

SAPLink for ABAP Development Tools



Installation Guide

Version 2.0

March 08, 2013

SAP® Mentor

Uwe Fetzner

<https://cw.sdn.sap.com/cw/groups/saplink-for-abap-in-eclipse>

Contents

1	Requirements	3
2	Installation	3
2.1	Installing the frontend components	3
2.2	Installing the backend components	6
3	Import / Export Code into / from your ABAP system	8
3.1	Create a new SVN project in the Eclipse Project Explorer	8
3.2	Import source code into the ABAP system	12
3.3	Export source from the ABAP system	14

1 Requirements

The following software must be installed and properly configured before the installation of SAPlink for ADT can be started:

- Netweaver ABAP Stack 7 Enhancement Package 3 (NW ABAP 7.03), for example the ABAP trial version available under <http://scn.sap.com/community/developer-center/abap> or Netweaver ABAP Stack 7.3 Enhancement Package 1 (NW ABAP 7.31)
- Eclipse Indigo 3.7 or Juno 4.2 (<http://www.eclipse.org>)
- ABAP Development Tools for SAP Netweaver (aka ABAP in Eclipse), see <http://scn.sap.com/community/abap/eclipse>

In addition the SAP Code Exchange namespace “/CEX/” must be created via transaction SE03 (see wiki <http://wiki.sdn.sap.com/wiki/pages/viewpage.action?pageId=153780645>)

2 Installation

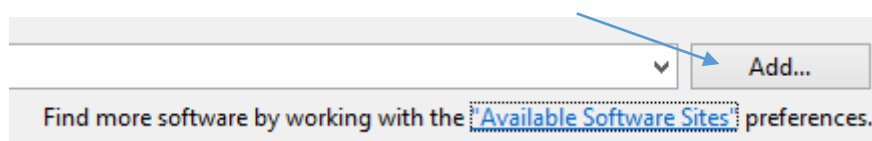
As an example the installation of SAPlink for ADT will be shown on a Windows desktop with Eclipse Juno 4.2. The tasks on other operating systems and Eclipse versions may vary a bit.

2.1 Installing the frontend components

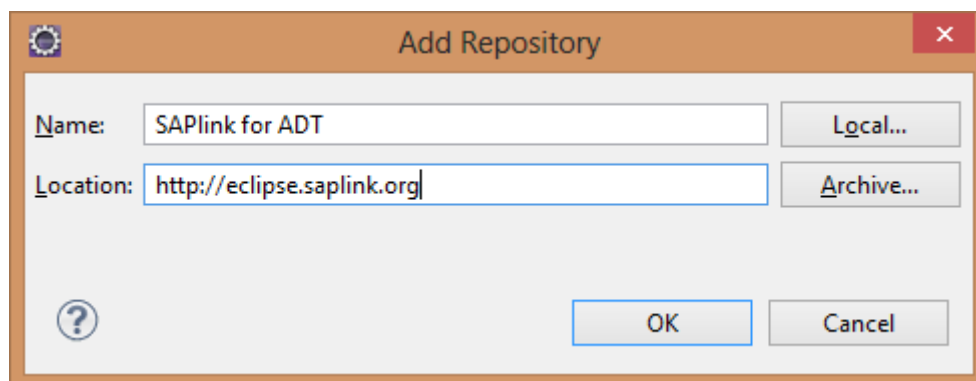
Start Eclipse

Menu: Help -> Install New Software ...

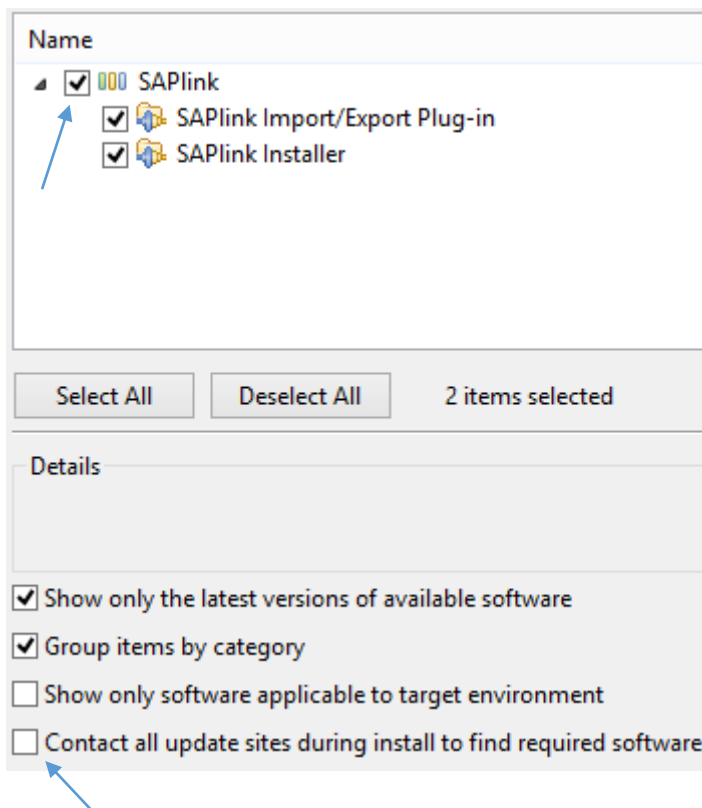
Add new Software Site



Enter the repository for “SAPlink for ADT” (eclipse.saplink.org)



Mark the checkbox “SAPlink” and uncheck “Contact all update sites...” and press “Next”



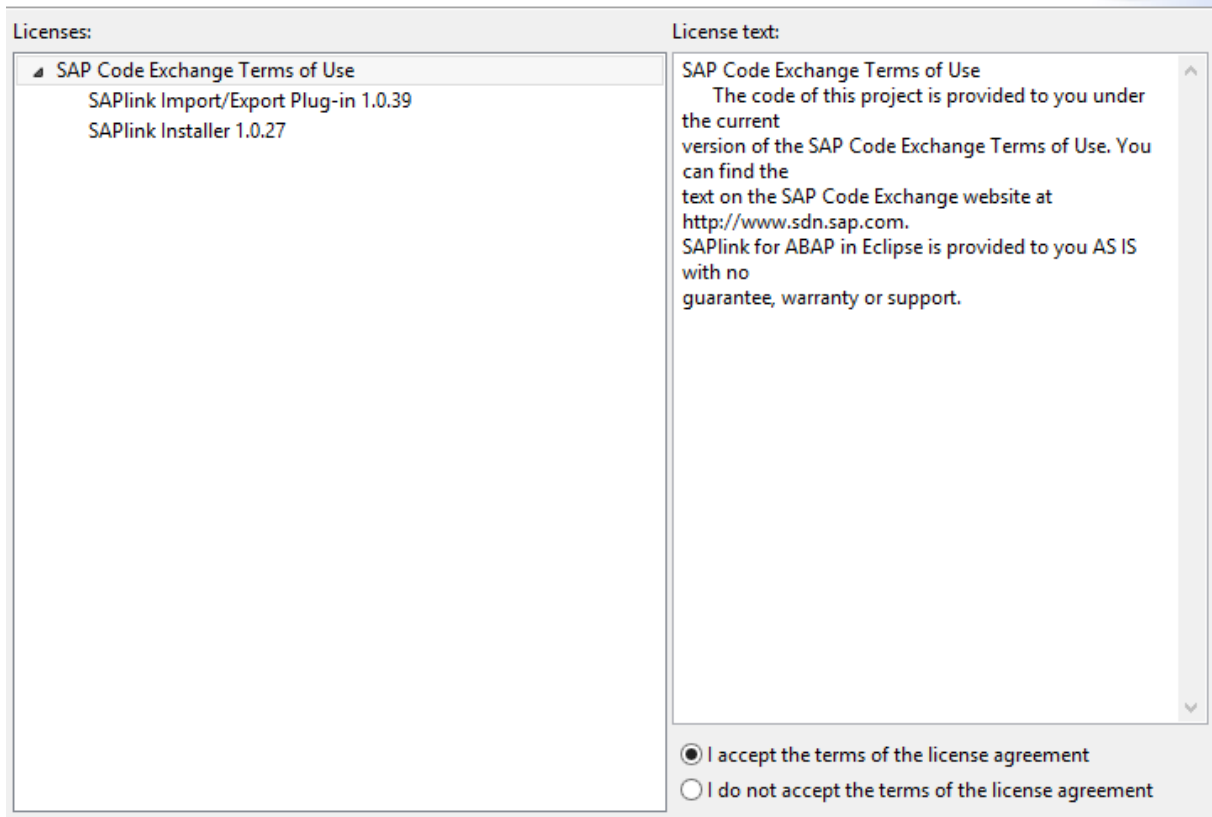
On the “Installation Details” screen press “Next”

