USER MANUAL

Microsoft Dynamics NAV Authorization Tool

PERMISSIONS LOGGER

TABLE OF CONTENTS: **Page**

COPYRIGHT NOTICE

1 Introduction

Microsoft Dynamics NAV has a built-in security system, so that only persons with the appropriate authorization can gain access to the information contained in the system (see picture). You can customize security to your needs, for example, by restricting access to resources or by tracking user IDs.

Which information can be read, inserted, modified or deleted by the user - all this is completely controlled in Microsoft Dynamics NAV. You can create users, give them roles and modify the permissions of these users and roles from within Microsoft Dynamics NAV.

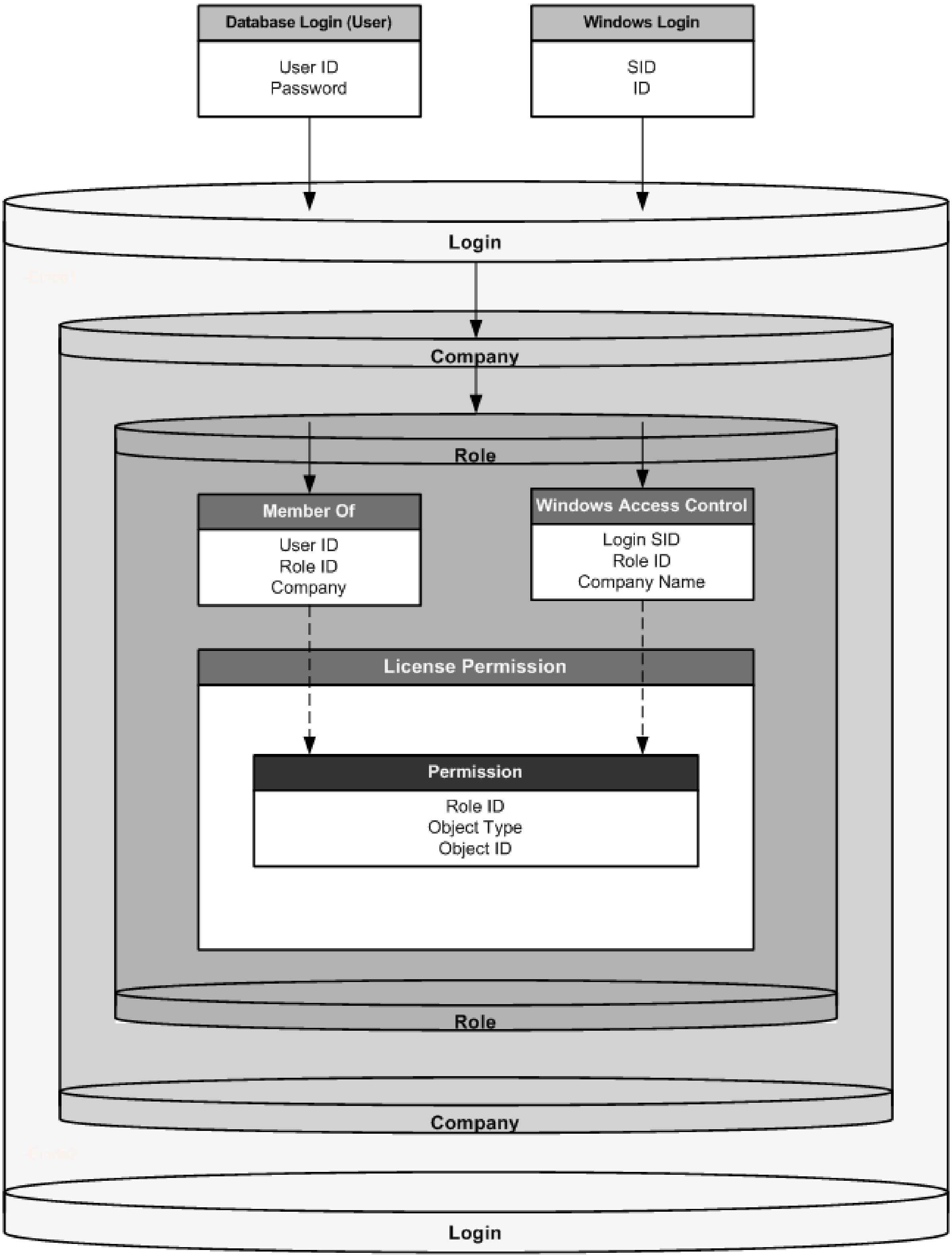
Determining the permissions is a time-consuming task. Using the Permissions Logger this task has been considerably simplified. Moreover, **specific knowledge is no longer required**.

