IdMUnit

Installation Guide

Prepared for **Public Release (GNU GPL)**

TriVir

11570 Popes Head View Lane
Fairfax, VA 22030
www.trivir.com

January 10, 2007 v.1.2

Summary

IdMUnit is a must-have automated testing tool that integrates with Designer for Identity Manager. This suite, based on xUnit and Spring Framework architecture will automatically transform documented test cases into actual automated tests, execute them, and record the results (pass/fail/warning) in text and/or HTML output. This suite provides integrated policy, driver and system testing to accelerate the project release time-frame, ease development risk, control scope, and ensure logical compliance with functional requirements from design until deployment and beyond. Complete functional testing can now be completed for the entire identity management solution in seconds rather than weeks.

IdMUnit provides a quality-control compliment to Identity Manager as it quickly certifies that installed drivers are functioning according to business rules and solution specifications. It also works with TriVir's Data Comparator, which validates the state of every synchronized object and attribute to ensure that synchronization is working as expected throughout connected production systems. TriVir is is working on the release of a freely-distributable, open-source version of IdMUnit. If interested in participating or receiving a demo of the software, please send a request to info@trivir.com to obtain more information.

This document contains instructions for the installation and basic use of IdMUnit. More information can be obtained by emailing info@trivir.com or going to www.idmunit.org.

License information:

www.TriVir.com
TriVir LLC
11570 Popes Head View Lane
Fairfax, Virginia 22030

IdMUnit

Keep the bar green to keep the solution clean!

Installation

The installation of IdMUnit consists of the following high-level steps:

- Download Designer for Identity Manager (Designer)
- Download the vanilla Eclipse 3.2 SDK
- Overlay Designer installation with Eclipse 3.2 (vanilla configuration)
- Install IdMUnit libraries
- Configure and execute IdMUnit tests

Note that IdMUnit 1.2 has been tested and validated with Designer 1.2 (the current shipping release as of January 2007) and with pre-release versions up to Designer 2.0 M5.1. Using IdMUnit with an officially tested/supported version of Designer is highly recommended. However, if a newer version of Designer is already being used, IdMUnit may be imported following the instructions in this document (skipping the step of downloading/installing Designer). Issues running IdMUnit with any newer release builds of Designer may be submitted to the IdMUnit project at http://sourceforge.net/projects/idmunit.

Also note that the instance of Designer running IdMUnit does not need to be the same instance of Designer where the project is installed. Running IdMUnit in one installation and/or instance of Designer while running another version of Designer is fully supported for those that like to run the nightly-builds of Designer and don't want to import IdMUnit into each installation of Designer.

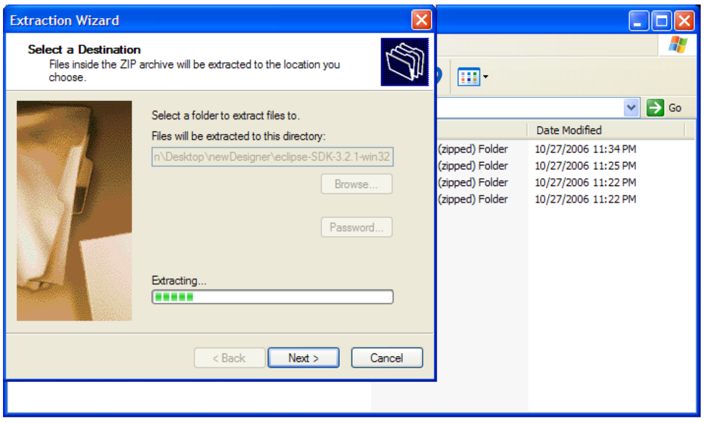
At least a couple GB of hard disk space is required in order to follow these instructions. 1GB RAM or more is recommended.

Note that this documentation assumes that the target audience has experience in each of these areas:

- Novell Identity Manager
- Designer for Novell Identity Manager
- Microsoft Excel or OpenOffice Calc spreadsheets (note that at least one of these products should be installed on the IdMUnit machine in order to manipulate test spreadsheets)

1. Installation Steps

1.1. Download and extract the latest built of Eclipse 3.2 to a temporary directory. Novell's packaged installation of Designer eliminated pieces of Eclipse that were not being used in order to streamline the install size. IdMUnit needs to leverage some additional Eclipse features not included in Designer distribution. Those dependencies will be included in the Eclipse archive. Eclipse may be downloaded from http://www.eclipse.org/downloads/



Extraction Wizard

Select a Destination

Files inside the ZIP archive will be extracted to the location you choose.

Select a folder to extract files to.

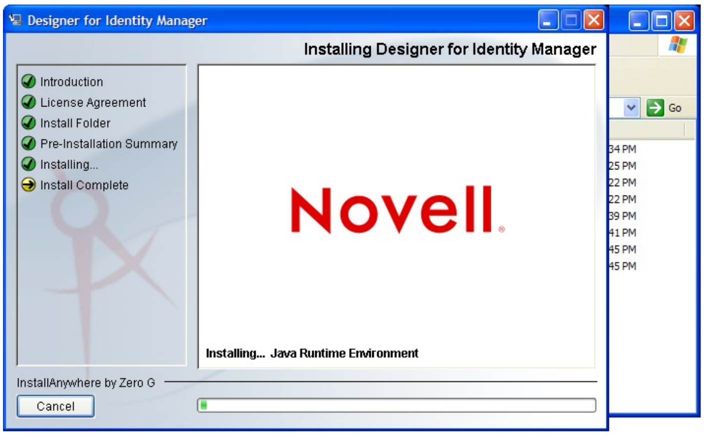
Files will be extracted to this directory:

n\Desktop\newDesigner\eclipse-SDK-3.2.1-win32

Extracting...

(zipped) Folder 10/27/2006 11:34 PM
(zipped) Folder 10/27/2006 11:25 PM
(zipped) Folder 10/27/2006 11:22 PM
(zipped) Folder 10/27/2006 11:22 PM

1.2. (Skip this step if Designer 1.2 or later is already installed) Download and install Novell Designer for Identity Manager from http://www.novell.com/coolsolutions/dirxml/designer/

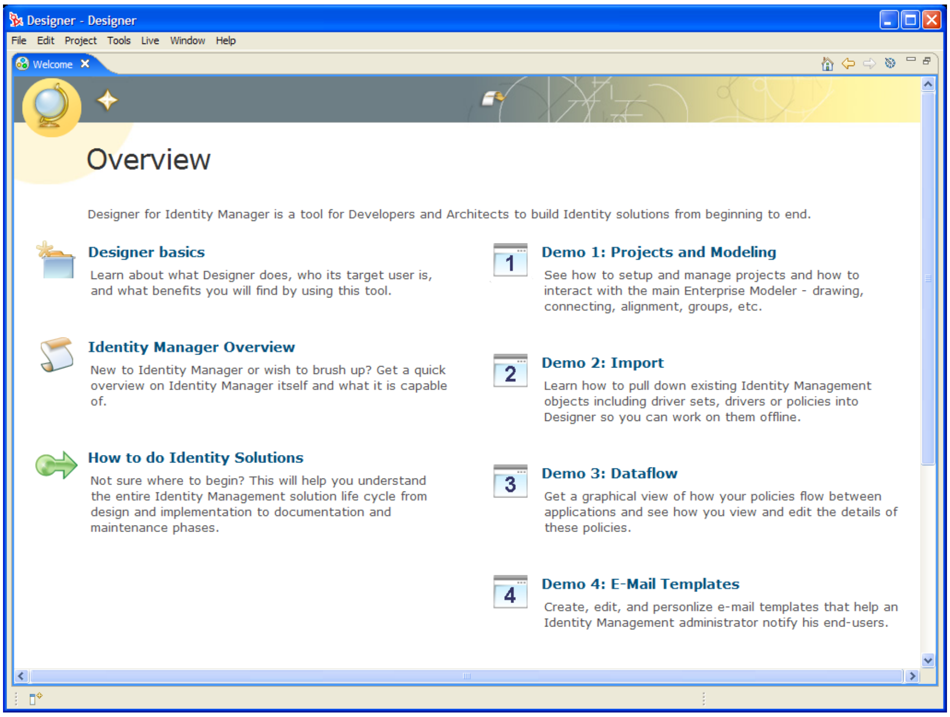


Designer for Identity Manager

Installing Designer for Identity Manager

Installing... Java Runtime Environment

1.3. Run and validate the Designer installation.



Overview

Designer for Identity Manager is a tool for Developers and Architects to build Identity solutions from beginning to end.

