Java Composer Projects can be deployed to any web application server that meets the requirements in the Deploying Composer Applications topic and supports Java Runtime Environment 1.7.0_0 or higher.

When considering a potential Web application server to use for a production deployment of Composer-generated applications, please refer to the vendor's documentation to ensure that the software meets these pre-requisites. In addition, you should run a few basic tests using a Composer sample application, to ensure that the application behaves as expected. For more details, please refer to the section Suggested Test Plan.

For developing SCXML applications and related resources that will be executed on the Orchestration Server platform, see the Genesys Supported Operating Environment Reference Guide.

Bundled Tomcat

Composer installs an embedded Tomcat 6.0 web server for your use (code generation and testing). Genesys does *not* recommend that you use this bundled Tomcat web server for deploying and running your Composer-generated applications as part of a production setup. You may use a separate instance of Tomcat 6.0 as a stand-alone web server, depending upon your needs and the recommendations of your IT department. For deployment of Composer-generated applications that use .NET resources, you can use any version of Microsoft IIS that is compatible with the Windows versions on which Composer is supported.

Web Application Server Configuration

For Composer's server-side pages to work effectively on your Web application server, some configuration changes are required.

Tomcat

See the following sections ahead:

- Configuring Tomcat Settings
- Configuring Proxy Settings

Internet Information Services (IIS)

See the following sections ahead:

- Internet Information Services
- Configuring IIS/.NET Preferences
- Adding MIME Types for IIS
- Proxy Settings for .NET Composer Projects

Suggested Test Plan

After configuring your Web application server as described above, you should run some basic tests. Here is a suggested testing approach:

1. Create a new Project based on Project templates supplied with Composer. The New Project Creation wizard will guide you through the process and show a list of Project templates to choose from. Choose a template Project depending on the Project type and feature(s). Note: If you are using both the voice and route features, Genesys recommends that you test both features by running two tests.

Composer Feature	Not Using Databases	Using Databases
Voice	Business Logic Project	Database Project
Route	Routing Using Web Request Project	Database Query Result Access Project

2. These sample Projects may require configuration to be done, which will be documented in the workflow or callflow diagram(s) in the Project template. Projects that access databases contain a *readme.htm* file in the *doc* folder that provides instructions on how to set up the database as well as SQL scripts that may be needed to set up the required database structure and populate tables with sample data.
3. Validate the diagrams in your Project and verify there are no errors. Generate the code for these diagrams.
4. Export the Project for deployment. See the Deployment book in the Composer Help for the steps.
5. Consult the documentation for your application server on how to deploy applications on it. Some application servers may require custom steps.
6. Once deployed successfully, make a test call to invoke the application. Verify the application behaves as expected. If it does not, check the configuration and Troubleshooting book in the Composer Help for more information.

Database Support

Composer 8.1 supports the following databases/servers:

- Microsoft SQL Server 2005 and 2008. Note: When installing SQL Server, select SQL Server authentication (Composer does not support integrated Windows authentication).
- Oracle 10g, R1 and Oracle 11g.

Note: Before you can use database blocks in a .NET Composer Project for accessing an Oracle database, you need to install and configure the Oracle client on the Composer machine. The Oracle client will be required on any deployment machines as well where the application will run. As a test, after installing the client software you should be able to connect to the Oracle database from SQLPlus. Once that works, database blocks in your .NET Composer Project should also be able to connect to your Oracle database. For configuring Oracle client, please contact your Oracle database administrator. Composer 8.1 has switched to an Oracle Provider from a Microsoft Provider for .NET Composer projects.

- At design time, both Java Composer Projects and .NET Composer Projects use bundled JDBC drivers to connect to Oracle, which is why your query will work in the Query Builder in both types of Composer projects. At runtime, Composer .NET projects use OLEDB database drivers, which are installed as part of Microsoft.NET Framework. These are required on each IIS where .NET Composer Projects will be run or deployed.
- Composer Java Projects use JDBC (Java Database Connectivity) drivers, which are bundled with Composer and are automatically installed with the software.
- Java Composer Projects continue to use JDBC, while .NET Composer Projects use Microsoft's OLEDB providers from the .NET Framework.