The Permissions Logger is designed to help you analyze, customize and update the Security System of the Navision applications and reduce the time required to correct authorizations. The operation is as follows: the user executes actions (commands) in the system as he/she would normally do. The Permissions Logger supervises and establishes, what objects and with what purpose these actions are being executed.

We pay your attention that this tool does not guarantee 100 % of completeness of permissions. It performs the basic routine work on formation of permissions, but you possibly should add some small part of work manually.

Permissions Logger is compatible with all versions MS Dynamics NAV from 3.0 (Classic Client).
In this version of Permission Logger, only objects logging is available, and not fields logging.

If the application language is English (United States) then the system automatically starts the Client Monitor. Therefore, in order to prevent mistakes it is recommended to close the remaining Navision/ Microsoft Dynamics NAV-applications so that only one application remains active.

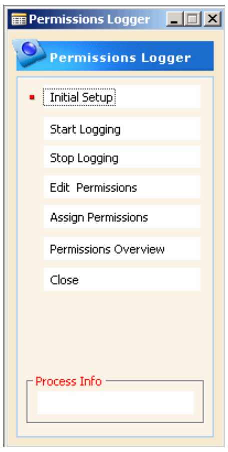


Security System of the Microsoft Dynamics Navision Database

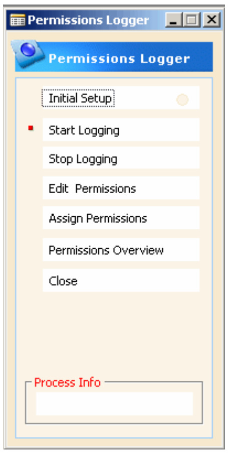
2 Permissions logging

Before you begin, make sure the Automation Server **NavPrmAnalyzer** is installed. For installation instructions see Appendix A.

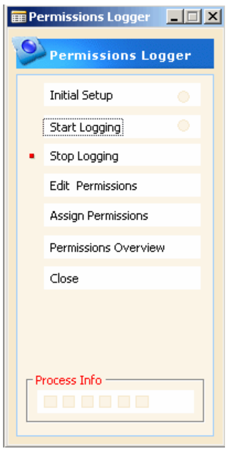
Giving permissions of the security roles is not easy, but with this tool you can do it in 5 simple steps.



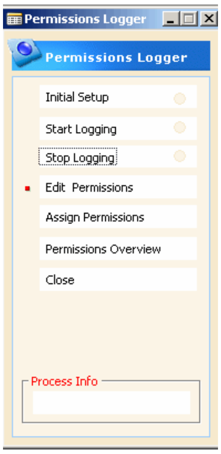
1.Initial Setup ->



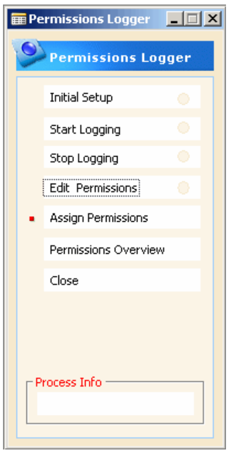
2.Start Logging ->



3.Stop Logging ->



4.Edit Permissions ->



5.Assign Permissions

You log permissions using the Permissions Logger as follows:

1. First change **"Object Cache (KB)"** property (**Tools – Options…**) to 0 and restart Microsoft Dynamics NAV.

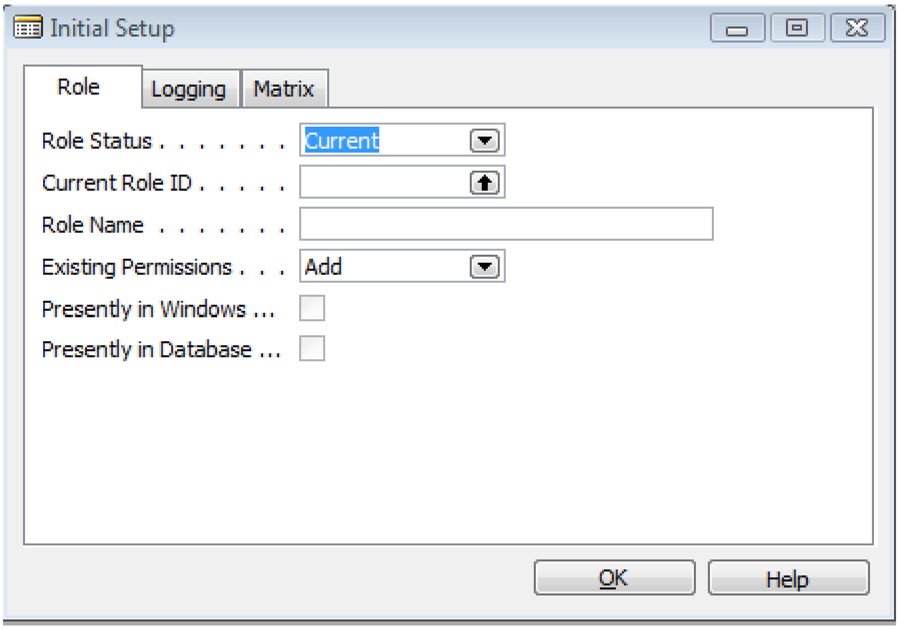
2. Make a new folder with as name e.g. Client Monitor; the location is not important. This folder is required for the storage of programs for automatically starting of the Client Monitor in different Microsoft Dynamics NAV application languages.

3. Modify the extension of the file ScriptSCM1033.txt to ScriptSCM1033.exe and place it in the map Client Monitor.

4. Open Microsoft Dynamics NAV Database as **Super User** with all the permissions and go to Object Designer (**shift F12**).

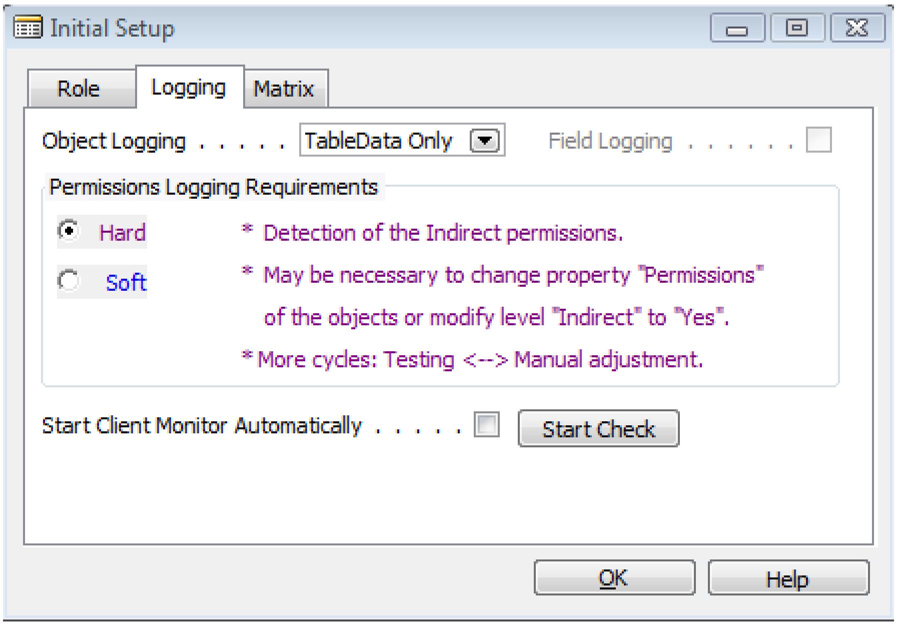
5. Find in the Object Designer Form "Permissions Logger" and Run it.

6. Click **Initial Setup** for the setup procedure. The next form appears:



Role Status . . . . Current
Current Role ID . . . . . .
Role Name . . . .
Existing Permissions . . . Add
Presently in Windows . . . □
Presently in Database . . . □

Select the correct choice for **Role Status**: Current or New.
If -**Role Status** is **New** then indicate **Role Name**, otherwise enter your **Current Role ID** and **Add** or **Delete** Existing Permissions.