Name	Version	Id
SAPlink Import/Export Plug-in	1.0.39	org.saplink.feature.group
SAPlink Installer	1.0.27	org.saplink.install.feature.group

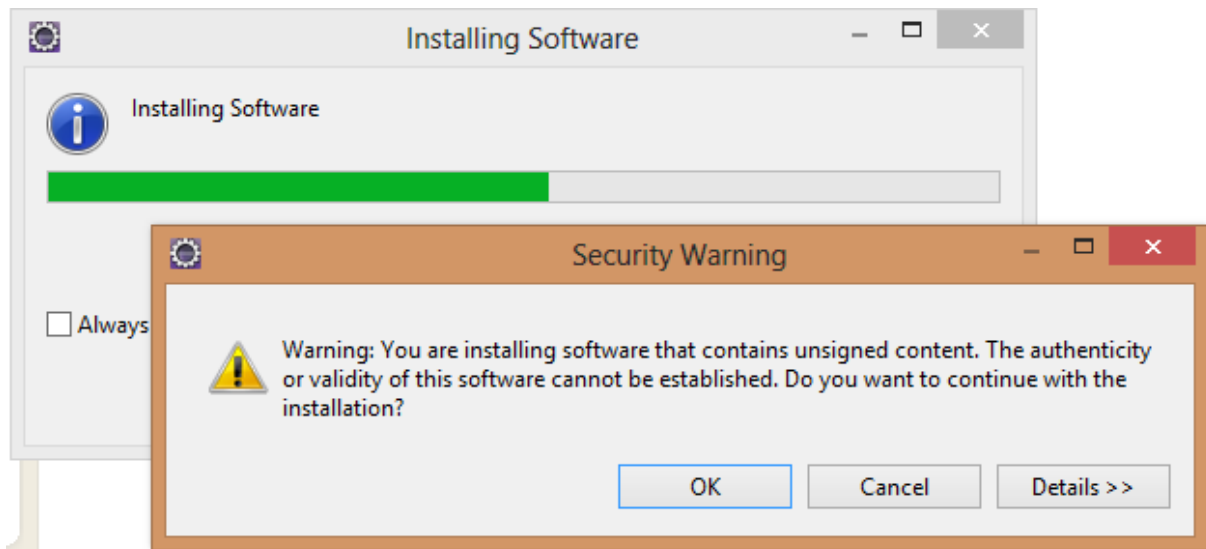
Accept the license agreement and press “Finish”

Review Licenses

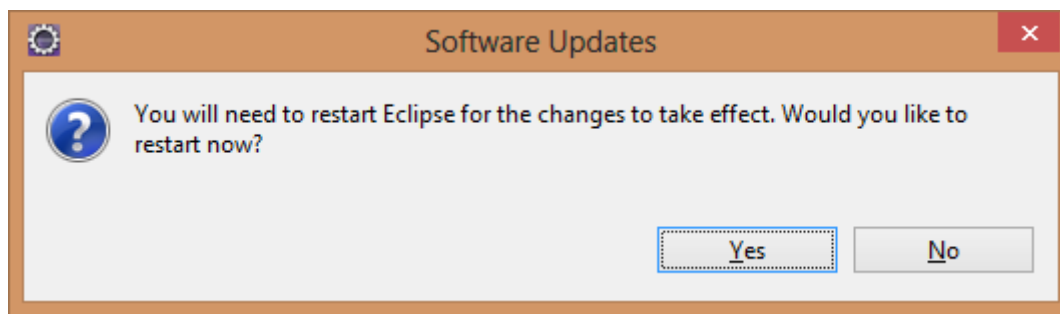
Licenses must be reviewed and accepted before the software can be installed.



Confirm the security warning (unsigned content)

