

# **Project management – integrated into Outlook**

InLoox PM 7.x off-line operation

## **An InLoox Whitepaper**

Published: November 2011

Copyright: © 2011 InLoox GmbH.

You can find up-to-date information at <http://www.inloox.com>

The information contained in this document represents the current view of InLoox on the issues discussed as of the date of publication. Because InLoox must respond to changing market conditions, it should not be interpreted to be a commitment on the part of InLoox, and InLoox cannot guarantee the accuracy of any information presented after the date of publication.

This White Paper is for informational purposes only. INLOOX MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS DOCUMENT.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of InLoox GmbH.

© 2011 InLoox GmbH. All rights reserved.

IQ medialab and InLoox are either registered trademarks or trademarks of InLoox GmbH in Germany and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

## Contents

General information .....	1
What is a replication? .....	1
The replication-distribution-model .....	1
The publisher .....	2
Article .....	2
Publication .....	2
The distributor .....	2
The subscriber .....	2
Subscription .....	2
Replication types .....	2
Snapshot replication .....	2
Transaction replication .....	3
Merge replication .....	3
Configure the replication .....	4
Requirements on the client and SQL server .....	4
Client .....	4
SQL server .....	4
Configure the snapshot folder .....	4
Prefaces .....	4
Configure the distributor .....	6
Configure the publication .....	12
Central subscription configuration on the server .....	19
Subscription type comparison .....	21
Configure the synchronization for a pull subscription .....	43
Setup a sync schedule .....	44
Initialize a client created subscription .....	46
Changing InLoox database connection on the Client .....	48
Replication – Upgrade the InLoox database .....	50



## The publisher

The publisher is the server/database instance, which provides data for replication on other servers. The publisher can possess one or more publications, which in each case contains a logically combined set of objects and data that can be replicated.

The publisher also contains the **articles** and the **publication**.

### Article

Article is the name for the database objects in a publication. A publication can contain different kinds of articles - from tables over views up to stored procedures and other objects. If tables are published as articles, it can be specified by filters, which columns and lines of the table are sent to the subscribers.

### Publication

A publication is a listing of one or several articles from a data base. The grouping of several articles to a publication facilitates the indication of a logically combined set of database objects and data, which are replicated as unit.

## The distributor

It's about servers, which distribute replicated data. The distributor maintains all data for the publications. The distribution database stores replication status data and metadata about publication and functions in some cases as a queue for data, which are distributed to subscribers by the publisher. In many cases the same database server instance contains both roles publisher and distributor. Such database server instances are also called local distributors. If the publisher and the distributor are on different database server instances, the distributor is called remote distributor.

## The subscriber

It's about the target servers for the replication. These servers store the replicated data and receive actualizations. The SQL Server 2005 can send data to all kinds of subscribers, also to Oracle or Microsoft Access. The subscribers can fetch the data from the distributor (pull subscription) or it is sent to them automatically (push subscription).

### Subscription

A subscription is understood as the request of a publication copy by a subscriber. The subscription defines, where and when which publication will be received. There are two kinds of subscription: push and pull.

## Replication types

Although the comparison with a magazine subscription is helpful for the understanding of the Replication, it has to be pointed out that the SQL server replication has additional features, which cannot be represented by this model. SQL Server 2005 can send the entire publication (**Snapshot replication**), send only the changes (**Transaction replication**) or send and receive the changes (**Merge replication**) to and from the subscribers.

### Snapshot replication

The first replication type is the snapshot, where the entire publication is send to the subscribers by the distributor. It can be created and implemented very simple. Actually all replication types proceeds from a snapshot, because the data need a starting point to receive the changes

### Transaction replication

With the transaction replication the transaction protocol is monitored for changes, which are transmitted to the distributor. The changes are either collected by the subscribers or send to them.

The procedure begins with a base snapshot. The most fastidious part with the setting-up of this variant concerns the approach with a conflict, if a data element is changed on both sides.

The transaction replication is typically used in pure server environments, which require a high throughput, and is suitable for the following cases: Improve the scaling and availability, Data warehousing and report servers, integrating data from several locations, integrating of heterogeneous data and outsourcing the batch processing.

### Merge replication

The Merge replication begins with a base snapshot. Changes at the publisher are sent to the distributor and then to the subscribers. A subscriber can change the data and sends the results to the publisher. Here the question arises, what happens, if the data is changed in both places, which is called conflict. Here three solution types exist. The first method lets the publisher always win, the second the subscriber, in the third case the user has to handle the conflict himself.

The merge replication is mainly for mobile applications or distributed server applications with possible data conflicts. The following frequent scenarios belong to it: Data exchange with mobile users and integrating data from several locations.

We recommend a **merge replication** for InLoox PM.

## Configure the replication

### Requirements on the client and SQL server

#### Client

To be able to replicate with the server the following software has to be installed on the client:

- ✓ **SQL Server Management Studio Express Edition (Version 2005 SP2, 2008 or 2008 R2)**
- ✓ **SQL Server Express Edition (Version 2005 SP2, 2008 or 2008 R2)**

#### SQL server

On the server the following has to be installed and configured:

- ✓ **SQL Server Workgroup, Standard or Enterprise Edition (Version 2005 SP2, 2008 or 2008 R2)** An Express Edition on the server is not sufficient, in order to be able to create a publication
- ✓ **SQL Server Management Studio (Version 2005 SP2, 2008 or 2008 R2)**
- ✓ **A snapshot folder**, which is used as standard for all publishers who use this distributor. It's advisable to create this folder before you start with the configuration. The folder has to be shared. See in addition the following chapter „Configure the snapshot folder“

Please note that the Client SQL Server Version should not be a new version than the main SQL server.