Object Logging . . . . . . TableData Only
Field Logging . . . . . . □

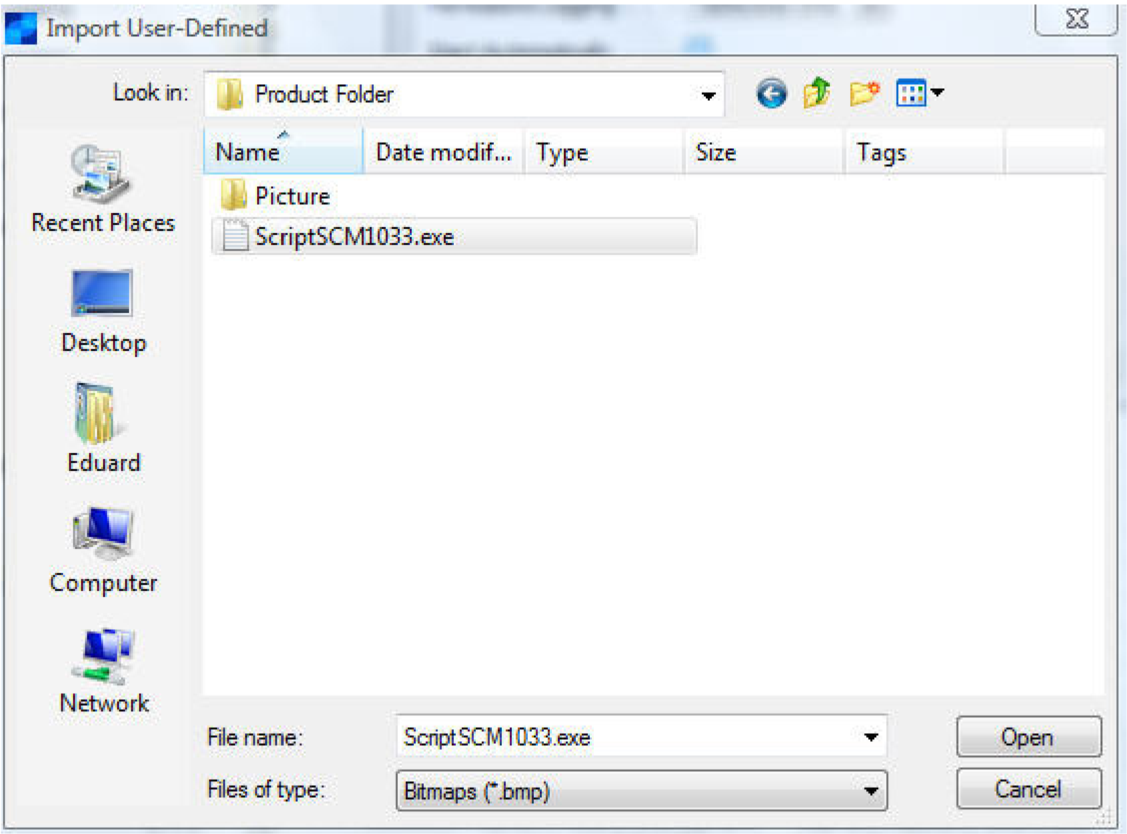
Permissions Logging Requirements

● Hard
○ Soft

* Detection of the Indirect permissions.
* May be necessary to change property "Permissions" of the objects or modify level "Indirect" to "Yes".
* More cycles: Testing <--> Manual adjustment.

Start Client Monitor Automatically . . . . . □

Fill in the other required fields. If you select the check box in the **Start Automatically** field (Tab Client Monitor), then this next form appears:



File name: ScriptSCM1033.exe

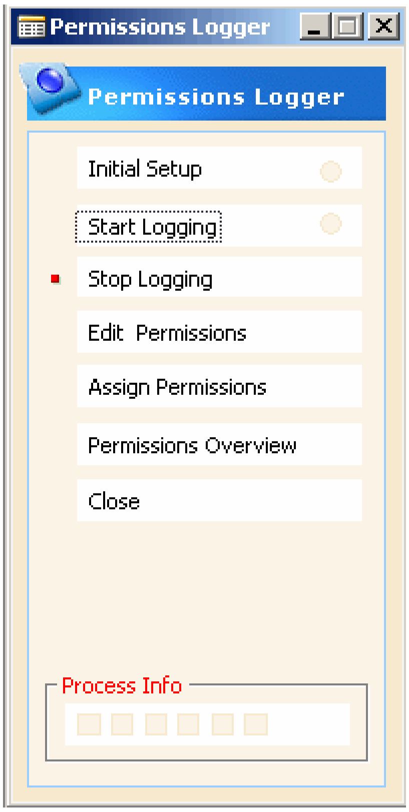
Files of type: Bitmaps (*.bmp)