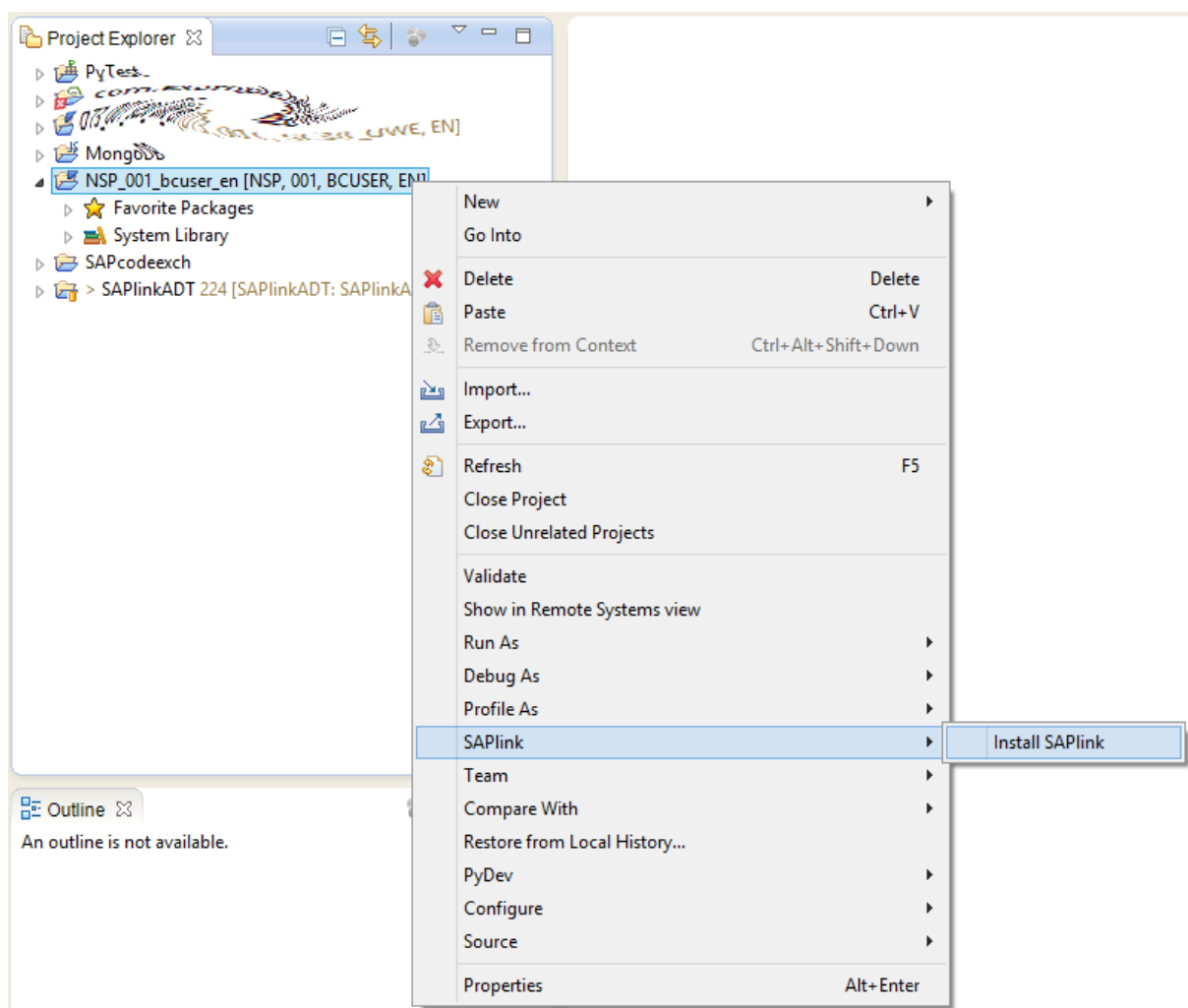


Restart Eclipse

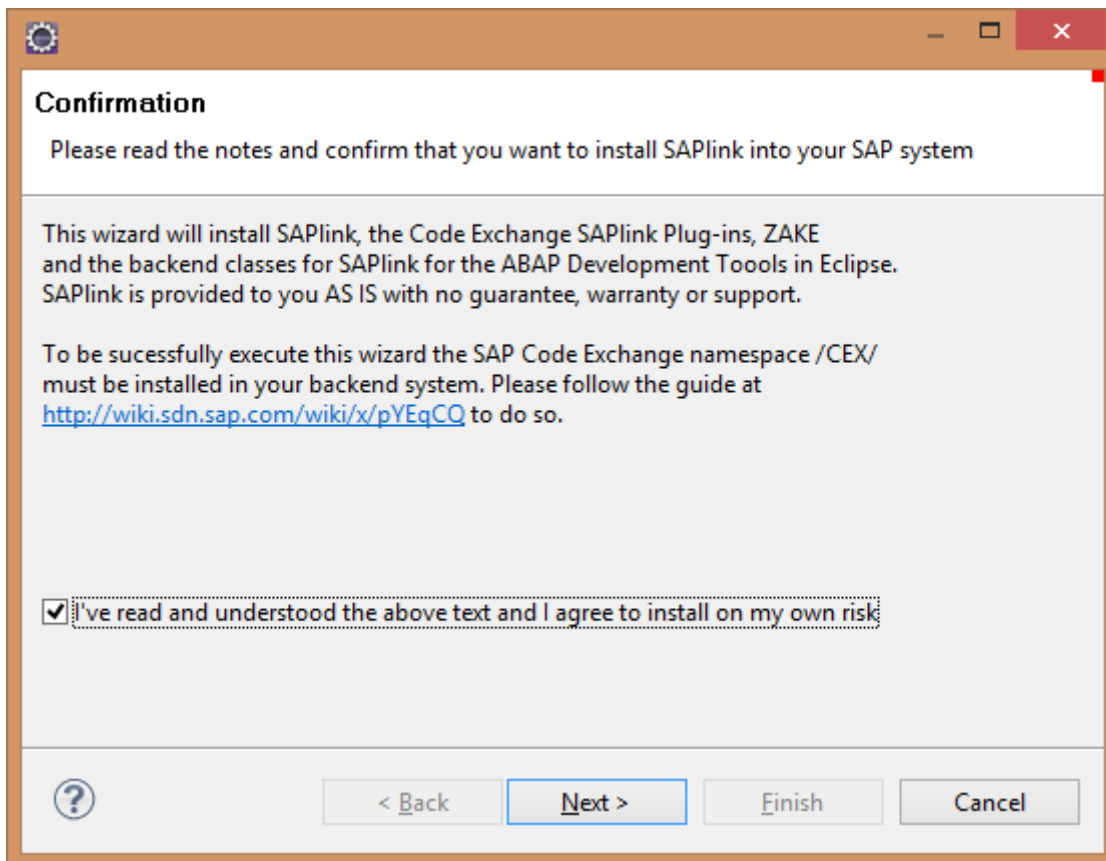


2.2 Installing the backend components

Right click on the system (Project) on which you want to install the SAPlink for ADT backend components and select "SAPlink -> Install SAPlink".

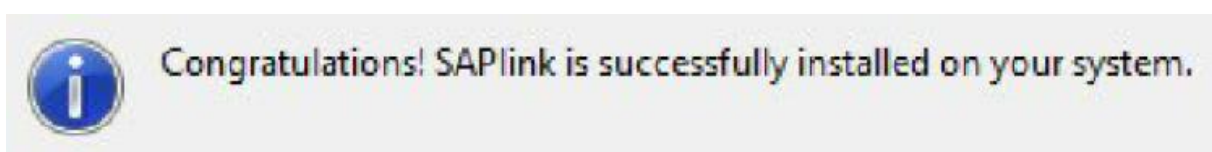
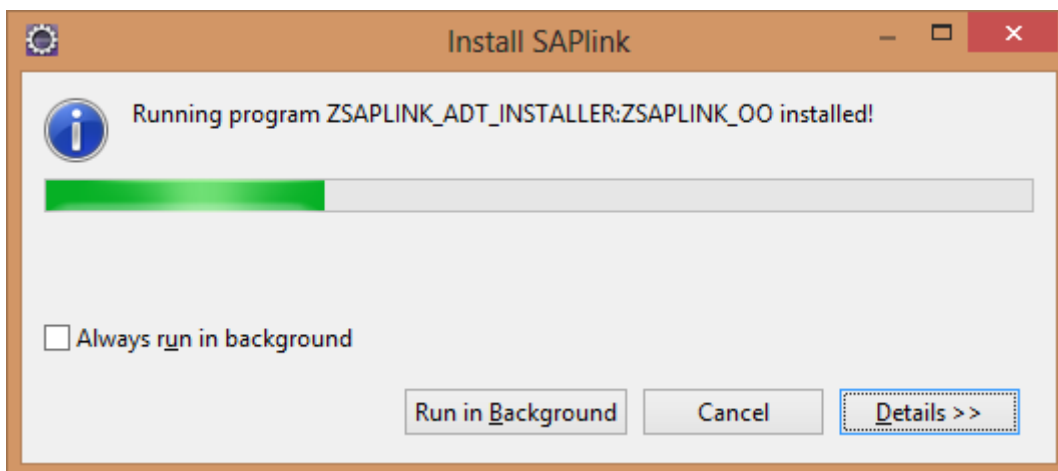


Please read, understand and accept the upcoming text. Press “Next”.



If you are not logged on already on the target system, you will be asked to enter your password.

The installation wizard is now ready to install the software on the backend system. Press “Finish”.