Designer basics
Learn about what Designer does, who its target user is, and what benefits you will find by using this tool.

Identity Manager Overview
New to Identity Manager or wish to brush up? Get a quick overview on Identity Manager itself and what it is capable of.

How to do Identity Solutions
Not sure where to begin? This will help you understand the entire Identity Management solution life cycle from design and implementation to documentation and maintenance phases.

Demo 1: Projects and Modeling
See how to setup and manage projects and how to interact with the main Enterprise Modeler - drawing, connecting, alignment, groups, etc.

Demo 2: Import
Learn how to pull down existing Identity Management objects including driver sets, drivers or policies into Designer so you can work on them offline.

Demo 3: Dataflow
Get a graphical view of how your policies flow between applications and see how you view and edit the details of these policies.

Demo 4: E-Mail Templates
Create, edit, and personlize e-mail templates that help an Identity Management administrator notify his end-users.

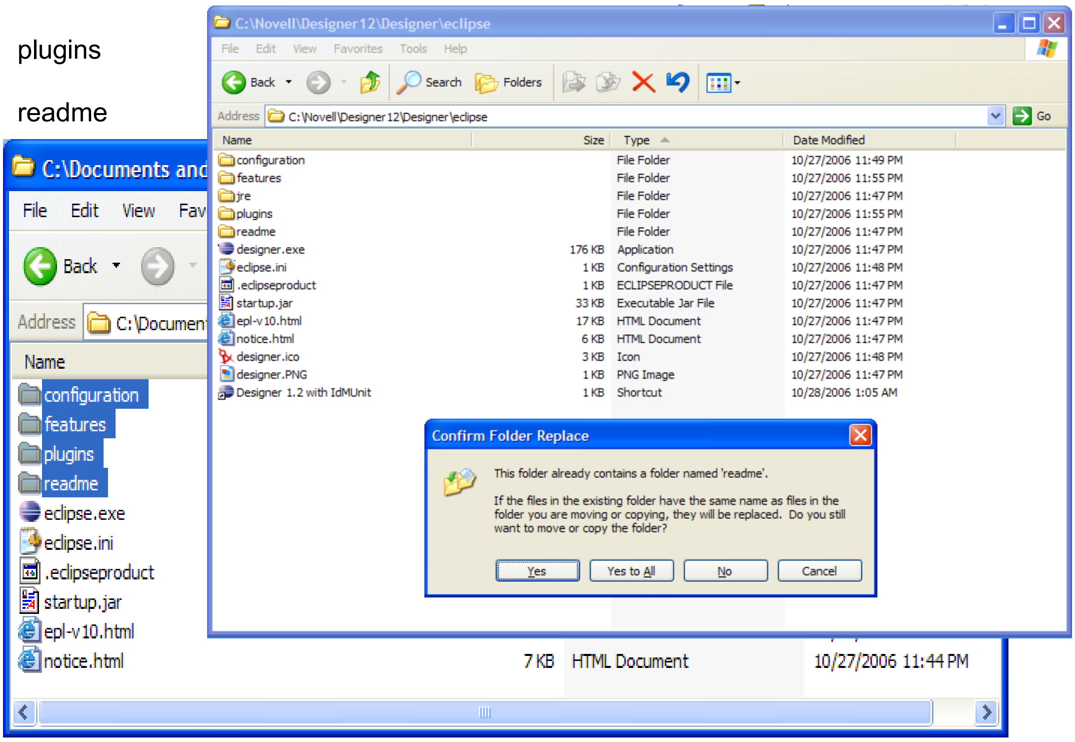
1.4. Next, overlay the latest Eclipse libraries over the Designer installation. This will enable standard Eclipse functionality (like the JUnit plug-in) that does not currently ship with Designer. Close Designer and copy the following extracted Eclipse 3.2 directories over the corresponding directories in the Designer installation directory. (Note that the Eclipse logo will now be displayed when loading Designer. In order to re-enable the Designer logo, copy the config.ini file from the Designer installation archive into eclipse/configuration).

a) configuration

b) features

c) plugins

d) readme


C:\Novell\Designer12\Designer\eclipse
File Edit View Favorites Tools Help
Back
Search
Folders
Address C:\Novell\Designer12\Designer\eclipse
Go
C:\Documents and
File Edit View Fav
Back
Address C:\Documen

Name	Size	Type	Date Modified
configuration		File Folder	10/27/2006 11:49 PM
features		File Folder	10/27/2006 11:55 PM
jre		File Folder	10/27/2006 11:47 PM
plugins		File Folder	10/27/2006 11:55 PM
readme		File Folder	10/27/2006 11:47 PM
designer.exe	176 KB	Application	10/27/2006 11:47 PM
eclipse.ini	1 KB	Configuration Settings	10/27/2006 11:48 PM
.eclipseproduct	1 KB	ECLIPSEPRODUCT File	10/27/2006 11:47 PM
startup.jar	33 KB	Executable Jar File	10/27/2006 11:47 PM
epl-v10.html	17 KB	HTML Document	10/27/2006 11:47 PM
notice.html	6 KB	HTML Document	10/27/2006 11:47 PM
designer.ico	3 KB	Icon	10/27/2006 11:48 PM
designer.PNG	1 KB	PNG Image	10/27/2006 11:47 PM
Designer 1.2 with IdMUnit	1 KB	Shortcut	10/28/2006 1:05 AM

Confirm Folder Replace

This folder already contains a folder named 'readme'.

If the files in the existing folder have the same name as files in the folder you are moving or copying, they will be replaced. Do you still want to move or copy the folder?

Yes | Yes to All | No | Cancel

Name	Size	Type	Date Modified
configuration			
features			
plugins			
readme			
eclipse.exe			
eclipse.ini			
.eclipseproduct			
startup.jar			
epl-v10.html			
notice.html	7 KB	HTML Document	10/27/2006 11:44 PM

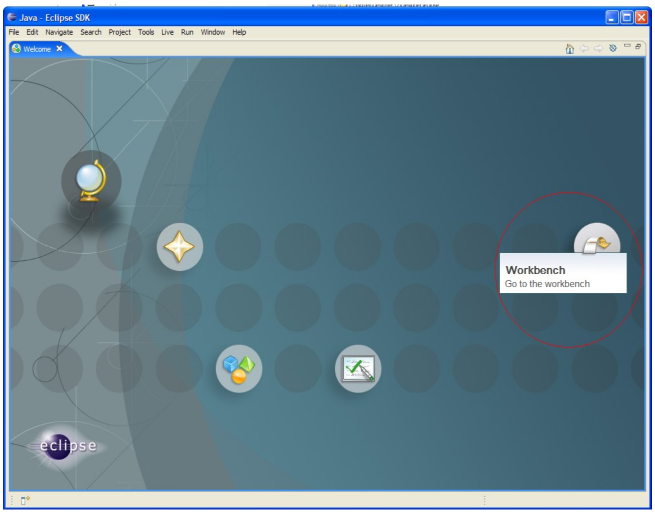
1.5. Download a copy of the appropriate version of IdMUnit from http://sourceforge.net/projects/idmunit. **IdMUnit versions correspond to the latest shipping release of Designer (i.e. v.1.2). However, IdMUnit should also work fine with stable/tested pre-releases of Designer (i.e. v.2.0 M5.1).**