Find file ScriptSCM1033.exe and click **Open**. Record with BLOB-file has been now added to the Table Permissions Logger Setup.

*Note: You can either have the Client Monitor start automatically (only for application language **English** (1033 ENU)), or you can set the Permissions Logger to start the Client Monitor manually (other application languages). If you have been started the Client Monitor manually, it's not necessary for you to stop it. Performance of the Permissions Logger does not decrease because all old records of the Client Monitor are automatically removed.*

Click **OK** if all fields have been filled.

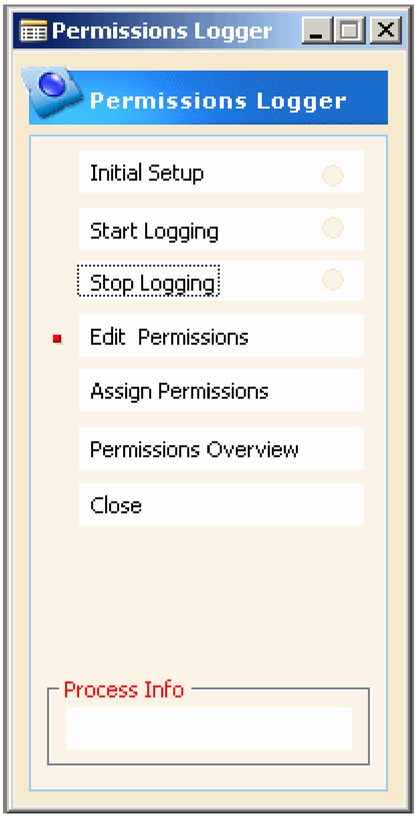
7. Go back on the Permissions Logger and click **Start Logging**.



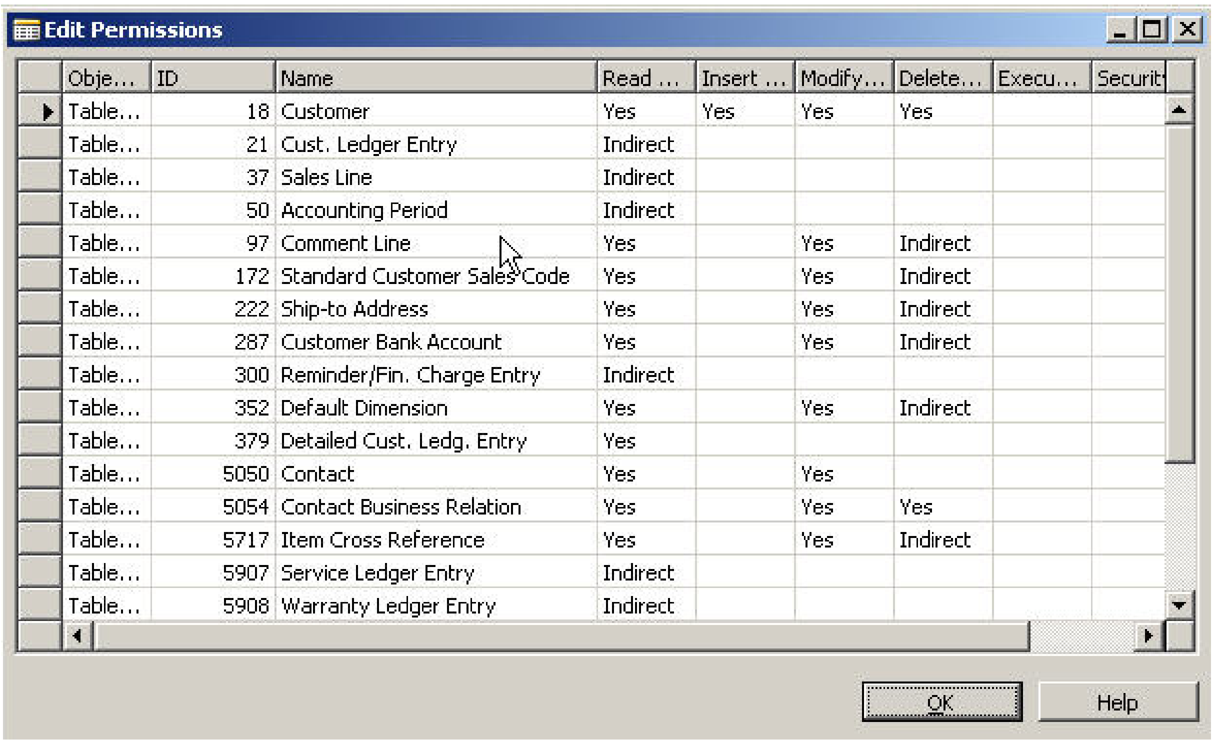
8. You can now start with the implementation of the operations which belong to this role. The program will now register the actions for the role and permissions. You can see that the indicator **Logging** is active.