3 Import / Export Code into / from your ABAP system

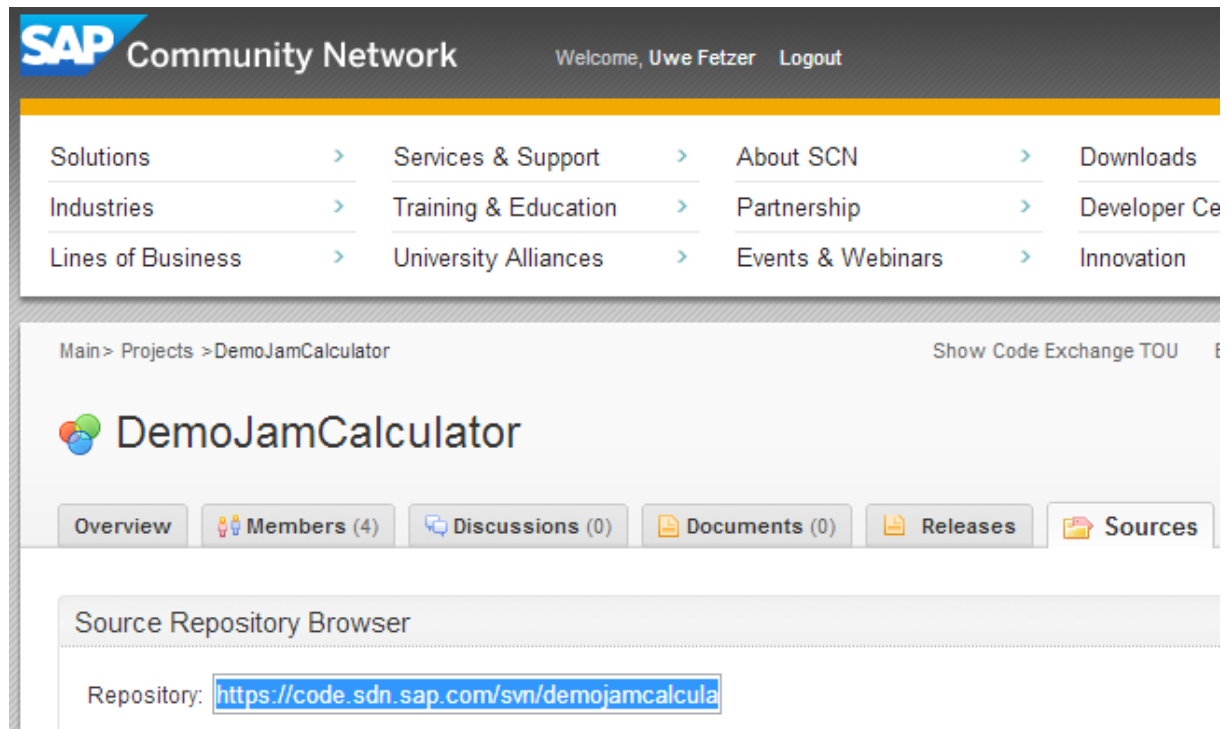
In this chapter we demonstrate how to handle code from external sources. As an example we use an SAP Code Exchange project.

Note: if you are using other SVN or GIT Eclipse plugins, the procedure for updating the repository may vary, but the import and export of objects via SAPLink for ADT will stay the same.

3.1 Create a new SVN project in the Eclipse Project Explorer

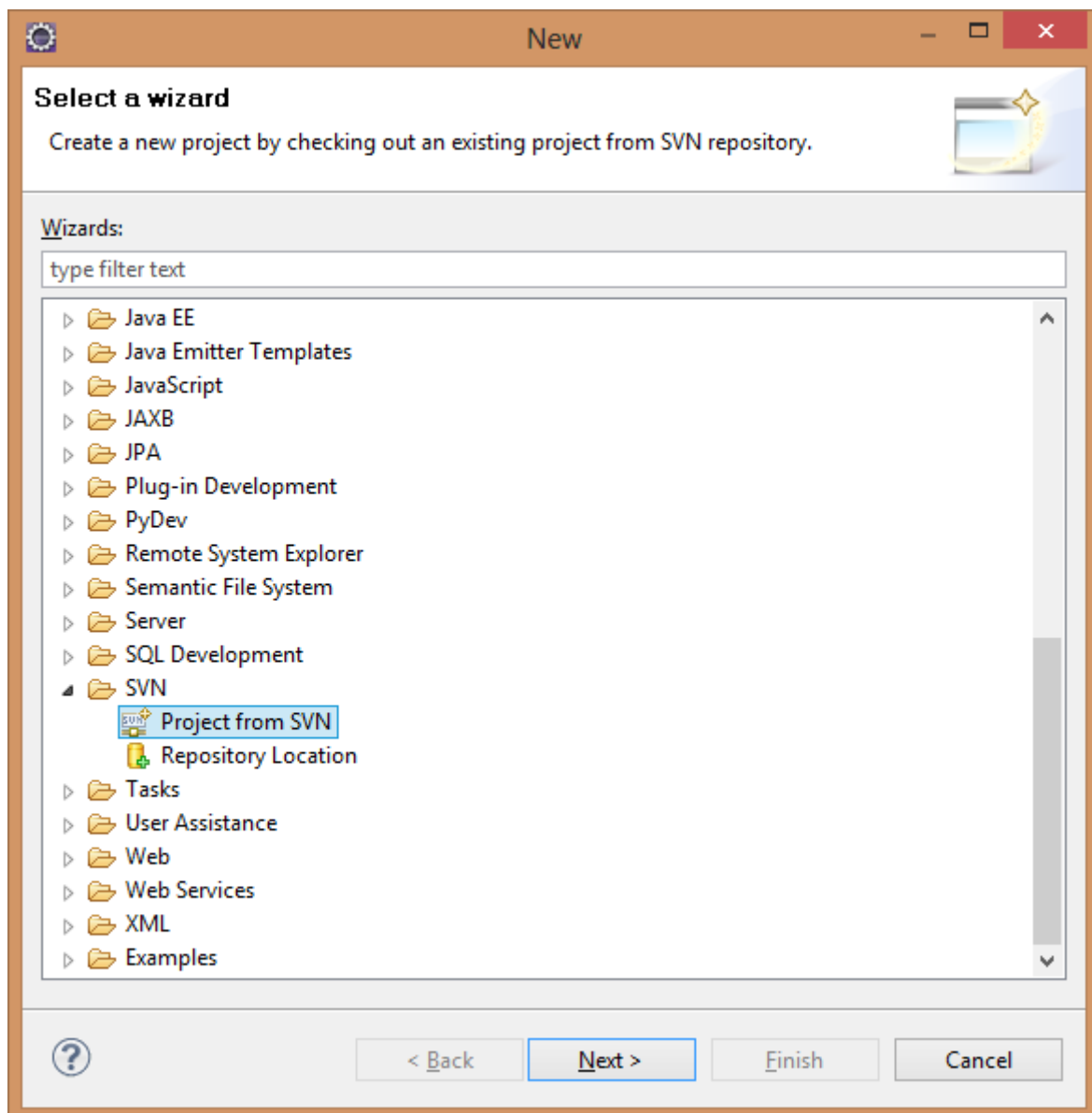
First we need to know the URL of the Code Repository of our SAP Code Exchange project.