Speech Engines Supported

Composer supports all Automatic Speech Recognition (ASR) and Text-to-Speech (TTS) engines that GVP 8.1 supports.

Web Browsers Supported

Composer supports the following web browsers:

- Microsoft Internet Explorer 7.0 and 8.0
- Mozilla Firefox 6.0 or earlier

Third Party Software Requirements

Composer requires the following third-party software on the computer on which Composer is installed:

1. **Eclipse must already be installed in your environment.** Composer 8.1.3 supports the following Eclipse versions: Eclipse. 3.7 (Indigo) and 4.2 (Juno).
    - Download the correct Eclipse for your computer's operating system, i.e., download 64-bit Eclipse or the 32-bit Eclipse based on your computer.
    - If installing Eclipse Juno EE 32-bit version,you must install the Oracle modeling plug-in, which Composer requires, but is not included with the Eclipse Juno EE 32-bit download. Note that many other dependencies are downloaded during the process that adds the Composer plugins to Eclipse. Genesys recommends using a dedicated instance of Eclipse for Composer.

2. **Java Development Kit 1.7.0_0 or higher must already be installed in your environment.**
   - Set the JAVA_HOME environmental variable to point to the JRE installation directory as described on the Oracle website. You must explicitly point to the Composer-supported version of JRE (1.7 or higher) and not use the system environment variable location. This is important if you have different versions of Java installed as the system environment variable may indicate another version, which may not be usable by the Composer version of Eclipse.
   - The above warning also applies to JDK installation. The JDK's JRE version could get lost if there are other JREs on a host. If you previously installed JDK separately (or did not enable the Public JRE option), JDK 1.7 installation by itself might require the JRE 1.7 installation to be explicitly called out. A symptom of this type of JRE confusion is Composer perspectives not being available after installation.
   - Java Development Kit and Eclipse must match, i.e., both 32-bit or both 64-bit.
3. **Microsoft .NET Framework** 2.0 and 3.5 (both are required if you intend to work with .NET Composer Projects).
4. **Web Service Enhancements** (WSE) 3.0 for Microsoft .NET. The WSE path must be specified in Composer’s IIS/.NET preferences before Composer .NET Projects can work.

Note: Composer 8.1 does not support silent installation.

Genesys Software Prerequisites

To obtain the full functionality of Composer 8.1, the following Genesys products/software components are required:

- Orchestration Server 8.1.3 for debugging and testing SCXML session-based applications.
- Genesys Voice Platform (GVP) Media Control Platform 8.1.6 for video text overlay and Context Services authentication.
- If you wish to use Context Services in routing workflows and voice callflows, you will need Universal Contact Server/Context Services 8.1,000,10+.
- If you wish to use the Outbound blocks, you will need Outbound Contact Server 8.1.100.09+.
- Genesys Rules System 8.1.0 or later for business logic, which can be customized, and then invoked by VXML and SCXML applications.

Minimum System Requirements

Genesys recommends the following:

- Pentium 4 2GHz or comparable, 2 GB RAM or higher.

Minimum Screen Resolution

The minimum resolution for the Composer user interface is 1024x768 on a standard 4:3 aspect ratio monitor. The recommended resolution is 1280x1024. Lesser resolutions, such as 800x600, are not supported.

Before Installing

Before you install the Composer plugin, make sure that:

- You have met the **third party software requirements**.
- You have an account that has administrative privileges to install Composer.
- You have installed a supported SIP phone on your desktop (required if you want to conduct test calls). This can be done before or after installing Composer.
- You have installed Microsoft .NET Framework 2.0 *and* .NET Framework 3.5, for ASP.NET support in Composer.
- You have reviewed the Composer 8.1 Release Advisory.
- You have read the important information on User Account Control(UAC) (see note under Installing Composer on Windows).
- You have reviewed the chapter on configuring a security banner in the Genesys Security Deployment Guide as well as the Security Banner Configuration section below. If configuring a security banner, decide whether you want to configure it before or after installation of Composer.
- YOu are not installing through a Microsoft Remote Desktop Connection. You should perform the installation locally. Genesys does not recommend installation of its components through Remote Desktop Caonnection.
- Composer 8.1 does not support silent installation.