9. Click **Stop Logging** when you have carried out all operations for the role.

10. Click **Edit Permissions**.



You get now to see which permissions have been determined.

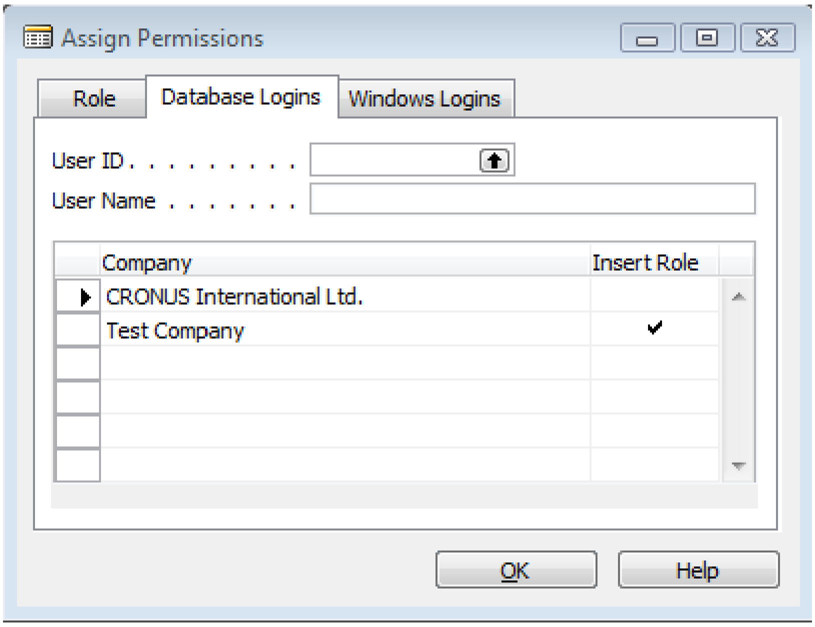


Obje…	ID	Name	Read …	Insert …	Modify…	Delete…	Execu…	Securit
Table…	18	Customer	Yes	Yes	Yes	Yes		
Table…	21	Cust. Ledger Entry	Indirect					
Table…	37	Sales Line	Indirect					
Table…	50	Accounting Period	Indirect					
Table…	97	Comment Line	Yes		Yes	Indirect		
Table…	172	Standard Customer Sales Code	Yes		Yes	Indirect		
Table…	222	Ship-to Address	Yes		Yes	Indirect		
Table…	287	Customer Bank Account	Yes		Yes	Indirect		
Table…	300	Reminder/Fin. Charge Entry	Indirect					
Table…	352	Default Dimension	Yes		Yes	Indirect		
Table…	379	Detailed Cust. Ledg. Entry	Yes					
Table…	5050	Contact	Yes		Yes			
Table…	5054	Contact Business Relation	Yes		Yes	Yes		
Table…	5717	Item Cross Reference	Yes		Yes	Indirect		
Table…	5907	Service Ledger Entry	Indirect					
Table…	5908	Warranty Ledger Entry	Indirect					

11. In some cases it is impossible to identify unequivocally what permissions to give: **Indirect** or **Direct**. In such doubtful cases, Permissions Logger will record the permissions as **Indirect**. If this is incorrect you can now change it to **Yes**. Of course, this can also be corrected at a later stage.