Go to the SAP Code Exchange project home, tab “Sources”

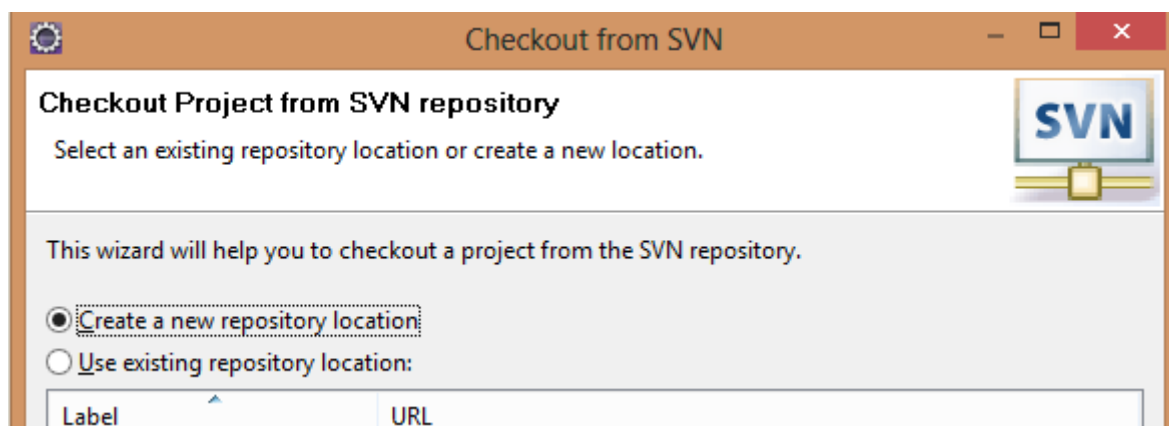


Mark and copy the URL.

In Eclipse go to “File -> New -> Other”, Select the wizard “SVN -> Project from SVN” and press “Next”



Create a new repository location



Paste the URL, enter your Assembla Credentials (NOT your SAP Code Exchange user/password) and press "Next".

Note: if you don't have an Assembla user/password yet, you can always create one, see <http://scn.sap.com/community/code-exchange/blog/2010/10/13/code-exchange-how-to-accessing-the-code-repository-of-a-project>

Checkout from SVN

Enter Repository Location Information

Define the SVN repository location information. You can specify additional settings for proxy and svn+ssh, https connections.

General | Advanced | SSH Settings | SSL Settings

URL:

Label

☐ Use the repository URL as the label

☒ Use a custom label:

Authentication

User:

Password:

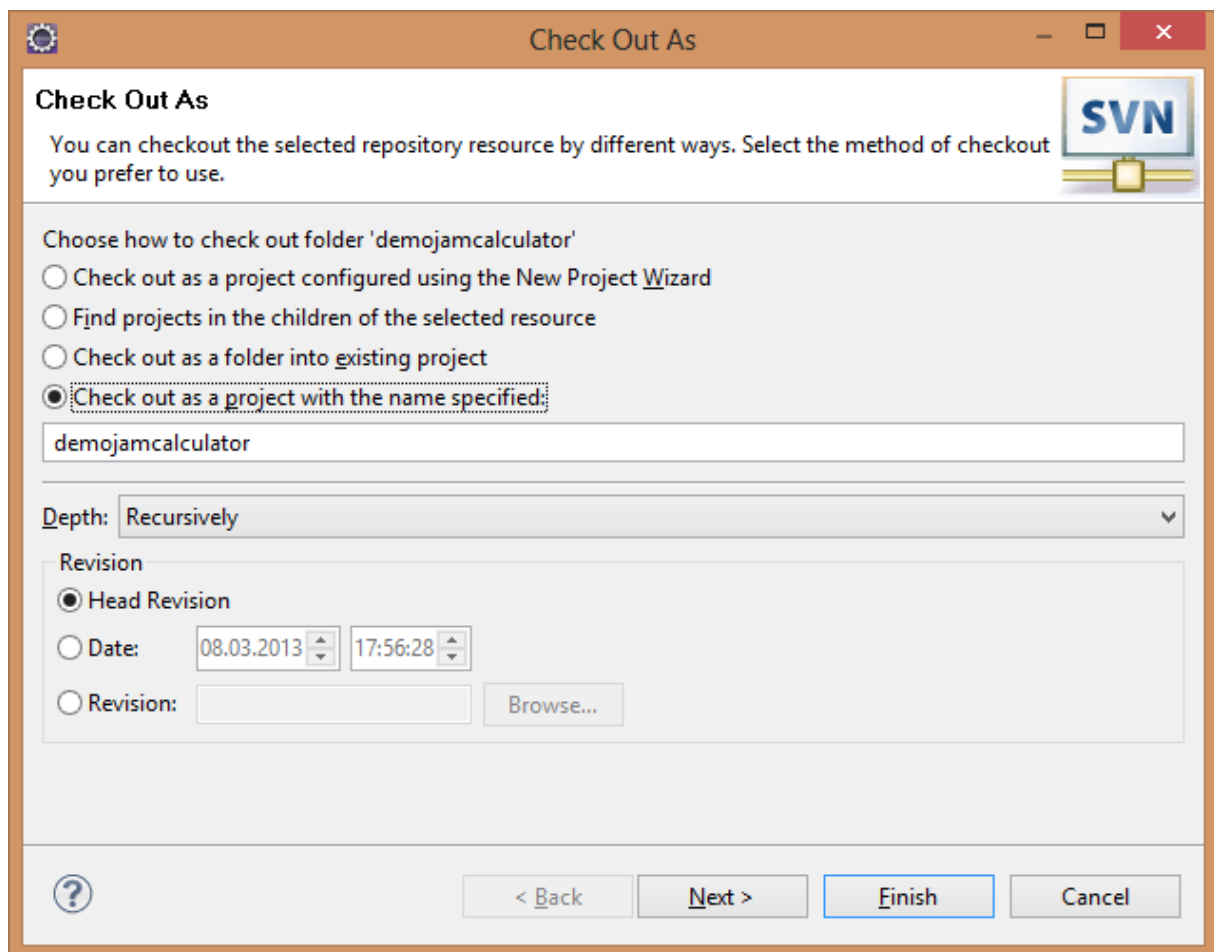
☒ Save authentication (could trigger secure storage login)

To manage your security data, please see ["Secure Storage"](#)

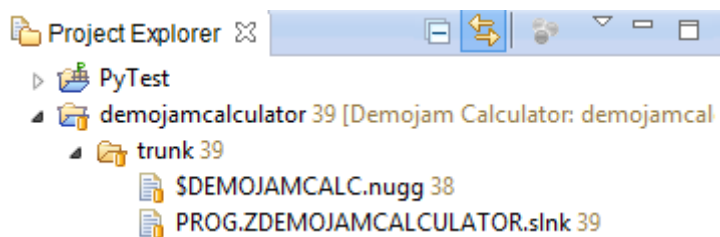
Show Credentials For:

☒ Validate Repository Location on finish

On the next screen confirm the location and revision and press “Finish”.