Security Banner Configuration

Decide whether you wish to configure a security banner.

Composer installation gives the option of configuring a security banner that displays when users connect to Configuration Server. In the case of Composer, the security banner appears in a separate window that is displayed when you connect to Configuration Server. The content of this window is defined by the system administrator, and can include such items as Terms of Use of the application or some kind of disclaimer.

While Composer supports the basic operation of specifying and displaying a security banner as described in the Genesys Security Deployment Guide, be aware of the following limitations for Composer:

- The banner only displays when you connect to Configuration Server after launching Composer.
- The following options are not supported:
  - List of security banner URLs.

- **Error URL** if banner cannot be retrieved.
- **ShowUpTimeout** , which displays a *Loading* message if loading takes more than this timeout.
- **NoCompleteTimeout** , which is the timeout for retrieving security banner.
- **AckMandatory** , which controls if user must acknowledge the security banner in order to proceed.
- Height, width and title of dialog.
- **AckMode** , which controls how often the banner is displayed. Note: A Java equivalent of unilogin library is not available.

- The security banner configuration option *Until each user chooses to turn it off once for application type* is not supported. It is handled the same as the option *Until each user chooses to turn it off* .

Under some circumstances, Composer may not try to display a banner page although the loading of a previous page in the list failed. This behavior is likely to happen if the embedded browser component returns that loading is finished (although the loading failed) before the loading timeout elapses.

Pending Operations Message

When installing or un-installing Composer in a Windows XP Professional or Windows Server 2003 Standard Edition environment, the prompt to reboot appears. After the reboot, the following message appears:

```
There are some pending operations and the system needs a reboot.
The target computer does not meet some mandatory requirements.
```

Subsequent reboots result in the same behavior and you temporarily cannot install Composer on this machine. This is a result of pending reboots from other installations. If this situation occurs, follow the procedure below.

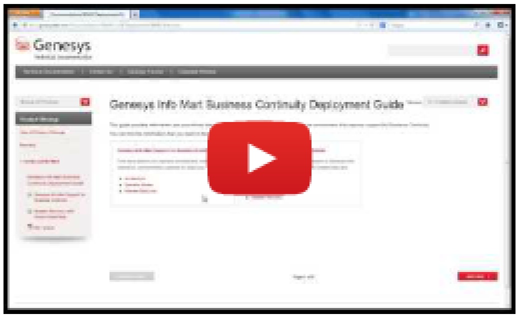
Registry Edit for Pending Operations Message

Use the information below if you receive the above message after un-installing or installing Composer:

1. Open the Registry editor.
2. Remove the following key: **HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session Manager\PendingFileRenameOperations**
3. Install Composer as described below.

Installing Composer on Windows

A video of the Composer 8.1.3 installation procedure is available here:

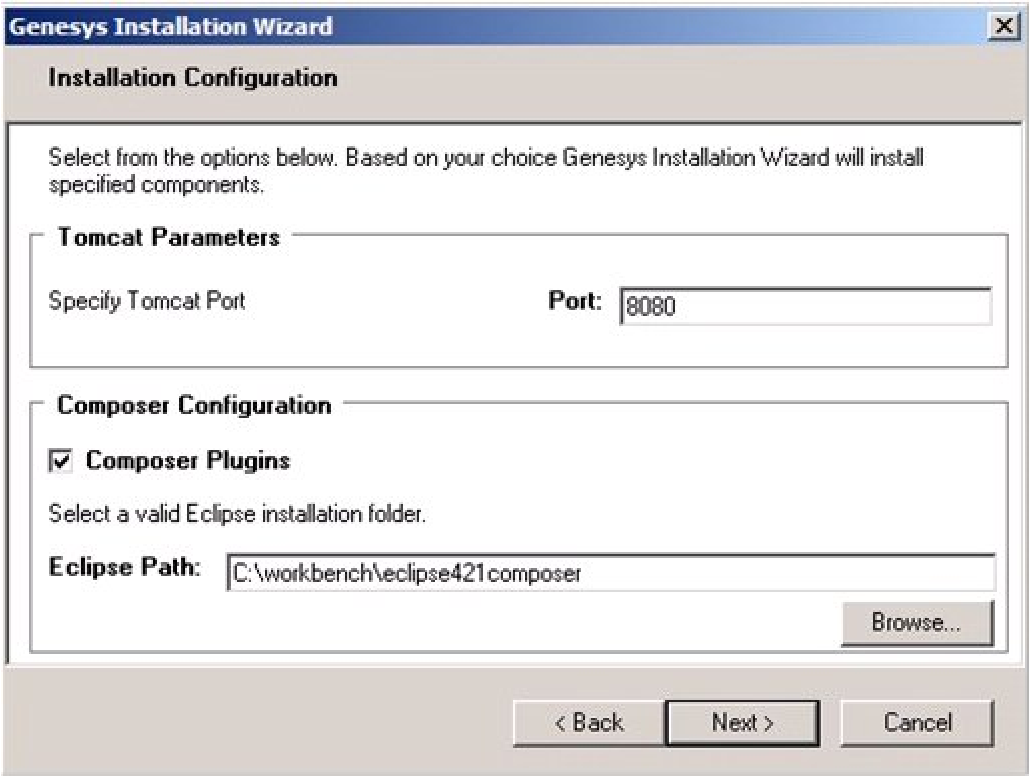


- **Note:** On Windows 7 and Windows Server 2008, UAC (User Account Control) can affect the installation process and subsequent addition of Composer plugins to Eclipse. If UAC is disabled, installing Eclipse in a Program Files location (any location that is under UAC control) will work. However, if UAC is enabled, then Eclipse should be installed in a location that is not under UAC control e.g., C:\GCTI\Composer.

- **Note:** While installing Composer 8.1.3, there is a step to download dependencies from Eclipse update sites. If your Internet connection is lost during this process, or some other error occurs, the Installer may not detect it and may incorrectly report success. Error messages will appear in the *scripts\composer-plugin-install.log* file in the Composer installation directory, (Example: *C:\Program Files\GCTI\Composer\scripts\composer-plugin-install.log*). Workaround: Instead of letting the Installer add Composer plugins directly to Eclipse, add Composer plugins using Eclipse's "Install New Software" feature (see Manual Plugin Installation below).

Prerequisites: See Third Party Software Requirements.

1. Run **setup.exe** from the Composer installation package.
2. If you want to configure a security banner, see Security Banner Configuration section above.
3. Enter **Destination** . The destination is also where com.genesyslab.composer.updateSite.zip will be located.
4. When you reach the screen shown below, you can choose to:
    - Leave the **Composer Plugins** box unchecked and manually install the Composer plugin. A common case for leaving Composer Plugins unchecked would be if Eclipse is not yet installed or you do not have internet access so that dependencies cannot be downloaded during the installation process. See Manual Plugin Installation section below.
    - Check the **Composer Plugins** checkbox and browse to the existing Eclipse location. This option automatically installs *com.genesyslab.composer.updateSite.zip* into your Eclipse installation.



Select from the options below. Based on your choice Genesys Installation Wizard will install specified components.

Specify Tomcat Port — Port: 8080

☑ Composer Plugins

Select a valid Eclipse installation folder.

Eclipse Path: C:\workbench\eclipse421composer

5. Specify the Tomcat port. For testing purposes, Composer supports automated deployment of routing applications to the bundled Tomcat server or to a local IIS server. Enter a port number to be used for Tomcat.

6. Click **Next** and **Finish** on the installation wizard. The installation may take some time because other plugin dependencies are also downloaded and installed.

Next Steps After Automatic Plugin Installation

- See Composer Interface.
- To uninstall/reinstall a later version, see Uninstalling and Reinstalling.

Manual Plugin Installation (Eclipse Install New Software)

If you left **Composer Plugins** unchecked, you can later manually install the Composer plugins if:

- You have an internet connection
- You have installed the required versions of Eclipse and Java Development in your environment.
- You have enabled the Public JRE option, which causes the JDK installer to install both JDK and JRE. Composer requires both components.