Click **OK** if you are finished correcting the permissions.

12. Click **Assign Permissions**. The next form appears:



User ID . . . .

User Name . . . .

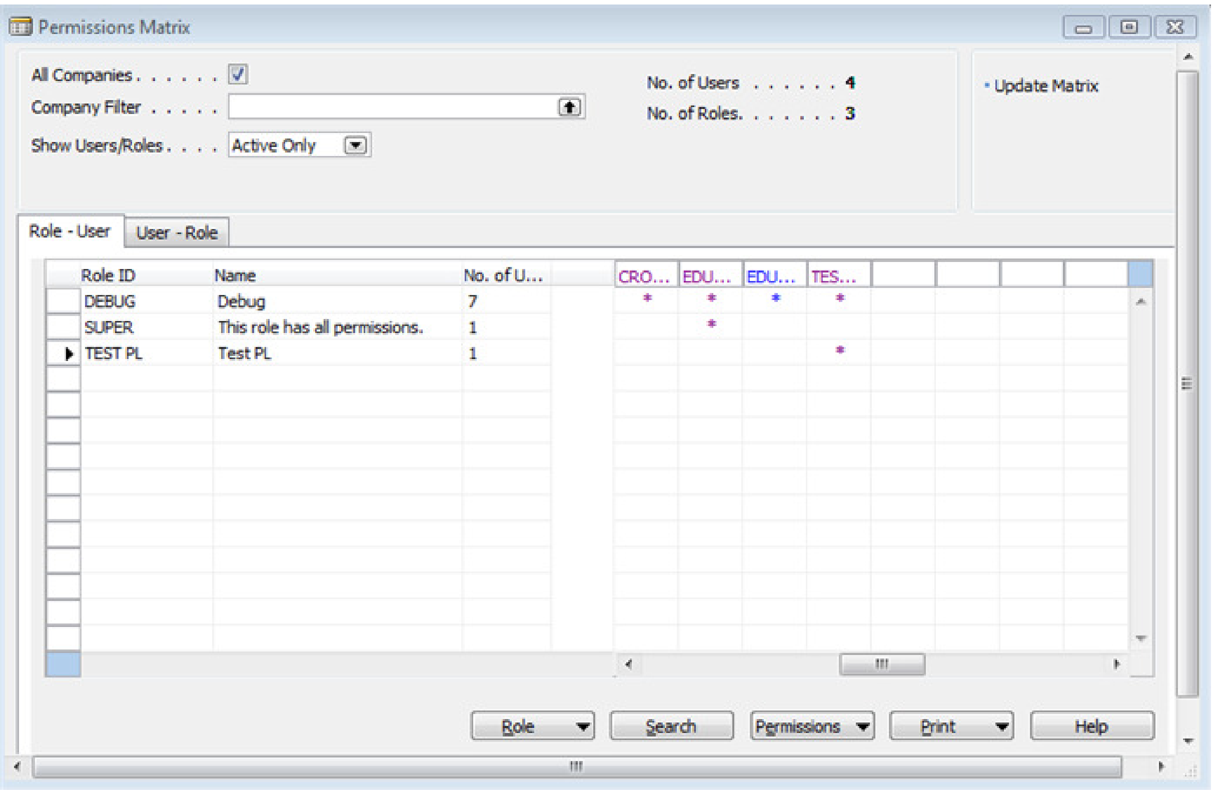
Company	Insert Role
CRONUS International Ltd.	
Test Company	✓

Choose the necessary **Database Logins** and **Widows Logins** to which you wish to attribute this role. If the Role Status is Current and this role already exists, then the **Insert Role** field is automatically selected.
Click **OK** if you are ready.
You will now get the message that the permissions have been already created.

13. Click **OK** and afterwards **Close**.

14. Click **Permissions Overview**.

You now get to see which roles and permissions have been coupled to users. You can also see a total overview of permissions (all roles) of the each user, filtered per Company.
You can use **Permissions Overview** Form to find in what roles is used a particular object, as well as other tasks such as copy the roles to other users, copy to other companies, unassign the roles.