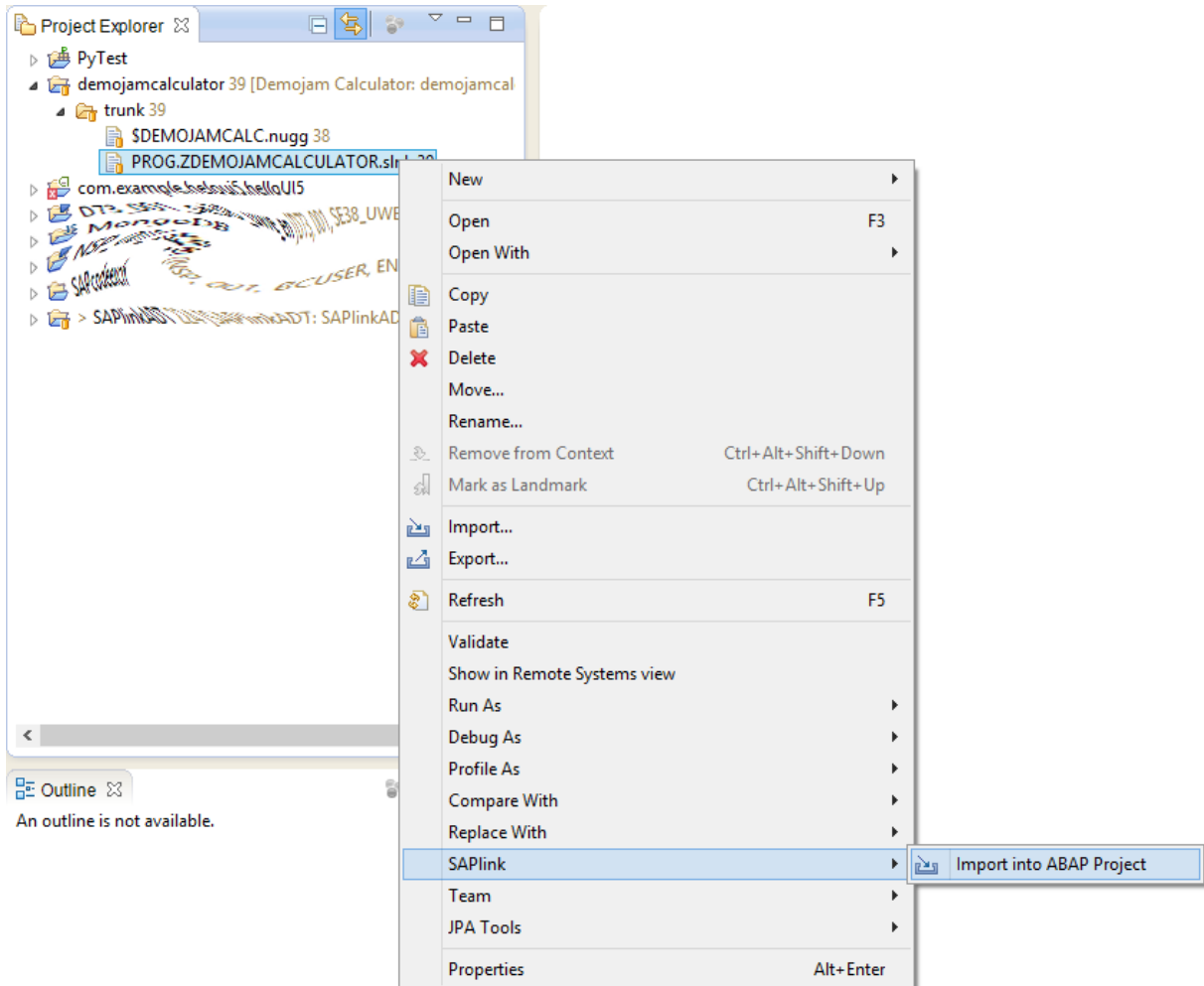
In the Project Explorer now we can see our new project and its content.



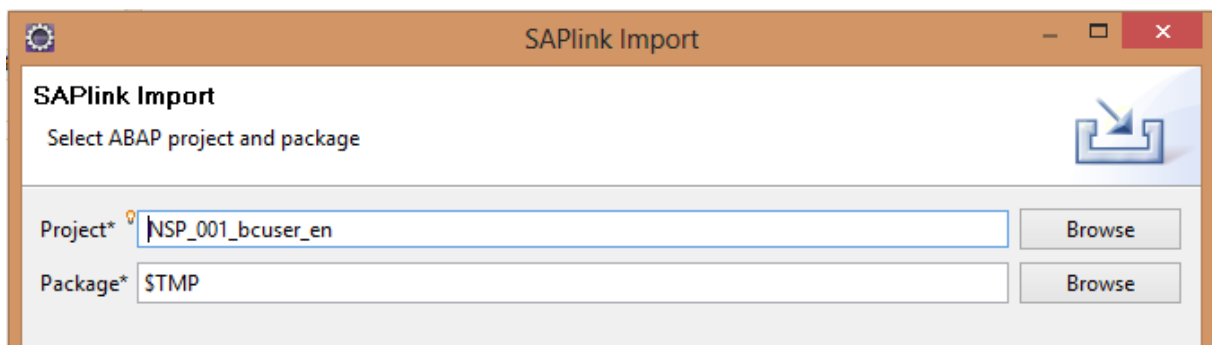
3.2 Import source code into the ABAP system

Right click on the source (in this case “PROG.ZDEMOJAMCALCULATOR.slnk”) and select “SAPlink -> Import into ABAP project”

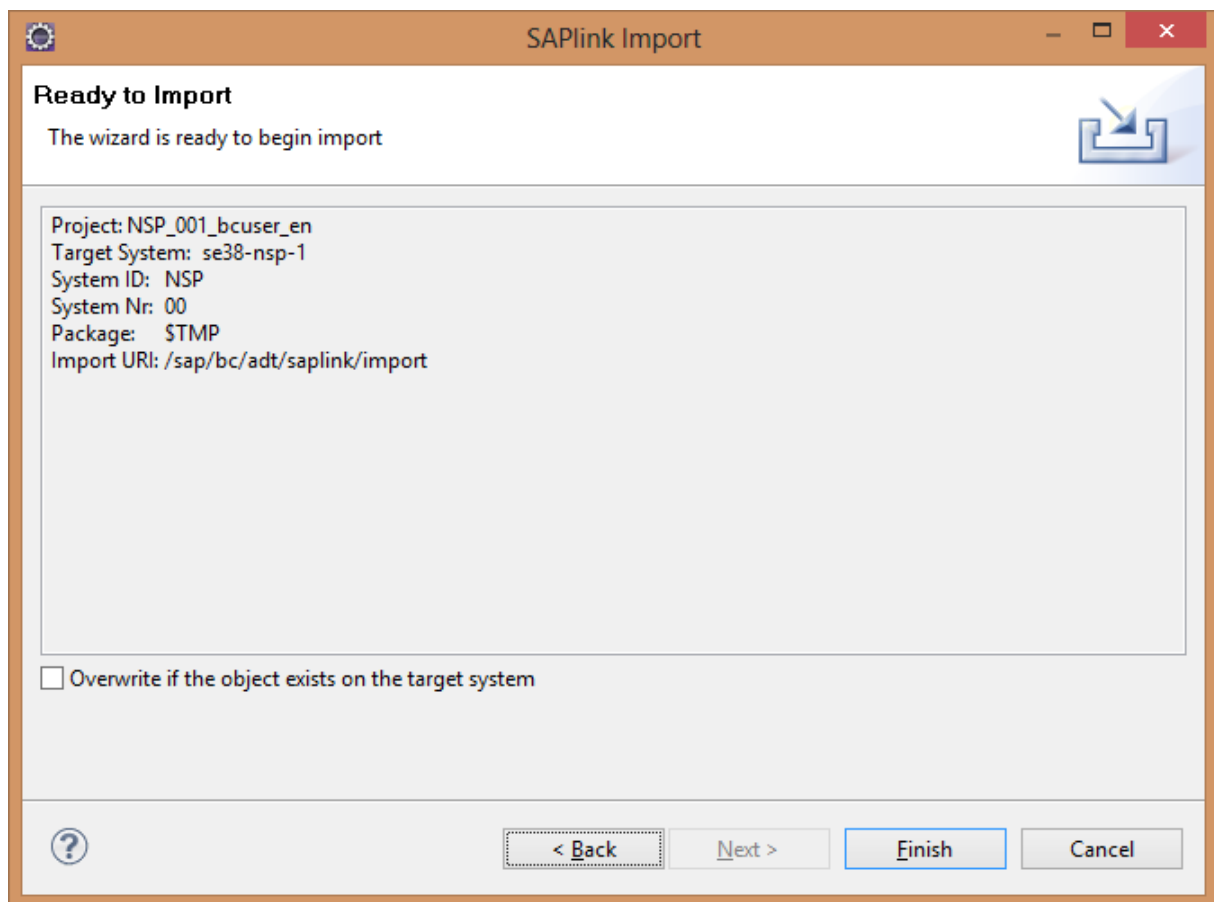
Note: Single objects in SAPlink are so called “Slinkees”. Bundles of objects are called “Nuggets” and have the file ending “.nugg”.