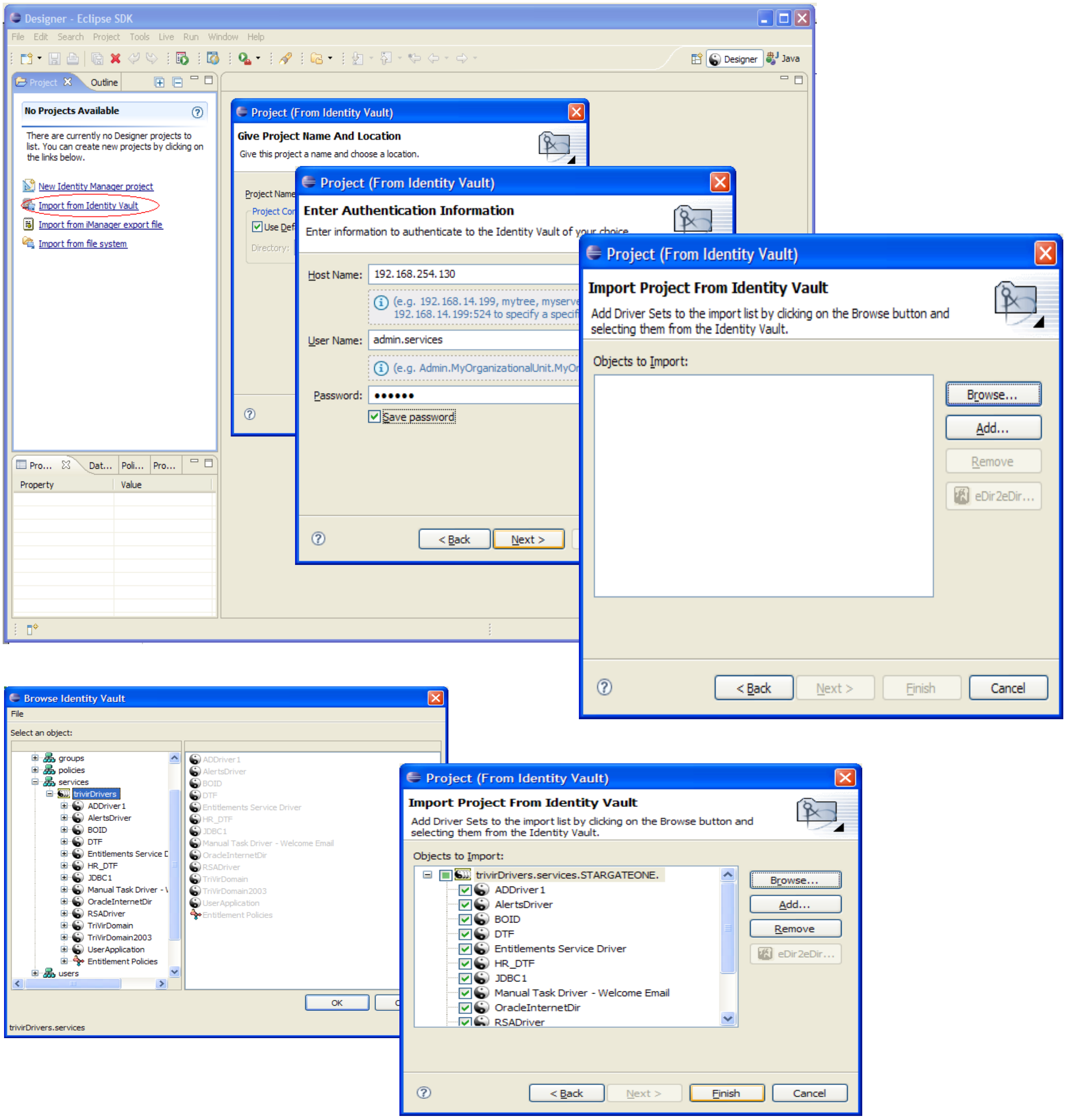
2. Extract the IdMUnit release into the desired location. Note that the myProfiles folder contains system connection information and encrypted credentials. Generally there is a separate IdmUnit folder for each user when working in a shared development environment. The myTests folder contains the actual test definition spreadsheets, and may be shared between users. It is recommended that these spreadsheets be managed in a version control system.

3. Setup the IdMUnit and Designer projects

Note that environment profile configuration for IdMUnit (i.e. IP addresses, credentials, etc.) is stored in the IdMUnit/myProfiles/idmunit-config.xml file. It is recommended that the following installation instructions are completed prior to modifying this file. Once installed and validated, IdMUnit may then be configured to talk to the local environment. Note that in addition to configuring the idmunit-config.xml file it will be necessary to update the example test spreadsheets to implement the appropriate test steps. See the IdMUnit usage guide for more details on this process.

3.1. Run Designer, go to the Workbench and import/create a Designer project (see the Designer documentation for assistance as necessary to complete this step)


Java - Eclipse SDK
File Edit Navigate Search Project Tools Live Run Window Help
Welcome
Workbench
Go to the workbench
eclipse



No Projects Available

There are currently no Designer projects to list. You can create new projects by clicking on the links below.

- New Identity Manager project
- Import from Identity Vault
- Import from iManager export file
- Import from file system

Project (From Identity Vault)

Give Project Name And Location

Give this project a name and choose a location.

Project (From Identity Vault)

Enter Authentication Information

Enter information to authenticate to the Identity Vault of your choice.

Host Name: 192.168.254.130

(e.g. 192.168.14.199, mytree, myserve
192.168.14.199:524 to specify a specif

User Name: admin.services

(e.g. Admin.MyOrganizationalUnit.MyOr

Password: ••••••

☑ Save password

Project (From Identity Vault)

Import Project From Identity Vault

Add Driver Sets to the import list by clicking on the Browse button and selecting them from the Identity Vault.

Objects to Import:

Browse Identity Vault

Select an object:

```
⊞ groups
⊞ policies
⊟ services
  ⊟ trivirDrivers
    ⊞ ADDriver1
    ⊞ AlertsDriver
    ⊞ BOID
    ⊞ DTF
    ⊞ Entitlements Service D
    ⊞ HR_DTF
    ⊞ JDBC1
    ⊞ Manual Task Driver - \
    ⊞ OracleInternetDir
    ⊞ RSADriver
    ⊞ TriVirDomain
    ⊞ TriVirDomain2003
    ⊞ UserApplication
    ⊞ Entitlement Policies
⊞ users
```

```
ADDriver1
AlertsDriver
BOID
DTF
Entitlements Service Driver
HR_DTF
JDBC1
Manual Task Driver - Welcome Email
OracleInternetDir
RSADriver
TriVirDomain
TriVirDomain2003
UserApplication
Entitlement Policies
```

trivirDrivers.services

Project (From Identity Vault)

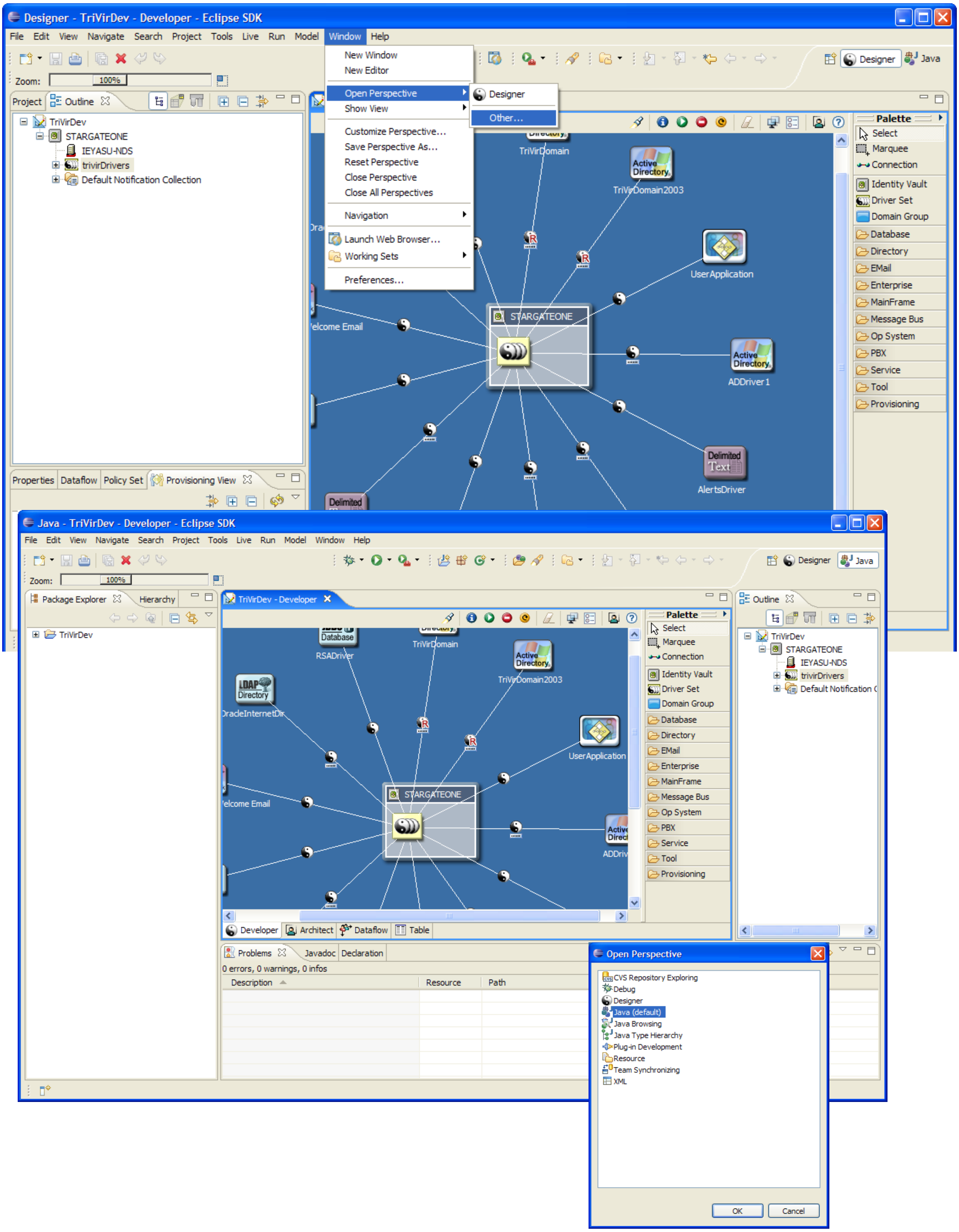
Import Project From Identity Vault

Add Driver Sets to the import list by clicking on the Browse button and selecting them from the Identity Vault.