### Configure the snapshot folder

#### Prefaces

The snapshot folder is a folder, which stores snapshot data. It is advisable to reserve this folder exclusively for storing snapshots. Grant the snapshot-agent write access on the folder, and make sure that only the Windows account receives read authorization, which is used by the merge agent and/or the distribution agent to access the folder. The agent assigned Windows **account must** be a **domain account**, so that the snapshot folder on the remote computer can be accessed.

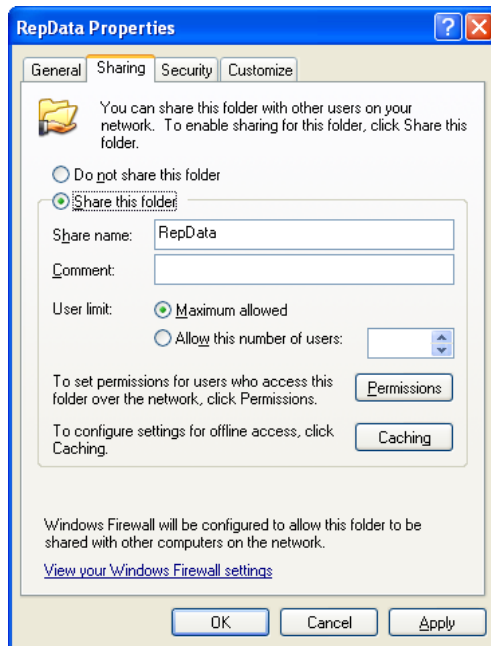
Windows Vista contains a new feature, the User Account Control (User Account Control, UAC), which administrators use to administer their extended user permissions (also called privileges). According to standard administrators do not use their administrator permissions on Windows Vista during execution. Instead most actions are executed as standard user (not as administrator) and administrator permissions are only used temporarily, if necessary. Eventually the user account control prevents the administrator access to the snapshot folder. You have to assign explicit authorizations to the Windows accounts used by the snapshot agent, distribution agent and merge agent for the snapshot folder. This is also necessary if the Windows accounts are members of the administrator group.

If you configure a distributor with the distribution configuration wizard or with the wizard for new publication, according to standard the local listing

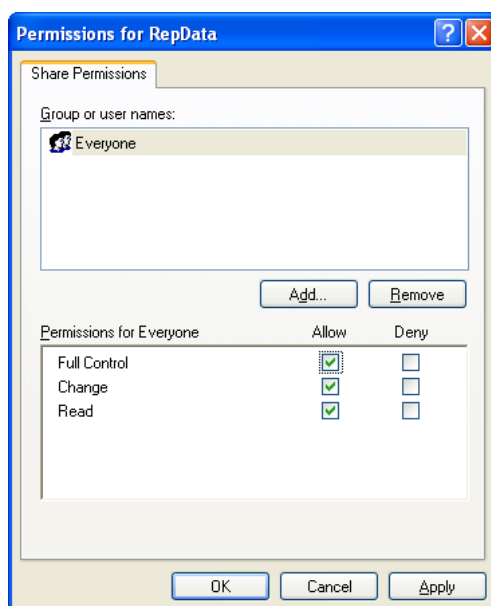
**X:\Programme\Microsoft SQL Server\<Instancename>\MSSQL\RepIData** is used as snapshot folder. If you use a **remote distributor** or a **pull subscription**, you must use a UNC-network-share (e.g. \\<Computername>\Snapshot) instead of a local path.

To configure a snapshot folder where the publication snapshot is created and stored proceed as follows:

1. Navigate to the SQL Server 2005 data folder using the Windows Explorer. The default folder is **C:\Program Files\Microsoft SQL Server\MSSQL1\MSSQL\**
2. Create a new folder with the name **RepData**
3. Right click this folder, and click on **Sharing and Security**
4. In dialog **RepData Properties** click on **Share this folder** below the tab **Share**. Make sure that the **Share name** is set to **RepData**



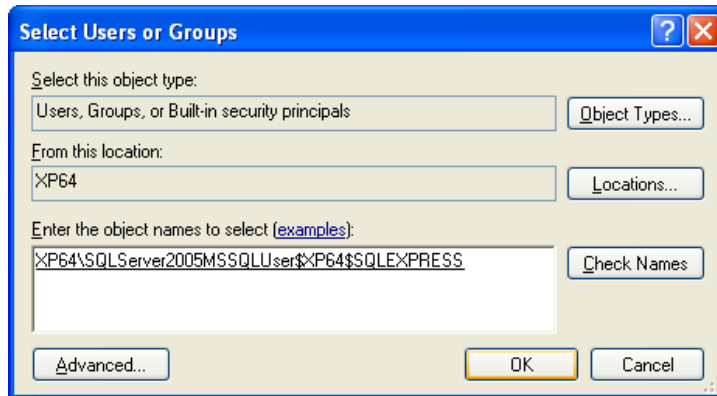
5. Click on **Permissions**
6. In the **Permissions for RepData** window click on **Add** if you like to add certain users, or select the group name of **Everyone** and check **Full Control**, if you would like to give full control to all users on the folder



7. In the window **Select Users or Groups** enter the name of the snapshot agent account or another account into the text field **Enter the object names to select**



- Click on **Check Names**, and then click on **OK**



## Configure the distributor

In this part you configure the distribution on the publisher and specify the necessary authorizations for the distribution and publication databases. If the distributor already has been configured, the publication and distribution has to be deactivated at first. You can deactivate the publication and the distribution with the help of the publication and distribution wizard.