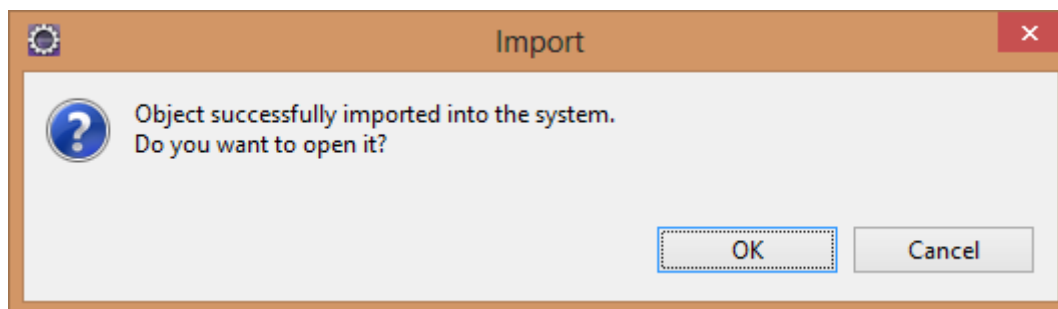
Enter the destination project (the ABAP system), a proper development package and press “Next”.



“Ready to import”: press “Finish”



That's it...

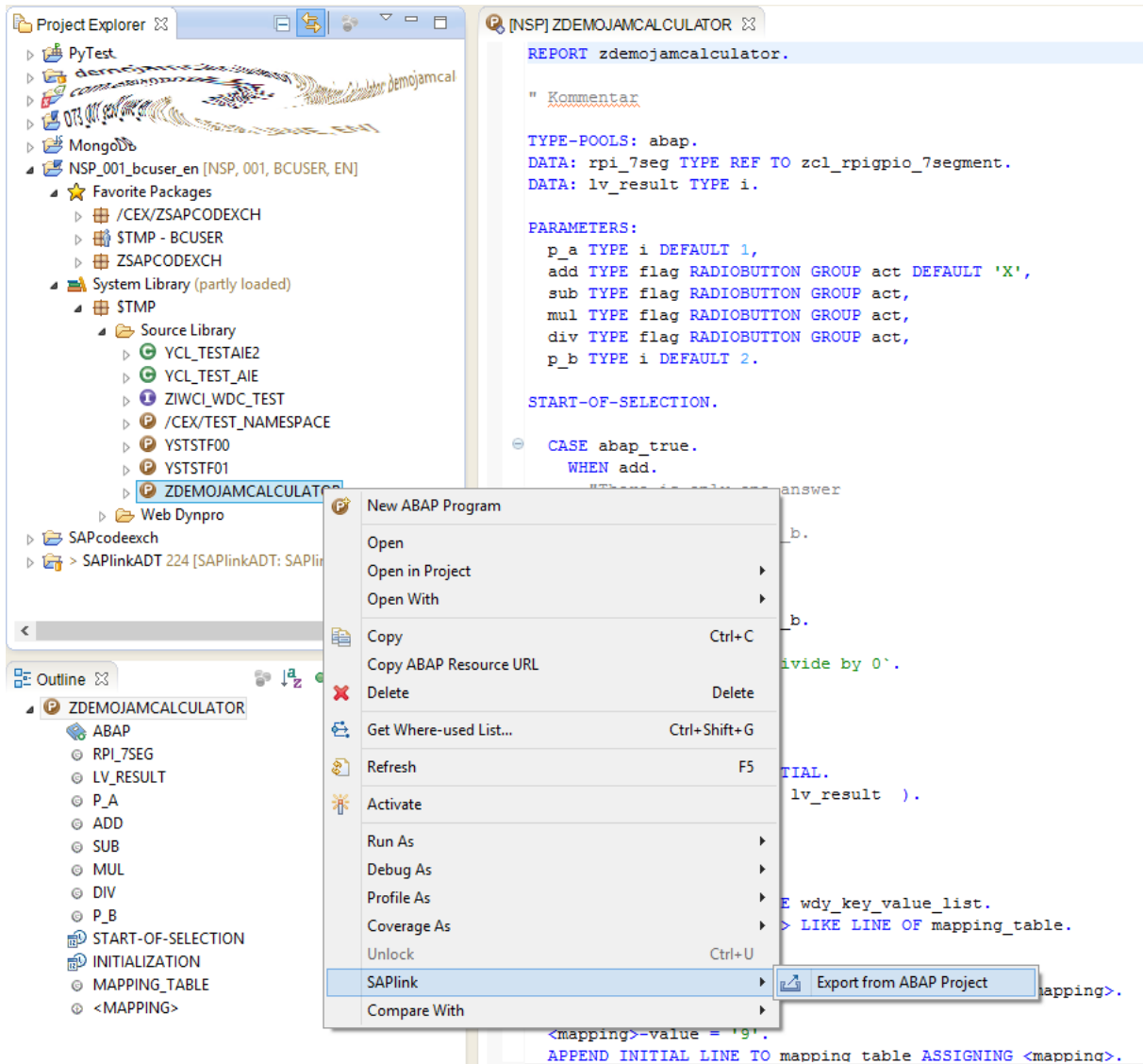


Now you can activate and test the code.