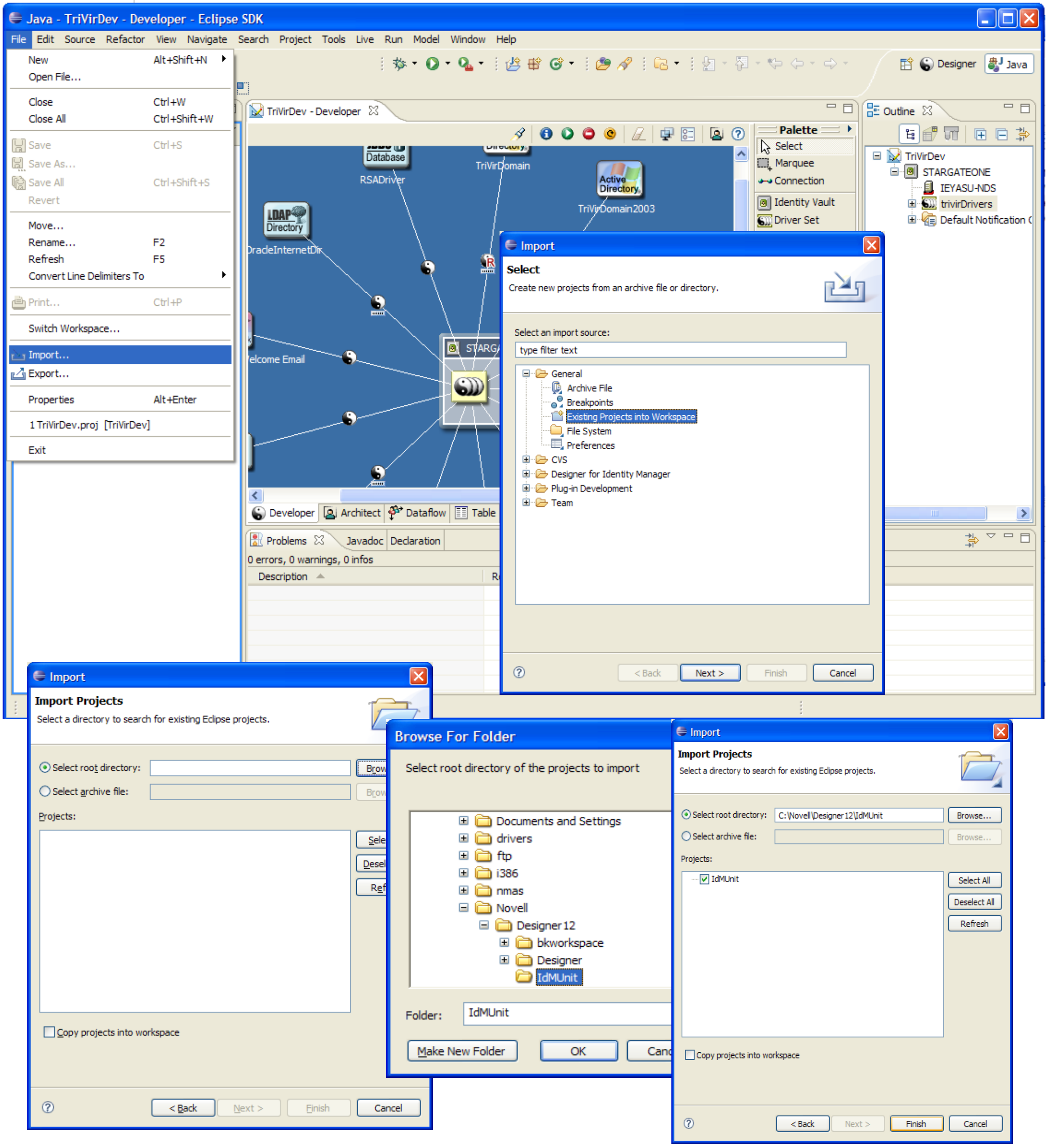
Objects to Import:

```
⊟ ☑ trivirDrivers.services.STARGATEONE.
    ☑ ADDriver1
    ☑ AlertsDriver
    ☑ BOID
    ☑ DTF
    ☑ Entitlements Service Driver
    ☑ HR_DTF
    ☑ JDBC1
    ☑ Manual Task Driver - Welcome Email
    ☑ OracleInternetDir
    ☑ RSADriver
```

3.2. Switch to the Eclipse Java perspective by clicking Window --> Open Perspective --> Other --> Java



4. Import the IdMUnit project by clicking File --> Import --> expand the General section --> select Existing Projects into Workspace --> browse to the IdMUnit folder and click Finish. IdMunit should be visible now in the Package Explorer on the left pane.



Java - TriVirDev - Developer - Eclipse SDK

0 errors, 0 warnings, 0 infos

Import

Select

Create new projects from an archive file or directory.

Select an import source:

type filter text

Import

Import Projects

Select a directory to search for existing Eclipse projects.

○ Select root directory:

○ Select archive file:

Projects:

☐ Copy projects into workspace

Browse For Folder

Select root directory of the projects to import

Folder: IdMUnit

Import

Import Projects

Select a directory to search for existing Eclipse projects.

● Select root directory: C:\Novell\Designer12\IdMUnit

○ Select archive file:

Projects:

☑ IdMUnit

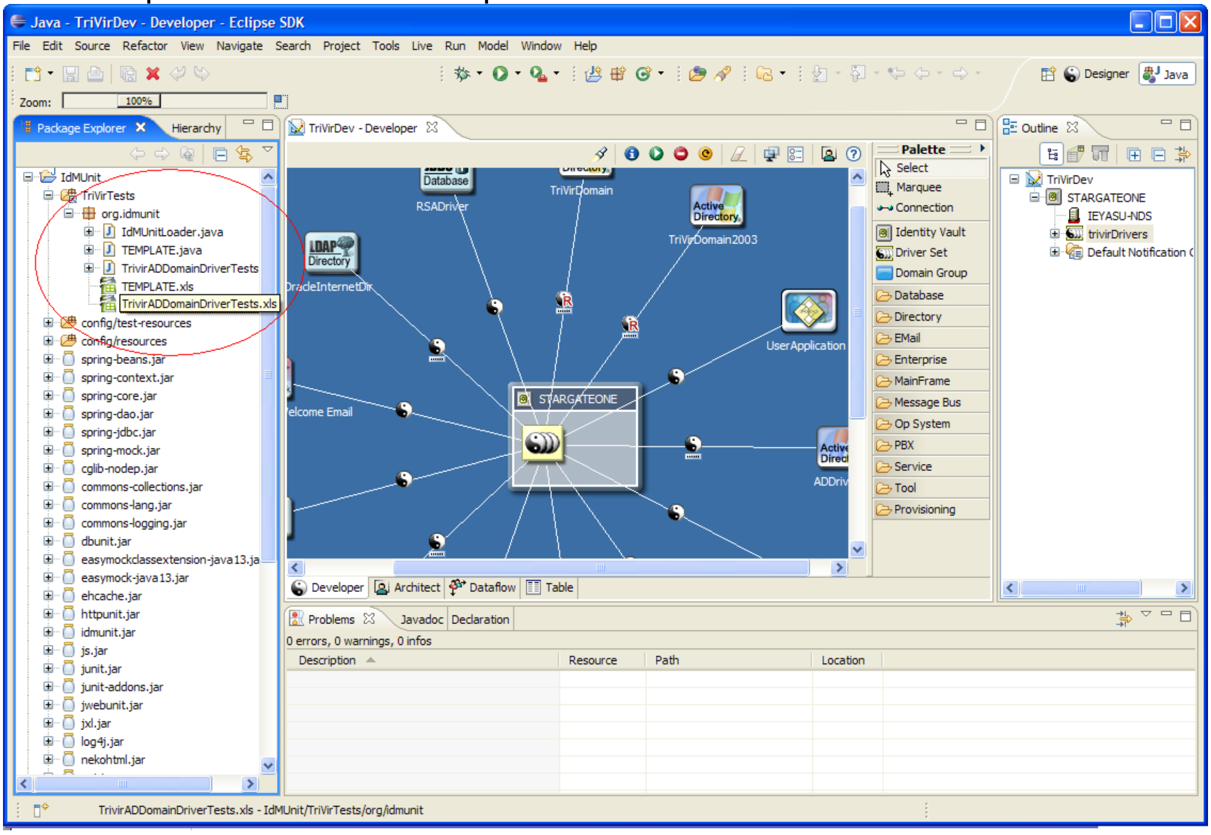
☐ Copy projects into workspace

Once the IdMUnit project has been imported, configure the IdMUnit project to link in the desired profile configuration and test spreadsheet location by performing the following:

Select the IdMUnit project in the Package Explorer View. Click Project --> Properties --> Java Build Path --> click the Source tab --> double-click the SampleProfiles folder and update the file system path to point to the folder that contains the desired idmunit-config.xml for this project --> click Finish. Also double-click the SampleTests folder and update the path to point to the location where test spreadsheets are located (this folder should point to the org folder in the path org/idmunit where spreadsheets are stored in the idmunit folder). Note that IdMUnit is supported in a multi-user environment (i.e. terminal server, etc.) when a separate IdMUnit folder exists for each user. This step #4 can be repeated for every user .

5.Basic Usage

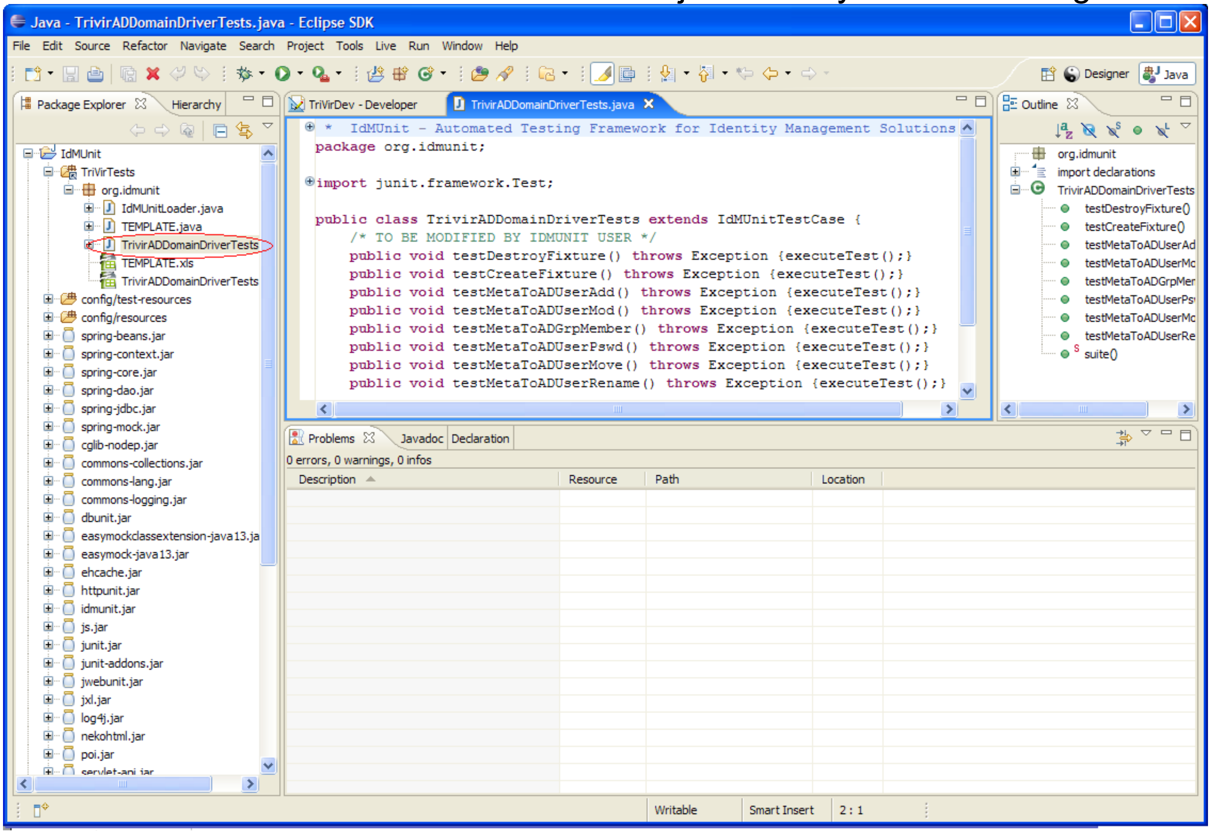
5.1. Expand the IdMUnit sample tests



0 errors, 0 warnings, 0 infos

TrivirADDomainDriverTests.xls - IdMUnit/TriVirTests/org/idmunit

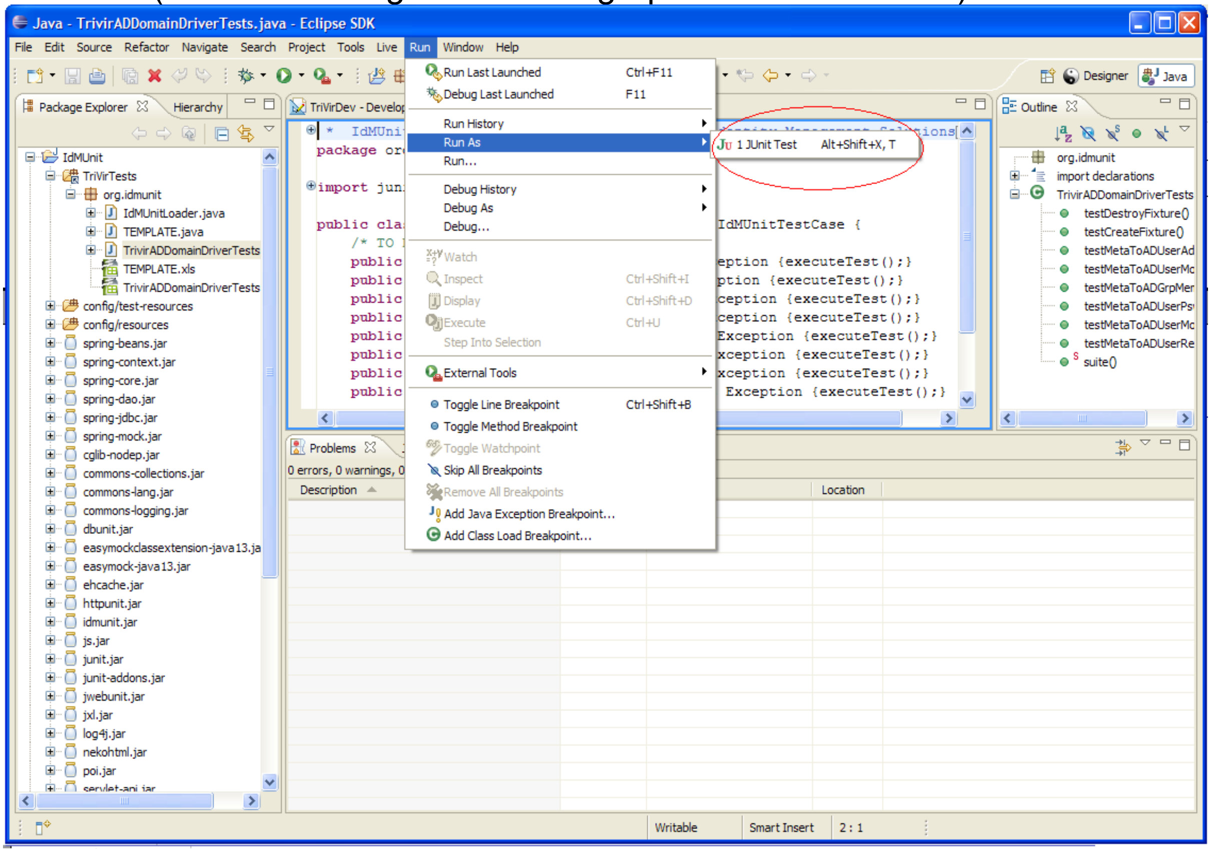
5.2. Load the TrivirADDomainDriverTests.java test by double-clicking



```
 *  IdMUnit - Automated Testing Framework for Identity Management Solutions
package org.idmunit;

import junit.framework.Test;

public class TrivirADDomainDriverTests extends IdMUnitTestCase {
    /* TO BE MODIFIED BY IDMUNIT USER */
    public void testDestroyFixture() throws Exception {executeTest();}
    public void testCreateFixture() throws Exception {executeTest();}
    public void testMetaToADUserAdd() throws Exception {executeTest();}
    public void testMetaToADUserMod() throws Exception {executeTest();}
    public void testMetaToADGrpMember() throws Exception {executeTest();}
    public void testMetaToADUserPswd() throws Exception {executeTest();}
    public void testMetaToADUserMove() throws Exception {executeTest();}
    public void testMetaToADUserRename() throws Exception {executeTest();}
```