All Companies . . . . . . ☑

Company Filter . . . . .

Show Users/Roles . . . . Active Only

No. of Users 4

No. of Roles. 3

Role ID	Name	No. of U...	CRO...	EDU...	EDU...	TES...
DEBUG	Debug	7	*	*	*	*
SUPER	This role has all permissions.	1		*		
TEST PL	Test PL	1				*

3 Check Results

Sometimes the Permissions Logger only gives you indirect permission to perform operations on some tables. In this instance you use a database object, for example codeunit that has been given extra permission to perform operations into this table. When you have indirect permission for example for ***Purchase Line*** table, you can only modify the table when you run the ***Sales-Post*** codeunit or another object that has permission to modify the ***Purchase Line*** table. Extra permissions can be determined by property **Permissions** of this codeunit. Often, the property **Permissions** is blank or it does not match the performed operations. This occurs also in the standard NAV database. Therefore, it is possible that you get the error: “You don't have permission to..”.
You have two options:

- find the object through which the transaction takes place and fix the properties **Permissions** or
- improve permissions level of the table from **Indirect** to **Yes**.

It is important that you check all automatically created permissions thoroughly, especially with regards to -what is mentioned in step 10 of chapter 2.
Do not forget that each user must be coupled to the general roles.

Note: In the ***SQL Server Option for Navision*** *you may need to synchronize Navision and the SQL Server security system. The synchronization process can be initiated from within Navision. To start the synchronization process, click* **Tools**, **Security** *and* **Synchronize**.

1. Log in with **User ID** to what you have coupled the roles.

2. Make all operations which concern a role.
   If not all permissions are correct you can change them. For this purpose it is necessary to be logged in again as **Super User**. Then go to the menu **Tools**, item **Security** to **Roles**. Select the concerning role and click **Permissions**. Change permissions or add new ones.
3. Changes will take effect after the next login with **User ID**.

4. Installation of NavPrmAnalyzer automation server

Copy the NavPermAnalyser.dll to the MS Dynamics NAV install folder (CSIDE Client or 60).
Start Command Prompt as Administrator and run commands, like the following:

```
C:\Windows\system32>regasm /tlb: NavPrmAnalyzer.tlb "C:\Program
Files\Microsoft Dynamics NAV\60\ NavPrmAnalyzer.dll"
```

or for 64-bit

```
C:\Windows\system32>regasm /tlb: NavPrmAnalyzer.tlb "C:\Program
Files (x86)\Microsoft Dynamics NAV\60\ NavPrmAnalyzer.dll"

C:\Windows\system32>gacutil /i "C:\Program Files\Microsoft
Dynamics NAV\60\ NavPrmAnalyzer.dll"
```

Regasm.exe is the Assembly Registration tool a part of .NET Framework, located in one of the child folders of directory:
C:\Windows\Microsoft.NET\Framework
or for 64-bit
C:\Windows\Microsoft.NET\Framework64
If Regasm.exe is missing in the registration folder, you can copy it from Framework directory.

Gacutil.exe is the Global Assembly Cache tool, located in :
C:\Program Files\Microsoft SDKs\Windows\v6.0A\bin or
C:\Program Files\Microsoft SDKs\Windows\v6.0\bin or in one of the child folders of
C:\Windows\Microsoft.NET\Framework.

If you copy gacutil.exe to another folder/machine/server you might come across the issue, that executing gacutil.exe just seems to do nothing. To fix this problem, you need to copy the files gacutil.exe.config and gacutlrc.dll to the same location where you have copied the gacutil.exe.

You will be getting following messages after successful registration:

```
Types registered successfully.
Assembly exported to 'C:\Program Files\Microsoft Dynamics NAV\60\
NavPrmAnalyzer.tlb', and the type library was registered successfully.
Assembly successfully added to the cache.
```

If you already have installed Trial Version of the automation, you need to uninstall it first. Use the following commands to uninstall the NavPrmAnalyzer from the global assembly cache:

```
C:\Windows\system32>gacutil /u "C:\Program Files\Microsoft
Dynamics NAV\60\ NavPrmAnalyzer.dll"
```

Note that the first time you need to run Navision (fin.exe / finsql.exe) as administrator, if you use the trial version of the Permissions Logger.