

Integration of the Local Repository with SVN repository

Explains in details the steps to integrate the local repository created by NavRepository with a SVN server and repository

Table of Contents

Introduction to SVN Server Integration.....	2
Finding-Out the Local Repository Path.....	2
Creation of a New SVN Module.....	2
Checking-Out of Existing SVN Module.....	3
Synchronization of the Local Changes with SVN Server.....	3

Introduction to SVN Server Integration

“*Subversion (SVN)* is a version control system initiated in 1999 by *CollabNet Inc.* It is used to maintain current and historical versions of files such as source code, web pages, and documentation. Its goal is to be a mostly-compatible successor to the widely used *Concurrent Versions System (CVS)*. *Subversion* was started in 2000 as an effort to write an open source version control system which operated much like *CVS* but which fixed the bugs and supplied the features missing in *CVS*.” –Wikipedia

We are not going to get here into great details about *SVN* features, installation or options. We suggest that you already have an *SVN-Server* installed and properly configured. You should also have a repository for your *Navision* objects created on *SVN-Server*. These are generally tasks for your system or network administrator.

You would also need a *SVN-Client* (such as *SmartSVN*, *TortoiseSVN* or similar). If you don't have one already installed, please take a look at http://en.wikipedia.org/wiki/Comparison_of_Subversion_clients and install a *SVN-Client* of your choice.

Finding-Out the Local Repository Path

In order to integrate the local repository of *Navision* database objects with a *SVN* repository, you would need to know the exact location of the local repository. *NavRepository* creates the local repositories in a way to be easily accessible by *SVN-Clients*.

To find out the local repository path, select the main menu *Navision / Show Database Repository Folder*. The *Repository Folder* dialog will show up. It displays the exact folder path to the local repository of current *Navision* database. You can copy the repository path to the clipboard or open the folder in *Windows Explorer*.

Creation of a New SVN Module

One of the *NavRepository* users should initially create a new *SVN* module on *SVN-Server* (e.g. upload the initial *Navision* database objects to the appropriate *SVN* repository). To do it, that user should follow these steps:

1. Start *NavRepository* and make sure that all *Navision* database objects are committed to the local repository.
2. Find out the exact path to the local repository and copy it to the clipboard (look at the topic *Finding-Out the Local Repository Path* above).
3. Start the *SVN-Client* and select the *Create Module* command.
4. Set the local directory to the local repository path (copied previously to the clipboard).

5. Select the SVN-Server, the SVN repository and the module root.

After that the SVN client will proceed to upload the initial state of the local repository created by *NavRepository* to the SVN repository.

Checking-Out of Existing SVN Module

After the new SVN module on the SVN-Server has been created (e.g. the initial state of the local repository created by *NavRepository* has been uploaded to the SVN repository), all other interested users (that need to keep the same *Navision* database objects in sync) need to check out that module **before** they start committing any changes to their local repositories. To do the module check-out, each user should follow these steps:

1. Start *NavRepository*.
2. Find out the exact path to the local repository and copy it to the clipboard (look at the topic *Finding-Out the Local Repository Path* above).
3. Select the SVN-Server, the SVN repository and the module root.
4. Set the local directory to the local repository path (copied previously to the clipboard).
5. Ensure that you're checking out the SVN-repository recursively.

After that the SVN client will proceed to download the initial state of the SVN repository (created by the first user in the topic *Creation of a New SVN Module*) to the local repository.

Synchronization of the Local Changes with SVN Server

After all users of the same SVN-repository have checked out the SVN-module (have synchronized their local repositories with the SVN repository) they can use *NavRepository* to start committing the *Navision* object changes from *Navision* database to the local repository or from the local repository to the corresponding *Navision* database.

After committing a several changes from the database to the local repository, each user should start his SVN-Client and commit these changes from the local repository to the SVN-repository in order to make them available to the other SVN-repository users.

It is a good practice to check also regularly the SVN-repository for updates (e.g. updated objects committed by other users of the same SVN-repository). The oftener you do this, the smaller is the chance to get into object-definition conflicts.

Commits and Updates covered above are only the most basic SVN-Client commands. The SVN servers and clients have other very useful features, such as conflict resolution, source merge, log of changes for each Integration of the local repository with SVN repository

file, transaction logs, etc. It's only a matter of time to learn the more advanced features and make use of the full power of *SVN* repositories.