The distributor is a server, which contains the distribution database. In the database metadata and history data for all replications and transaction types of the transaction replication are stored. To setup the replication you have to configure a distributor. Each publisher can only be assigned to one distributor instance however a distributor can be released for several publishers.

For configuring the distributor you proceed as follows:

- Start **SQL Server Management Studio**
- Connect to the Server via **SQL Server Management Studio**, which should work as distributor, and extend the server node (in many cases the publisher and the distributor are on the same server)
- Extend the **Replication** node
- Right click on the **Replication** object
- Then click on **Configure Distribution...**

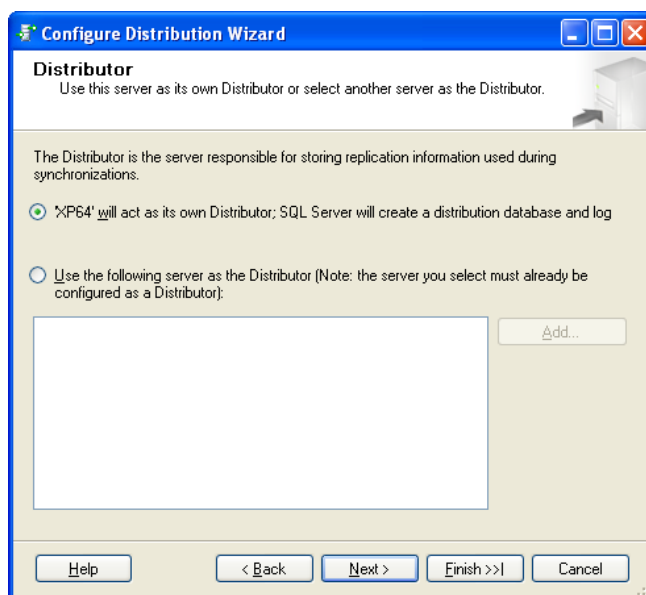


6. The **Configure Distribution Wizard** is started:



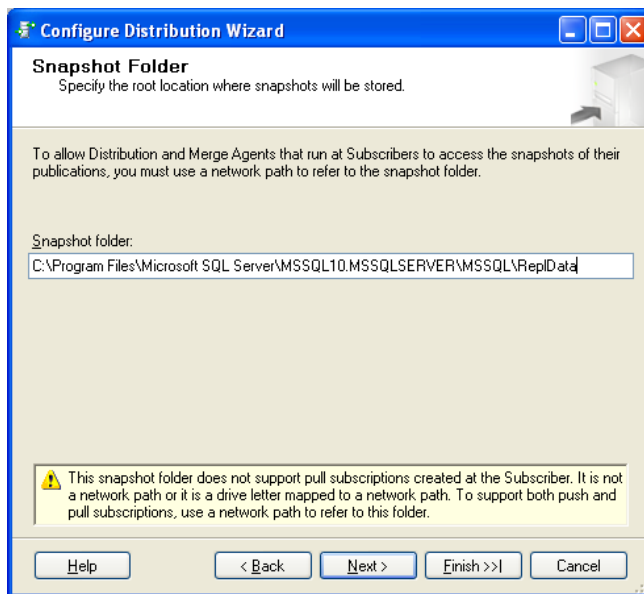
7. Click on **Next**

8. The window **Configure Distribution Wizard** and **Distributor** appears:



9. Here you select the server, which you determined as distributor
10. As suggestion the server you are connected to appears, thus e.g. **MSWINSERVER**. Thus you determine that distributor and publisher reside on the same server
11. **Optionally:** Select **Use the following server as the Distributor**, then choose the server. Important: The server already has to be configured as distributor and the usage of the distributor has to be activated on the publisher. If you select a remote distributor, you have to enter a password for connections on the page **Administrator Password**, which are created between the publisher and the distributor.
12. Click on **Next**