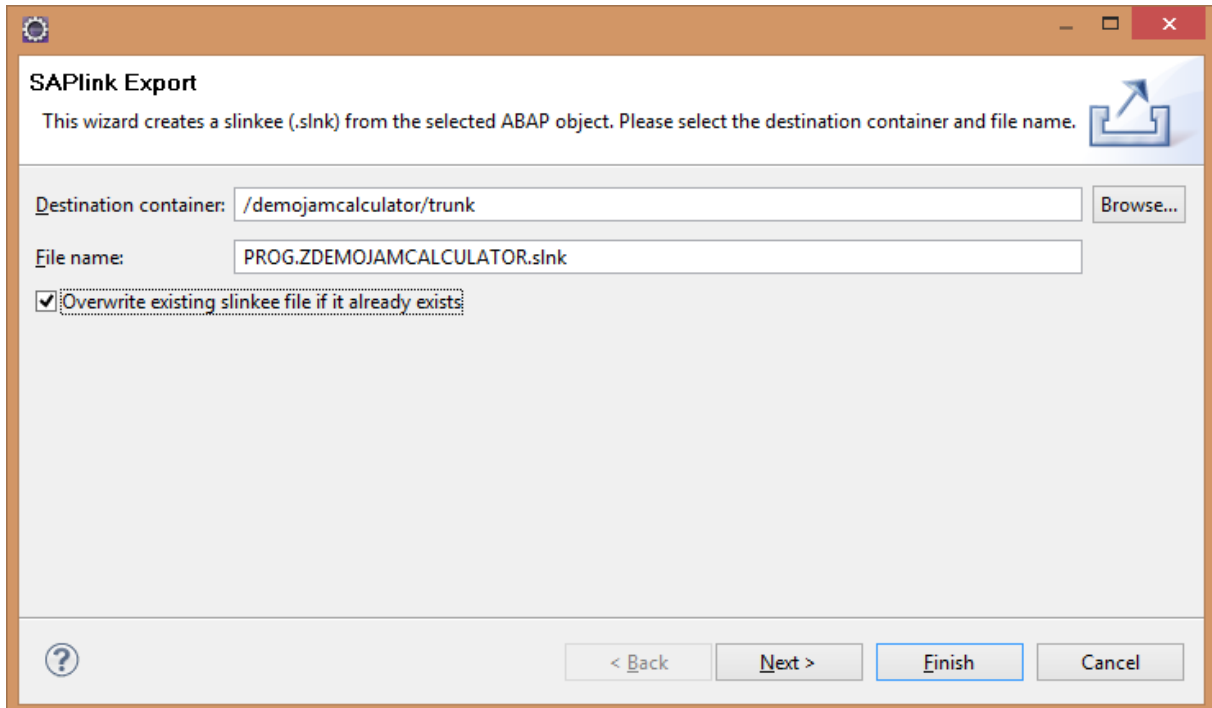
3.3 Export source from the ABAP system

After correcting (and testing!) the code, we are ready to export the Slinker from the ABAP system to the SVN repository.

In the Project Explorer right click on the source name and select “SAPlink -> Export from ABAP project”.



Enter the destination container (the target SVN repository project) and don't forget to check the "Overwrite" flag. Press "Next".

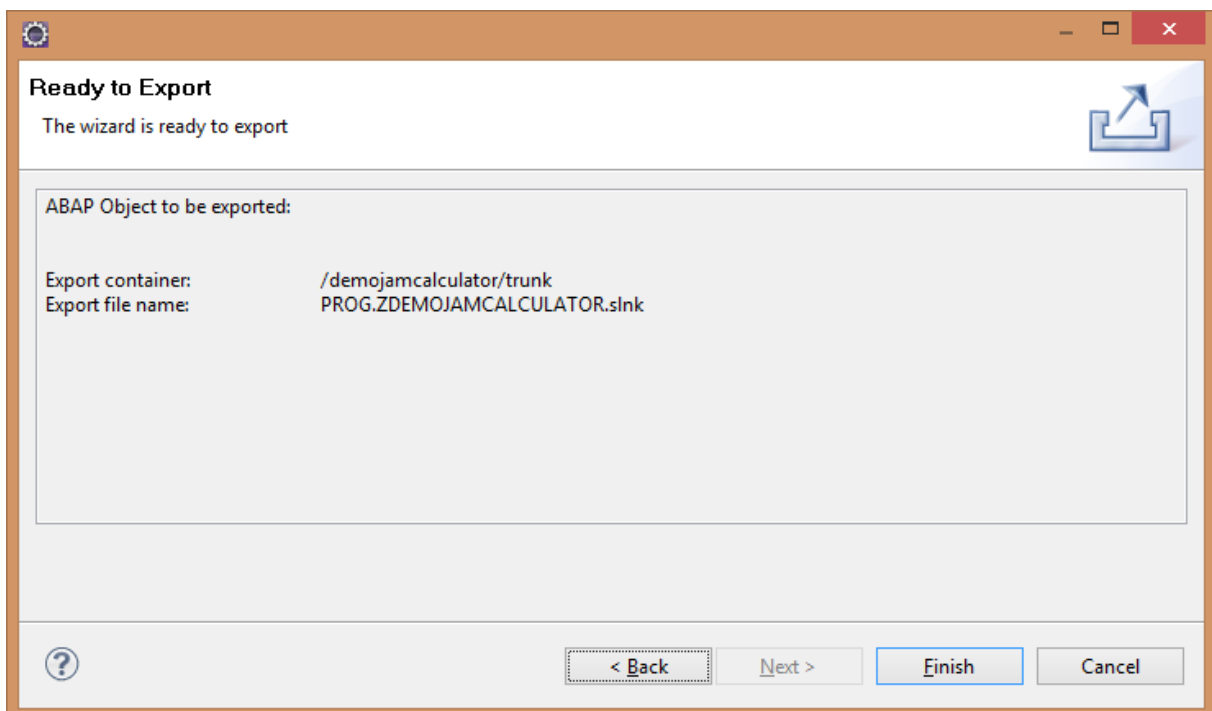


The screenshot shows the 'SAPlink Export' dialog box. It has a title bar with a question mark icon and standard window controls. The main area contains the following fields and options:

- Destination container:** A text field containing '/demojamcalculator/trunk' and a 'Browse...' button to its right.
- File name:** A text field containing 'PROG.ZDEMOJAMCALCULATOR.slnk'.
- Overwrite flag:** A checked checkbox labeled 'Overwrite existing slinkee file if it already exists'.

At the bottom, there is a row of buttons: a help icon (?), '< Back', 'Next >', 'Finish' (highlighted in blue), and 'Cancel'.

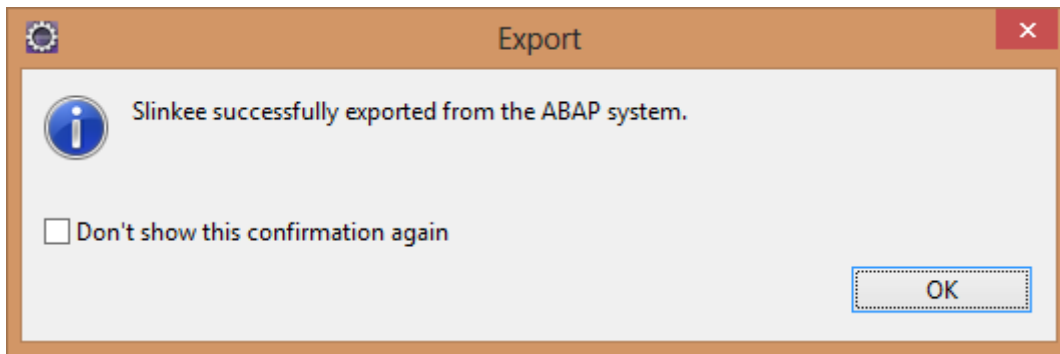
"Ready to Export": Press "Finish"



The screenshot shows the 'Ready to Export' dialog box. It has a title bar with a question mark icon and standard window controls. The main area contains the following information:

- ABAP Object to be exported:** A large empty text area.
- Export container:** A label followed by the value '/demojamcalculator/trunk'.
- Export file name:** A label followed by the value 'PROG.ZDEMOJAMCALCULATOR.slnk'.

At the bottom, there is a row of buttons: a help icon (?), '< Back' (disabled), 'Next >' (disabled), 'Finish' (highlighted in blue), and 'Cancel'.



The Slinkee is now stored in the **local repository** (on your HD). To commit the changes into the SAP Code Exchange SVN repository, right click on the exported Slinkee file and select "Team -> Commit...".