0 errors, 0 warnings, 0 infos

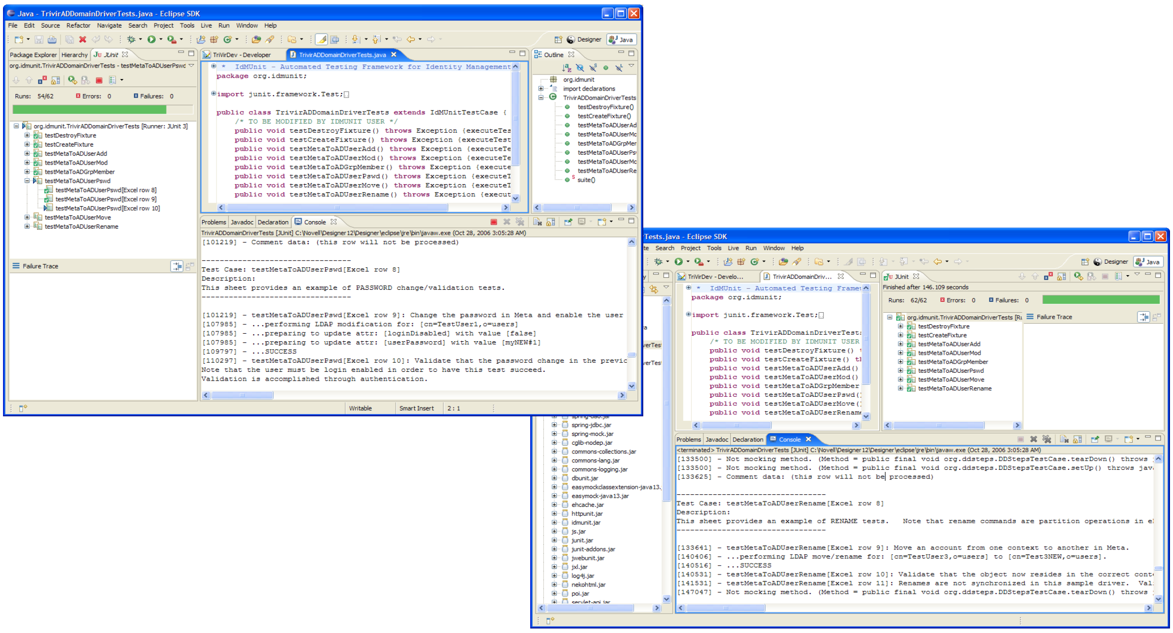
5.3. Execute the TrivirADDomainDriverTests by running the test as a JUnit test (IdMUnit leverages the JUnit graphical user interface)



Highlight the TriVirADDomainDriverTests and click Run --> Run-As --> JUnit test

Note that firewall software such as Norton or McAfee might complain about the Java sub-system trying to listen on a network interface on port 4296. It is not necessary to allow this rule in the firewall. However, the firewall might need to be opened if remote debugging is desired.

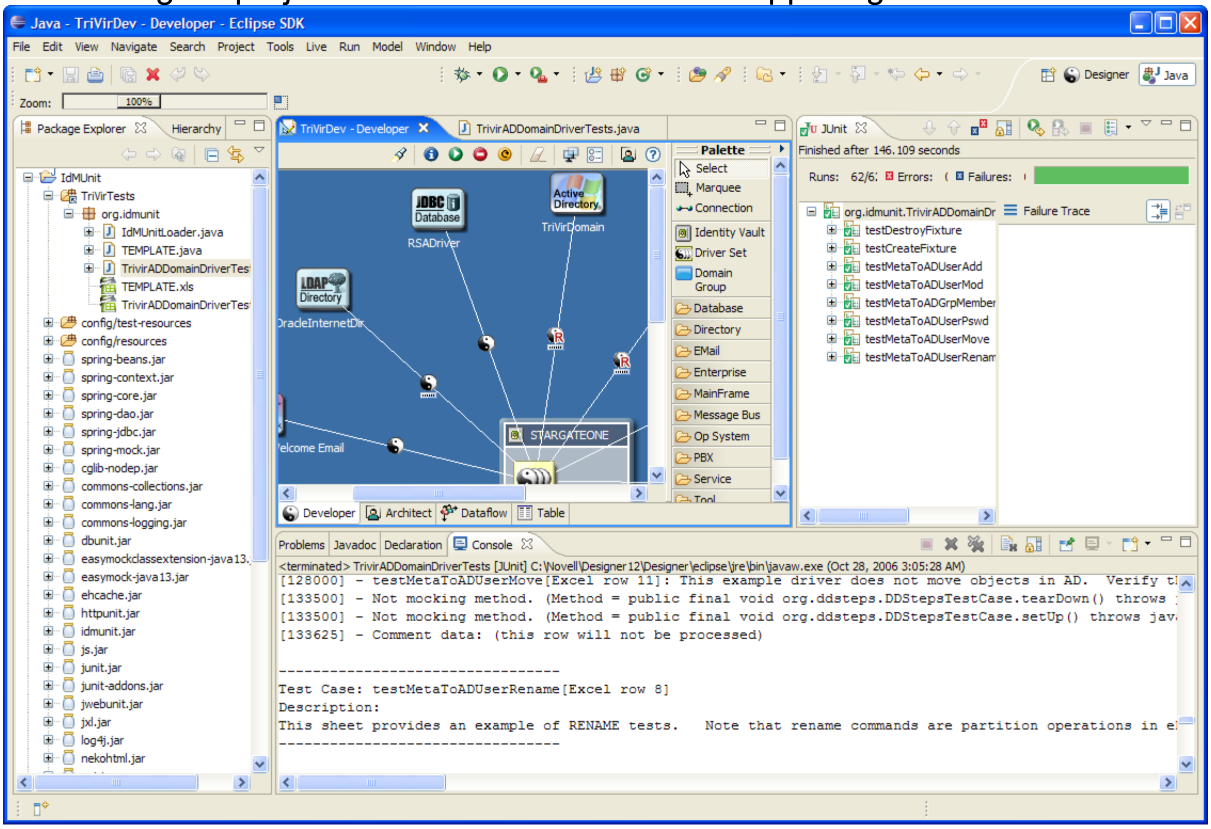
5.4. As the IdMUnit tests for the selected test spreadsheet launch, the layout and placement of the test runner may be configured by dragging and dropping the Eclipse tabs



```
TrivirADDomainDriverTests [JUnit] C:\Novell\Designer12\Designer\eclipse\jre\bin\javaw.exe (Oct 28, 2006 3:05:28 AM)
[101219] - Comment data: (this row will not be processed)

-------------------------------
Test Case: testMetaToADUserPswd[Excel row 8]
Description:
This sheet provides an example of PASSWORD change/validation tests.
-------------------------------

[101219] - testMetaToADUserPswd[Excel row 9]: Change the password in Meta and enable the user
[107985] - ...performing LDAP modification for: [cn=TestUser1,o=users]
[107985] - ...preparing to update attr: [loginDisabled] with value [false]
[107985] - ...preparing to update attr: [userPassword] with value [myNEW#1]
[109797] - ...SUCCESS
[110297] - testMetaToADUserPswd[Excel row 10]: Validate that the password change in the previc
Note that the user must be login enabled in order to have this test succeed.
Validation is accomplished through authentication.
```