13. The **Snapshot Folder** window appears



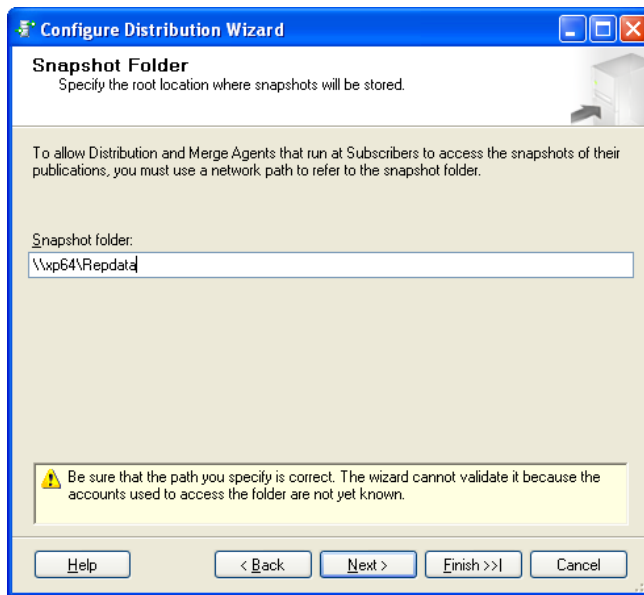
14. Independently of their type each publication needs a base snapshot. This Snapshot is stored in a folder, which is used together by all replication agents.

The standard path is **C:\Program Files\Microsoft SQL Server\MSSQL.1\MSSQL\RepData** on the server, which is intended as distributor.

**Important: pull subscriptions**, which were created on the subscriber, are **not** supported by this snapshot folder. It is not a network path, or it is a drive letter which is linked to a network path. Refer to this folder with a network path, in order to support both **Push- and pull subscription**, e.g. **\\mswinserver\Repldata**. If you choose to use a **pull subscription** later, you just have to specify a **network drive** here, e.g. **S:\Replication\Repldata**

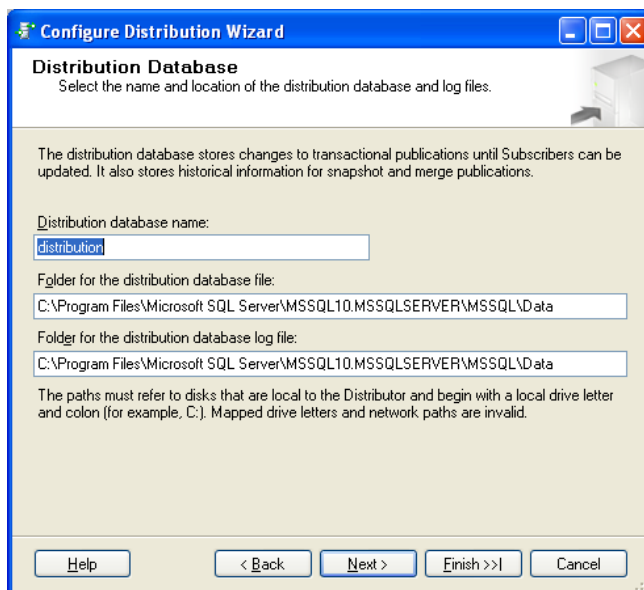
15. With a **push subscription** the publisher is responsible to make a replication of all changes on all subscribers without request. Usually push subscriptions are used, if changes have to be sent immediately to the subscribers or if actualizations have to be executed periodically. Since the publisher initiates the replication, push subscriptions also offer a higher security than pull subscriptions. However the work for the publisher is increased. With pull subscriptions the subscribers demand periodic actualizations of all changes from the publisher. Usually pull subscriptions are used at a high number of subscribers or if the work for the publisher has to be reduced. Pull subscriptions are also suitable for independent mobile users.

16. Create a **push subscription** and a network share on the Windows server with **\\mswinserver\repldata** at first



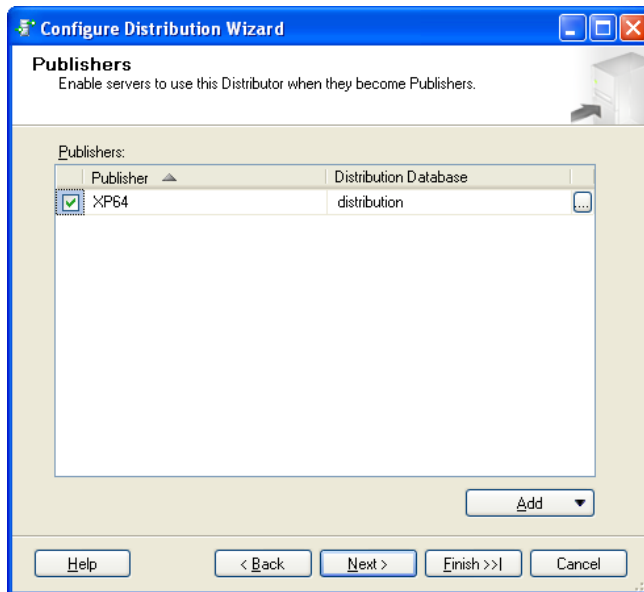
17. Click on **Next**

18. The window **Distribution Database** appears



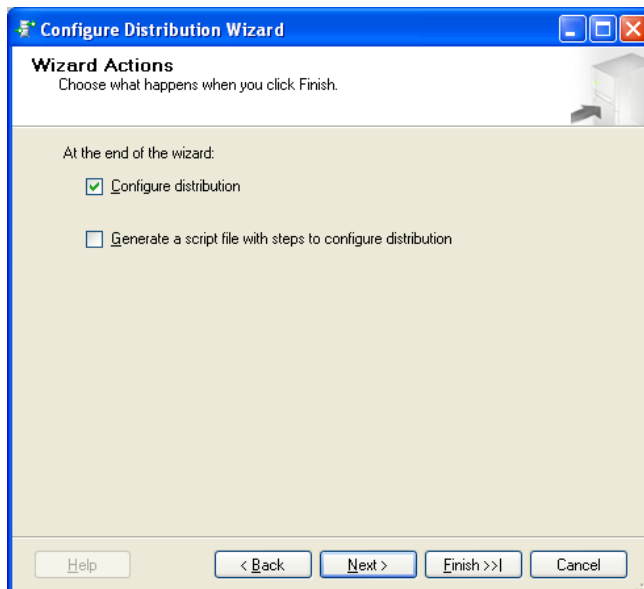
19. In the **Distribution Database** window you configure the distribution database, which the distributor needs, in order to keep track which actions were accomplished. Click on **Next**