The Composer installation will have created a directory, e.g., *C:\Program Files\GCTI\Composer 8.1* , in which there is a zipped update site called *com.genesyslab.composer.updateSite.zip* .

1. In a running Eclipse instance, go to **Help** > **Install New Software** .
2. If installing Composer in Eclipse Indigo (3.7), the following extra steps are required.
   - On the Eclipse **Install'** ***dialog box, click*** *Add* .
   - Opposite **Name** , enter **GMF Tooling** .
   - Opposite Location, enter the URL for GMF Toolng: *http://www.eclipse.org/modeling/gmp/downloads/?project=gmf-tooling*
   - Click **OK.**
3. Click **Add.**
4. Type something in the *Name* field, such as *Composer* .
5. Click **Archive** .
6. Browse to the *com.genesyslab.composer.updateSite.zip* file in the *Destination* folder you previously specified in the **Destination** field. The destination is also where com.genesyslab.composer.updateSite.zip will be located.
7. Click **OK** .
8. Check the three Composer Features (*Common*' , **Voice, Route**).
9. Click **Finish** , and click through the license pages. The installation will now begin. Again, this may take some time because other plugin dependencies are downloaded and installed.
10. After installation is complete, the Installer will prompt you to restart Eclipse. **Click OK** .

Displaying the Interface

1. Click the Eclipse shortcut on your desktop or run **eclipse.exe** . The Eclipse splash screen displays (no Genesys-specific information is shown). A workspace dialog box opens to allow you to select the location for your project files.
2. Accept the default entry, or click **Browse** to navigate to a location that will serve as your workspace folder. Note: Your workspace folders must be outside of the Program Files folder. When prompted for a workspace folder, do not specify parenthesis '(' or ')' in the workspace path. In Windows, this is C:\Users\<username>. On Mac, this is /Users/<username>.
3. If you want your selected workspace to be your default and do not want to select a location the next time that Composer opens, select the **Use this as the default and do not ask again** check box.
4. Click **OK** ***to proceed. The first time you run Eclipse, a*** *Welcome*' tab appears.
5. If you choose not to explore the Eclipse or Composer tutorial links at this time, click the **X** on the **Welcome** tab to display the Composer GUI. You can always access the Welcome screen at any time from within Composer by selecting **Help** > **Welcome** .
6. If not already open, open the Composer perspective by selecting **Window** > **Open Perspective** > **Other** , then select **Composer** or **Composer Design**.

At this point, you may find it helpful to load a Project template so you can see a sample application. Follow the procedure in the next section if you wish to do this.

Windows 7 and Server 2008 32 Bit

The procedure below also applies to 64-bit Windows 7 Ultimate in 32-bit compatibility mode.

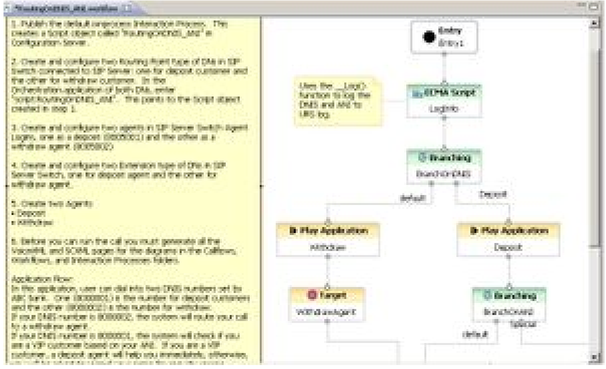
1. Run / launch Eclipse as Administrator.
2. DO NOT keep your workspace under the *Program Files* location as prompted by Composer. Instead specify your workspace location outside of the *Program Files* location e.g.,
   C:\My Composer 8.1.4 Applications or
   C:\Users\<your user>\MyWorkspaceFolder

Viewing a Sample Application

Composer provides a set of predefined Project templates containing sample applications. If using Composer for the first time, before creating your own Project, you may find it helpful to load one or more of the sample Projects. The procedures below lead you through the process of loading a sample applications.