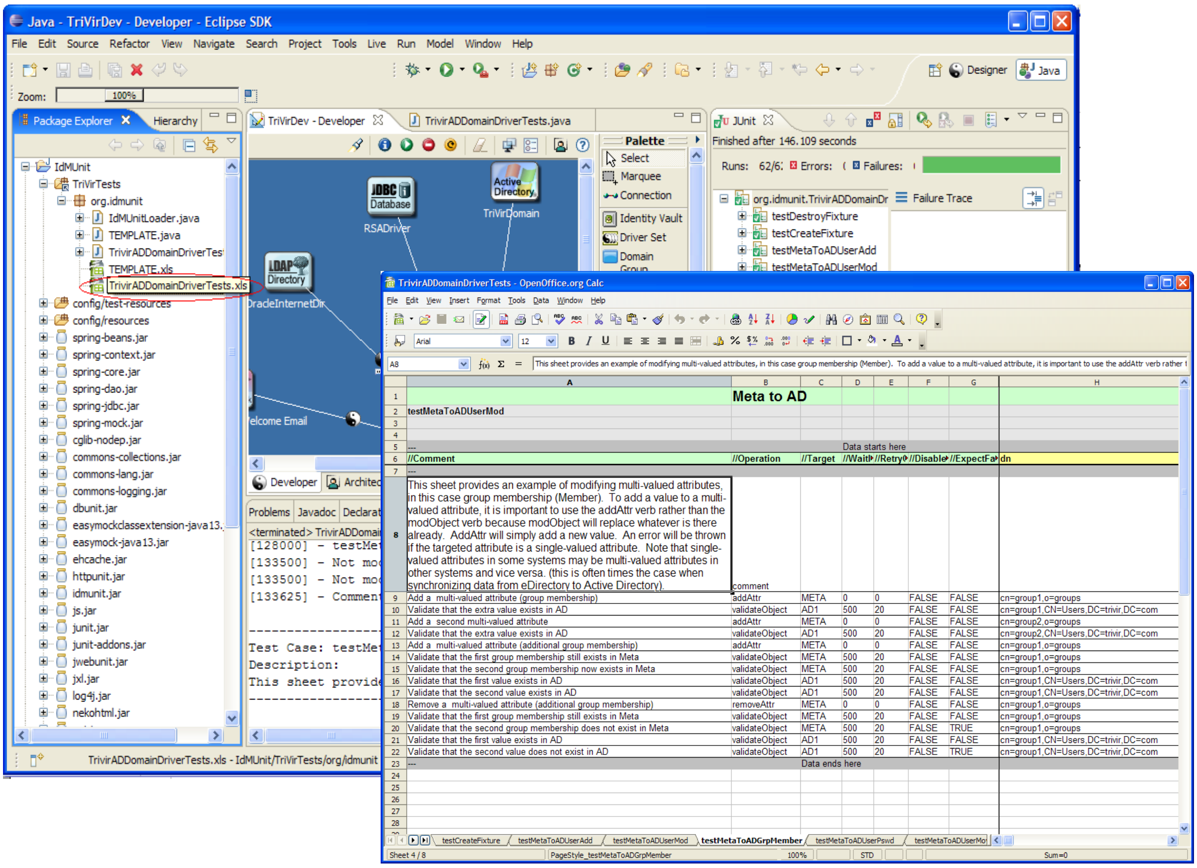
```
<terminated> TrivirADDomainDriverTests [JUnit] C:\Novell\Designer12\Designer\eclipse\jre\bin\javaw.exe (Oct 28, 2006 3:05:28 AM)
[133500] - Not mocking method. (Method = public final void org.ddsteps.DDStepsTestCase.tearDown() throws
[133500] - Not mocking method. (Method = public final void org.ddsteps.DDStepsTestCase.setUp() throws jav
[133625] - Comment data: (this row will not be processed)

-------------------------------
Test Case: testMetaToADUserRename[Excel row 8]
Description:
This sheet provides an example of RENAME tests.   Note that rename commands are partition operations in e
-------------------------------

[133641] - testMetaToADUserRename[Excel row 9]: Move an account from one context to another in Meta.
[140406] - ...performing LDAP move/rename for: [cn=TestUser3,o=users] to [cn=Test3NEW,o=users].
[140516] - ...SUCCESS
[140531] - testMetaToADUserRename[Excel row 10]: Validate that the object now resides in the correct conte
[141531] - testMetaToADUserRename[Excel row 11]: Renames are not synchronized in this sample driver.  Val
[147047] - Not mocking method. (Method = public final void org.ddsteps.DDStepsTestCase.tearDown() throws
```

5.5. The Designer perspective and Java perspective may both be activated simultaneously and navigated by simply by clicking between the Designer project tab and the test tab in the upper right corner



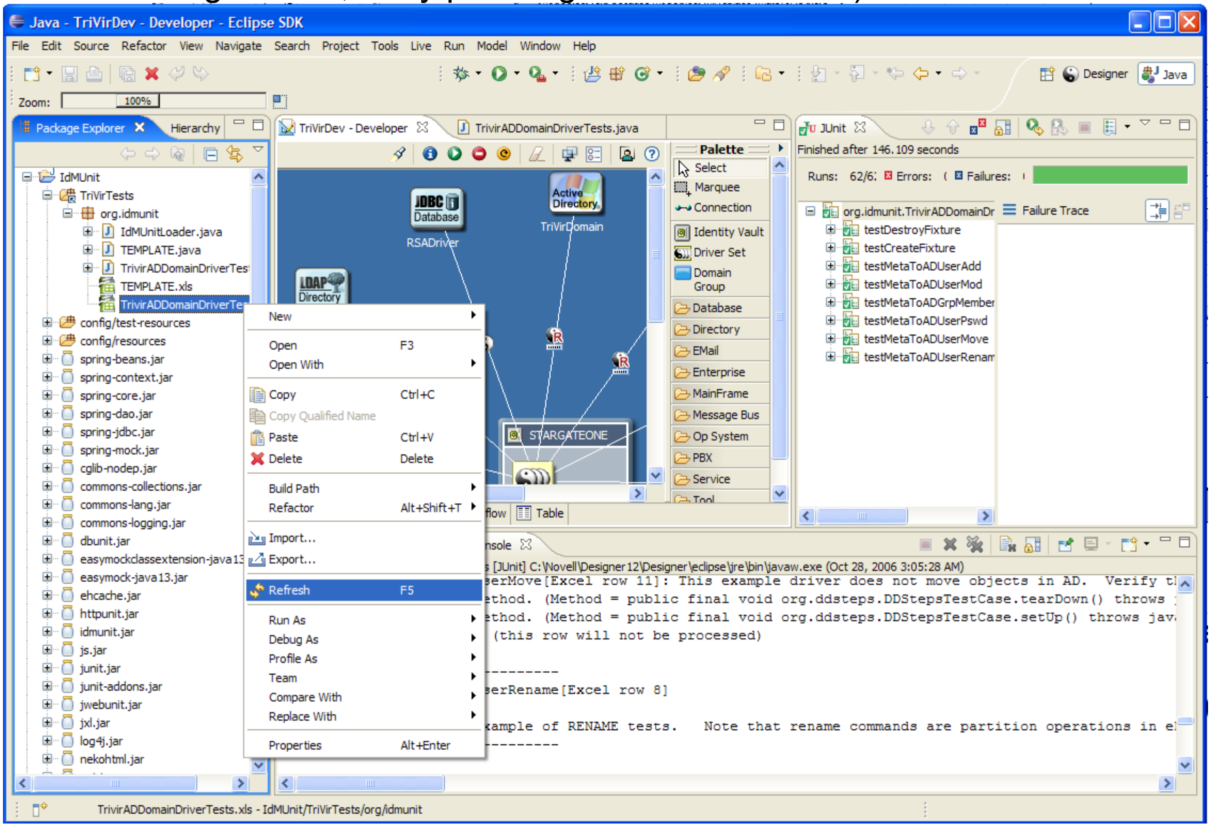
```
<terminated> TrivirADDomainDriverTests [JUnit] C:\Novell\Designer 12\Designer\eclipse\jre\bin\javaw.exe (Oct 28, 2006 3:05:28 AM)
[128000] - testMetaToADUserMove[Excel row 11]: This example driver does not move objects in AD.  Verify tl
[133500] - Not mocking method. (Method = public final void org.ddsteps.DDStepsTestCase.tearDown() throws
[133500] - Not mocking method. (Method = public final void org.ddsteps.DDStepsTestCase.setUp() throws jav
[133625] - Comment data: (this row will not be processed)

--------------------------------
Test Case: testMetaToADUserRename[Excel row 8]
Description:
This sheet provides an example of RENAME tests.   Note that rename commands are partition operations in e
--------------------------------
```

5.6. Test spreadsheets may be accessed and modified by double-clicking within the Eclipse Package Explorer view (Use OpenOffice or MS Excel for editing). Note that the first time a spreadsheet is opened from Designer, right-click and open with "System Editor".