20. The **Publisher** window appears



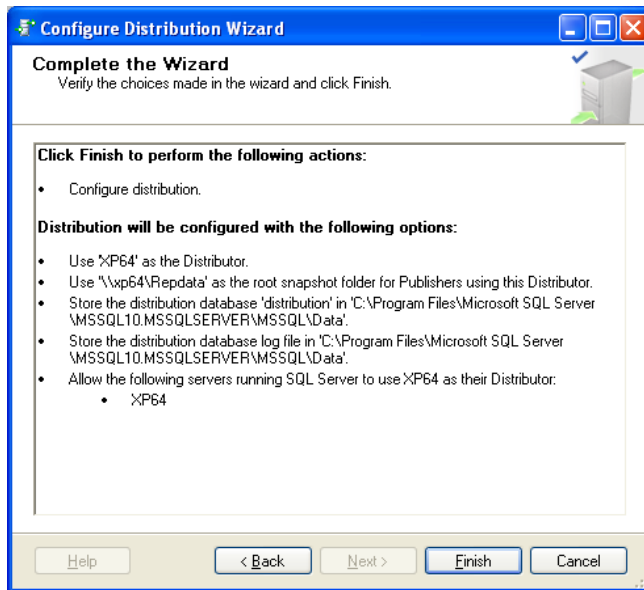
21. Here you see all servers on the network and you are able to select the publishers, who use this system as distributor. If you like to add a server, so that it can be used as distributor, as soon as it is changed to a publisher, click on **Add** and select one of the offered possibilities afterwards