Sample Integrated Voice and Route Application

1. By default, when you enter Composer for the first time, you will be taken inside the Composer perspective.
2. On the toolbar, click the button to create a Java Composer Project.
3. In the **Java Composer Project** dialog box, name your Project and indicate whether you want to use the default location.
4. Select the Project type: **Integrated Voice and Route' ,** *Voice* , or **Route** . Your selection will determine which Project templates are shown in the next dialog box. For this example, select *Integrated Voice and Route* .
5. Click **Next** . The **Select a Composer Project Template** dialog box opens.
6. Select the **Routing Based on DNIS and ANI** Project template and click **Finish** . A commented *RoutingOnDNISandANI.workflow* diagram appears on the canvas.



*RoutingOnDNIS_ANI.workflow

1. Publish the default orchestration Interaction Process. This creates a Script object called "RoutingOnDNIS_ANI" in Configuration Server.

2. Create and configure two Routing Point type of DNs in SIP Switch connected to SIP Server: one for deposit customer and the other for withdraw customer. In the Orchestration application of both DNs, enter "script:RoutingOnDNIS_ANI". This points to the Script object created in step 1.

3. Create and configure two agents in SIP Server Switch Agent Logins, one as a deposit (8005001) and the other as a withdraw agent (8005002).

4. Create and configure two Extension type of DNs in SIP Server Switch, one for deposit agent and the other for withdraw agent.

5. Create two Agents:
• Deposit
• Withdraw

6. Before you can run the call you must generate all the VoiceXML and SCXML pages for the diagrams in the Callflows, Workflows, and Interaction Processes folders.

Application Flow:
In this application, user can dial into two DNIS numbers set by ABC bank. One (8000001) is the number for deposit customers and the other (8000002) is the number for withdraw.
If your DNIS number is 8000002, the system will route your call to a withdraw agent.
If your DNIS number is 8000001, the system will check if you are a VIP customer based on your ANI. If you are a VIP customer, a deposit agent will help you immediately, otherwise,

Uses the __Log() function to log the DNIS and ANI to URS log.

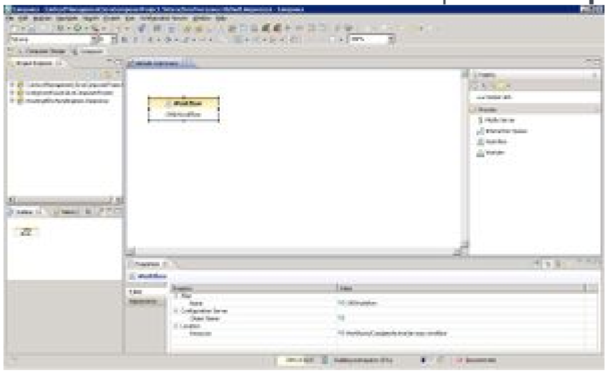
Double-clicking a block opens its **Properties** view so you can view the fields.

The next procedure differs from the above procedure in that it demonstrates how to view a workflow contained within an interaction process diagram.

Viewing a Routing Workflow Inside an Interaction Process Diagram

By default, when you enter Composer for the first time, you will be taken inside the Composer perspective.

1. On the toolbar, click the button to create a Java Composer Project.
2. In the **Java Composer Project** dialog box, name your Project and indicate whether you want to use the default location.
3. Select the Project type: **Integrated Voice and Route** , **Voice** , or **Route** . For this example, select **Route** .
4. Click **Next** . The **Select a Composer Project Template** dialog box opens.
5. Select **Context Services Service Project** click **Finish** . This automatically creates an interaction process diagram for voice interactions with a single Workflow block in the *default.ixnprocess* tab.
6. Double-click the Worfklow block to open the **Properties** view in the tab underneath.



7. In the **Properties** view, note that the **Resource** property indicates that the name of the workflow is *CompleteActiveServices.workflow* . To view this workflow, expand the Project in the **Package Explorer** on the left.
8. Expand the **Workflows** folder.

9. Double-click *CompleteActiveServices.workflow* . A commented workflow appears.
10. View the properties for each block by double-clicking a block.

Use the above method to review the various routing Project templates.