Java - TriVirDev - Developer - Eclipse SDK
File Edit Source Refactor View Navigate Search Project Tools Live Run Model Window Help
Package Explorer
TrivirADDomainDriverTests.xls
TrivirADDomainDriverTests.xls - IdMUnit/TriVirTests/org/idmunit
TrivirADDomainDriverTests - OpenOffice.org Calc
File Edit View Insert Format Tools Data Window Help
Arial
12
A8
testCreateFixture
testMetaToADUserAdd
testMetaToADUserMod
testMetaToADGrpMember
testMetaToADUserPswd
testMetaToADUserMo
Sheet 4 / 8
PageStyle_testMetaToADGrpMember
100%
STD
Sum=0

	A	B	C	D	E	F	G	H
1		Meta to AD						
2	testMetaToADUserMod							
3								
4								
5	---			Data starts here				
6	//Comment	//Operation	//Target	//Waitl	//Retry	//Disable	//ExpectFa	dn
7	---							
8	This sheet provides an example of modifying multi-valued attributes, in this case group membership (Member). To add a value to a multi-valued attribute, it is important to use the addAttr verb rather than the modObject verb because modObject will replace whatever is there already. AddAttr will simply add a new value. An error will be thrown if the targeted attribute is a single-valued attribute. Note that single-valued attributes in some systems may be multi-valued attributes in other systems and vice versa. (this is often times the case when synchronizing data from eDirectory to Active Directory).	comment						
9	Add a multi-valued attribute (group membership)	addAttr	META	0	0	FALSE	FALSE	cn=group1,o=groups
10	Validate that the extra value exists in AD	validateObject	AD1	500	20	FALSE	FALSE	cn=group1,CN=Users,DC=trivir,DC=com
11	Add a second multi-valued attribute	addAttr	META	0	0	FALSE	FALSE	cn=group2,o=groups
12	Validate that the extra value exists in AD	validateObject	AD1	500	20	FALSE	FALSE	cn=group2,CN=Users,DC=trivir,DC=com
13	Add a multi-valued attribute (additional group membership)	addAttr	META	0	0	FALSE	FALSE	cn=group1,o=groups
14	Validate that the first group membership still exists in Meta	validateObject	META	500	20	FALSE	FALSE	cn=group1,o=groups
15	Validate that the second group membership now exists in Meta	validateObject	META	500	20	FALSE	FALSE	cn=group1,o=groups
16	Validate that the first value exists in AD	validateObject	AD1	500	20	FALSE	FALSE	cn=group1,CN=Users,DC=trivir,DC=com
17	Validate that the second value exists in AD	validateObject	AD1	500	20	FALSE	FALSE	cn=group1,CN=Users,DC=trivir,DC=com
18	Remove a multi-valued attribute (additional group membership)	removeAttr	META	0	0	FALSE	FALSE	cn=group1,o=groups
19	Validate that the first group membership still exists in Meta	validateObject	META	500	20	FALSE	FALSE	cn=group1,o=groups
20	Validate that the second group membership does not exist in Meta	validateObject	META	500	20	FALSE	TRUE	cn=group1,o=groups
21	Validate that the first value exists in AD	validateObject	AD1	500	20	FALSE	FALSE	cn=group1,CN=Users,DC=trivir,DC=com
22	Validate that the second value does not exist in AD	validateObject	AD1	500	20	FALSE	TRUE	cn=group1,CN=Users,DC=trivir,DC=com
23	---		Data ends here					

5.7. After saving a change in a test spreadsheet, it must be refreshed in the Eclipse project before it will go live (may refresh by right-clicking and selecting refresh, or by pressing F5 while selected)



```
[JUnit] C:\Novell\Designer12\Designer\eclipse\jre\bin\javaw.exe (Oct 28, 2006 3:05:28 AM)
erMove[Excel row 11]: This example driver does not move objects in AD.  Verify tl
ethod. (Method = public final void org.ddsteps.DDStepsTestCase.tearDown() throws
ethod. (Method = public final void org.ddsteps.DDStepsTestCase.setUp() throws jav
 (this row will not be processed)

---------
erRename[Excel row 8]

xample of RENAME tests.   Note that rename commands are partition operations in e
---------
```

A new test spreadsheet may be created by navigating on the file system to IdMUnit/myTests/org/idmunit and copying/pasting the test spreadsheet xls file and corresponding .java file. After copying and pasting, right-click the IdMUnit project in the Package Explorer view of Designer and select refresh. The .java class will need to be updated to contain the correct class name in the top of the class, and in the bottom test method. A simple search and replace with Ctrl-F can perform this function.

Appendix A – IdMUnit Profiles

1. Profile Configuration

Profiles contain a collection of connection information required for target systems. A single profile represents all of the necessary information required to talk to Identity Manager-connected systems within a single environment, like the development lab, test lab, or production environment. For additional details on the layout of a profile, see idmunit.dtd.

Sample profile excerpt:

```
<profile name="trivirLab">
     <connection>
        <name>META</name>
        <description>Connector for the identity vault</description>
        <type>org.idmunit.connector.LDAP</type>
        <server>172.19.18.132</server>
        <user>cn=admin,o=services</user>
        <password>B2vPD2UsfKc=</password>
        <keystore-path/>
```

2. Profile Selection

The “live-profile” specified at the top of the idmunit-config.xml file will be the profile used by IdMUnit for test execution. For example:

```
<idmunit live-profile="trivirLab" enable-email-alerts="true" enable-log-alerts="true">
```

This configuration would execute tests against the connections specified in the “trivirLab” profile.

3. Profile Use in a Multi-User Environment

The classpath of the project will determine which idmunit-config.xml file is applied. The Designer/Eclipse project properties link in the directory that contains the idmunit-config.xml file. This may be modified, per project to point to a separate directory. If multiple users are leveraging separate Designer workspaces on the same machine, they can each link in their own profiles folder so that tests can be run concurrently against different profiles. Note also that the build.xml file may be updated to point to any path for the profiles directory (this is the command-line Ant runner for IdMUnit).

4. Encrypt a password for use in the IdMUnit profile configuration

- Open the EncTool class in the Package Explorer
  - Expand the idmunit-core folder in the Package Explorer --> expand the org.idmunit package --> right-click EncTool and select Run-AS --> then select the green Run button
  - Select Java application in the left pane of the new pop-up window
  - Click the “new” icon in the top left-hand corner and a configuration layout will appear to the right

  - Select the arguments tab
  - In the Program Arguments field, type the following where mypassword is the actual password of the system

    ```
    IDMUNIT1 mypassword
    ```

  - Click Run at the bottom and the password will be encrypted with DES and base-64 encoded.
  - Copy the generated password to the idmunit-profiles.xml for the target system
- Note that the key IDMUNIT1 may be changed, as long as the bytes for the key are specified in the top of the EncTool source in the “iv” class member variable:

  ```
  private final byte[] iv = { 0x0a, 0x01, 0x02, 0x03, 0x04, 0x0b, 0x0c, 0x0d };
  ```

5. Import a certificate into IdMUnit for SSL communication via the LDAP connector

- Export a DER encoded trusted-root certificate (from the target AD DC, eDirectory or other LDAP server)
- Copy the DER encoded certificate onto the machine where IdMUnit is running. It may be copied into IdMUnit/util/security where a batch file resides that will help with importing the certificate into a Java certificate store
- Edit the IdMUnit/util/security/importCert.bat file to import the certificate into a known keystore as follows where MYCERT.cer is the name of the exported certificate file and the MYSSLKEY is the alias name of this certificate in the keystore:

  SAMPLE importCert.bat:

  ```
  java sun.security.tools.KeyTool -import -file MYCERT.cer -keystore keystore -alias "type=r.name=MYSSLKEY"
  ```

When executed, the keytool will ask for a password. Any password may be assigned to the keystore (remember this password in order to add additional certificates later on). The certificate file name and alias name may be parameterized with %1 and %2.

Questions/Comments?

Refer to www.idmunit.org or http://sourceforge.net/projects/idmunit for more information or email info@trivir.com for training, support, consulting and mentoring services.

TriVir