22. Click on **Next** and the window **Wizard Actions** appears



23. Click on **Next**

24. The window **Complete the Wizard** appears



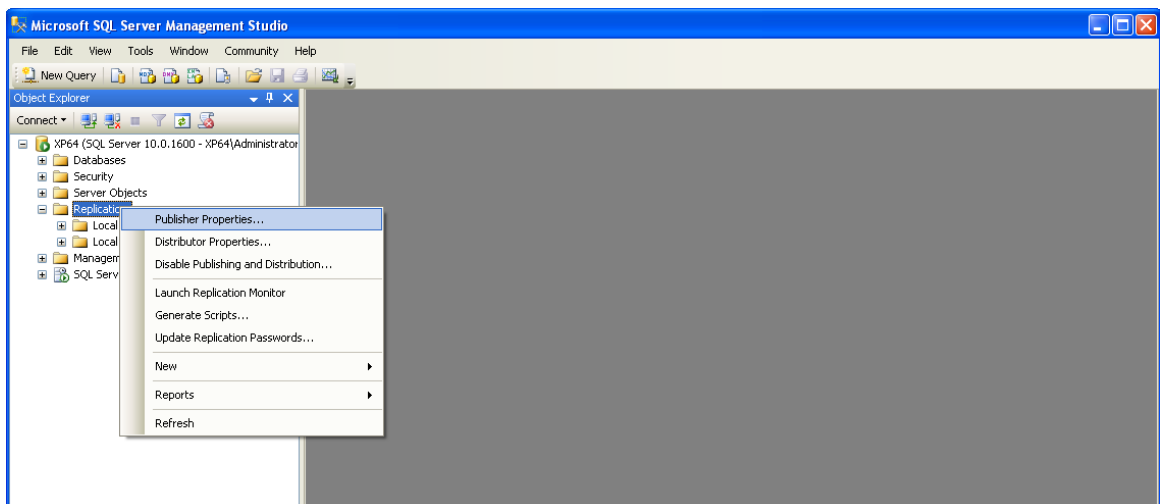
25. Click on **Finish**

The publisher will be installed

26. Click on **Close**

The distributor is established and operational

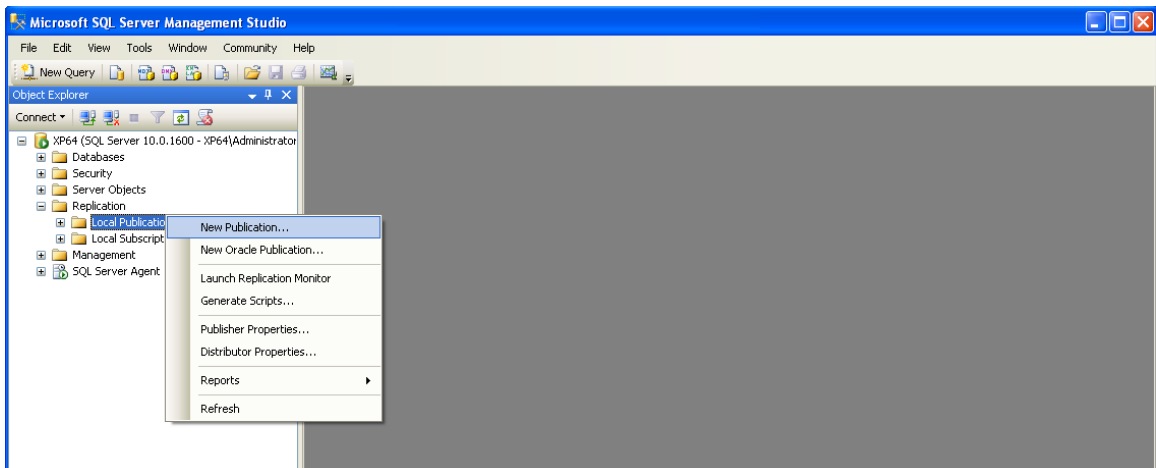
27. You can verify the settings status and if the publisher is set up correctly, by right clicking on the **Replication** node



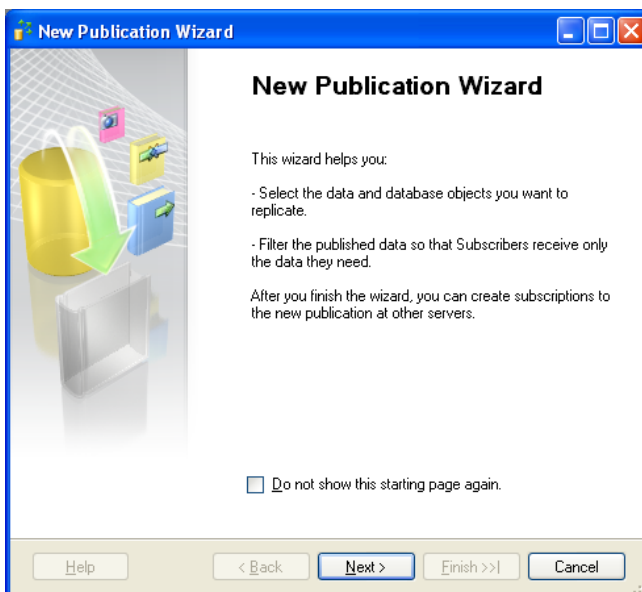
28. Here you can examine the **Publisher Properties**

## Configure the publication

1. Start **SQL Server Management Studio**
2. Connect to the desired server
3. Extend the **Replication** node
4. Right click in the **Object Explorer** on the **Local Publications** node
5. Click on **New Publication...**

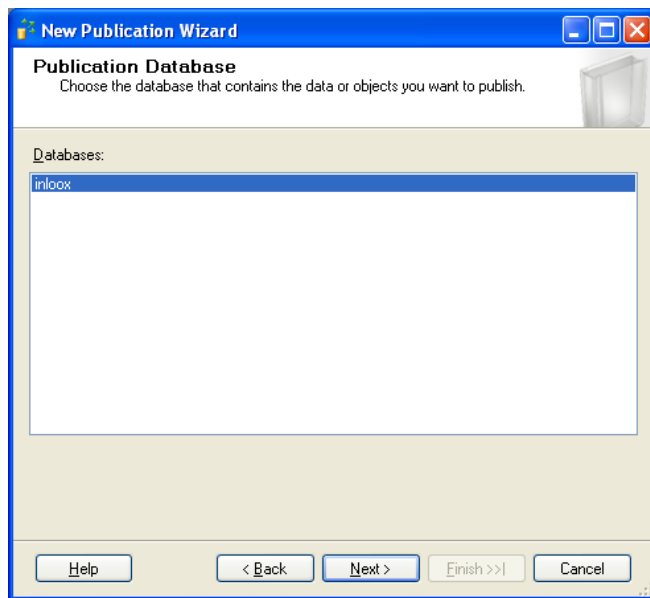


6. The window **New Publication Wizard** appears



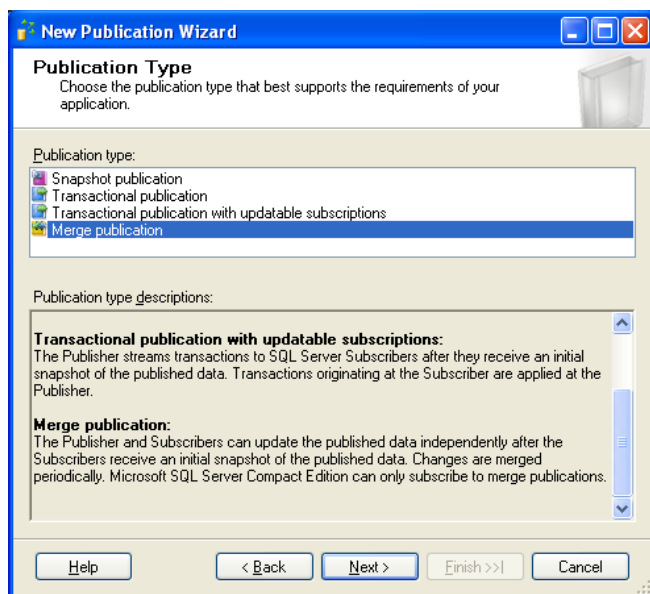
7. Click on **Next**

8. The window **Publication Database** appears



9. Select the desired database and click on **Next**

10. The window **Publication Type** appears

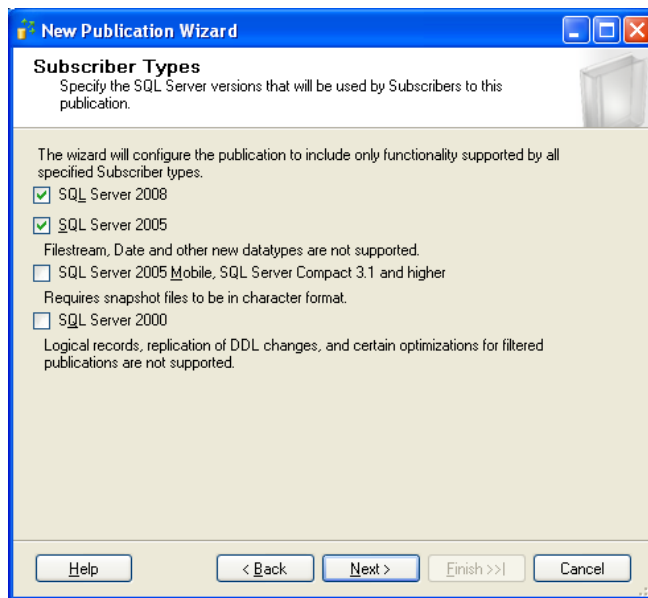


11. Select the desired **Replication Type**. We recommend a **merge publication**, because this type makes independent data updating possible from client and server.

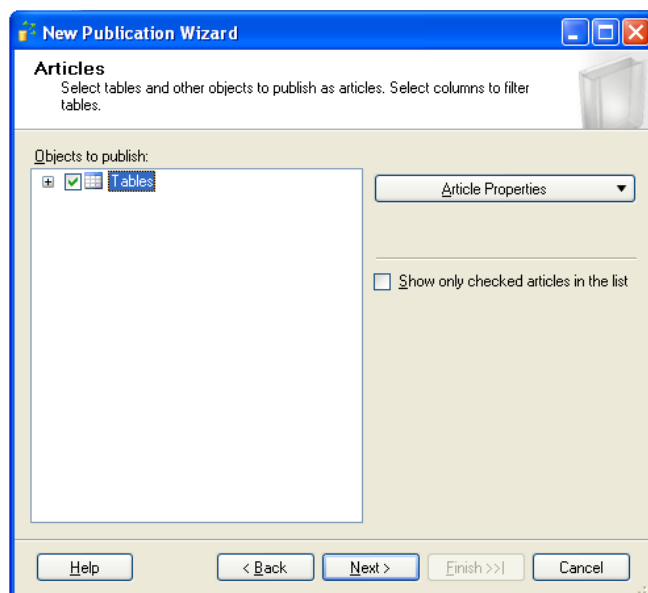
12. Click on **Next**



13. The window **Subscriber Types** appears

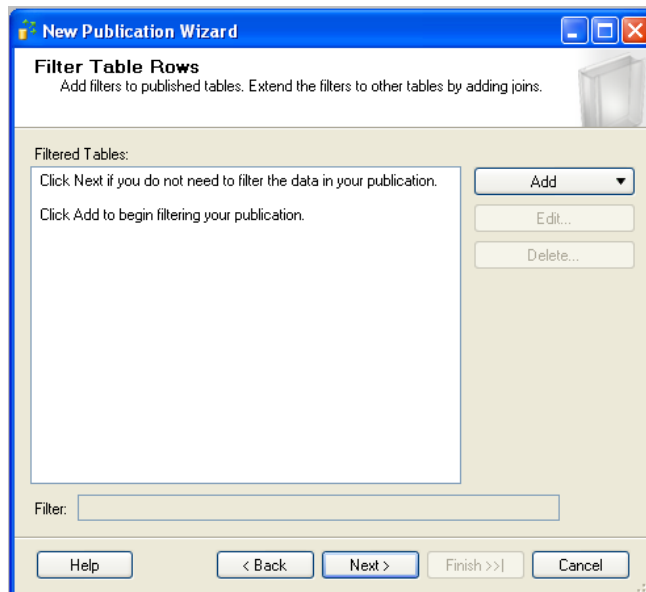


14. Don't activate SQL Server 2000 and SQL Server Mobile here. Otherwise database updates cannot be performed. Click on **Next**
15. In the window **Articles** you specify the tables, which should be send to the subscriber. Select the checkbox in front of **Tables**, in order to replicate all tables:

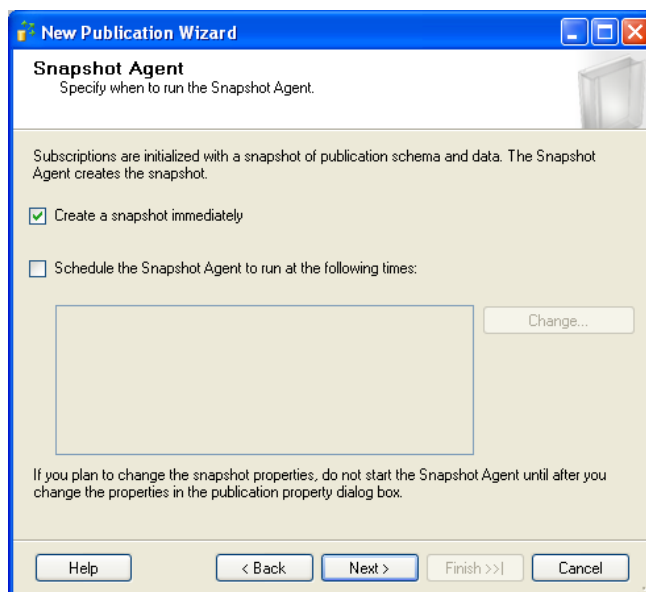


16. Click on **Next** afterwards

17. The window **Filter Table Rows** appears. All lines should be published. Therefore click on **Next**

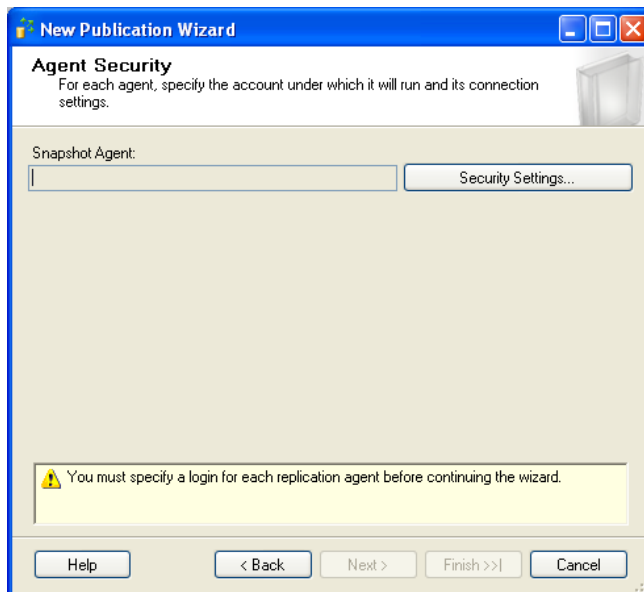


18. Specify the base snapshot of the data for immediate or after a schedule creation in the **Snapshot Agent** window

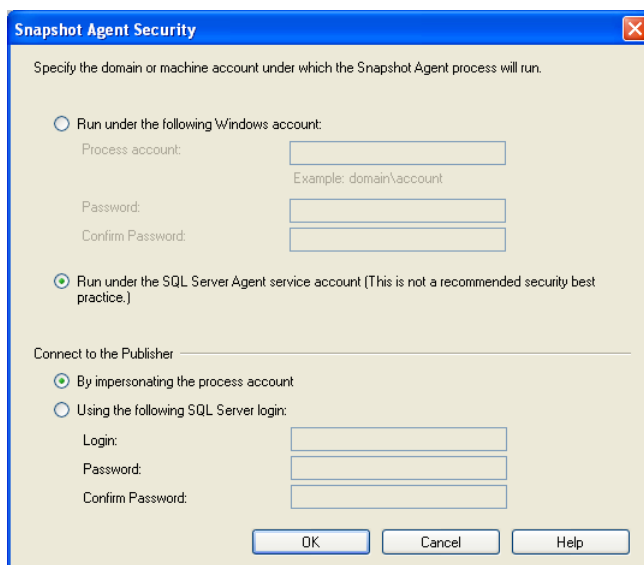


19. Click on **Next**

20. In the next window **Agent Security** specify through security settings, which account the agent will use for communication with the servers or computers.

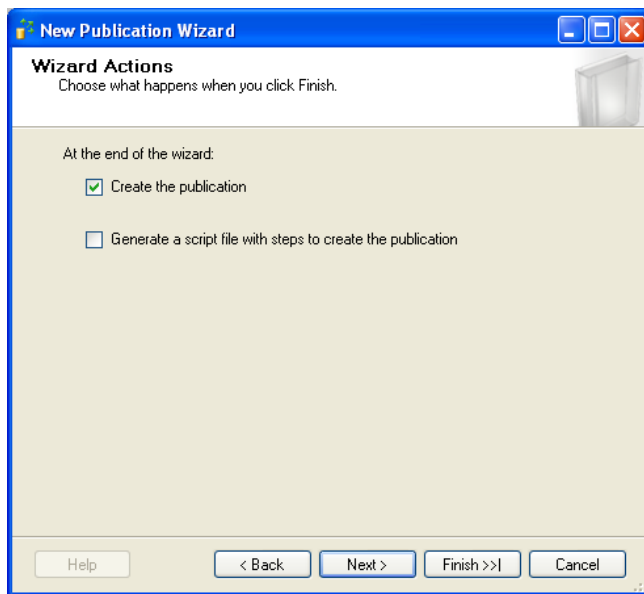


21. With the button **Security Settings...** you can specify a Windows or an SQL Server account. If you have created an account for the replication procedure, you can specify it here. Please remind that it needs access to **all systems** (Servers/all clients with off-line availability) as well as the possibility to work on all system



22. Click on **OK** afterwards

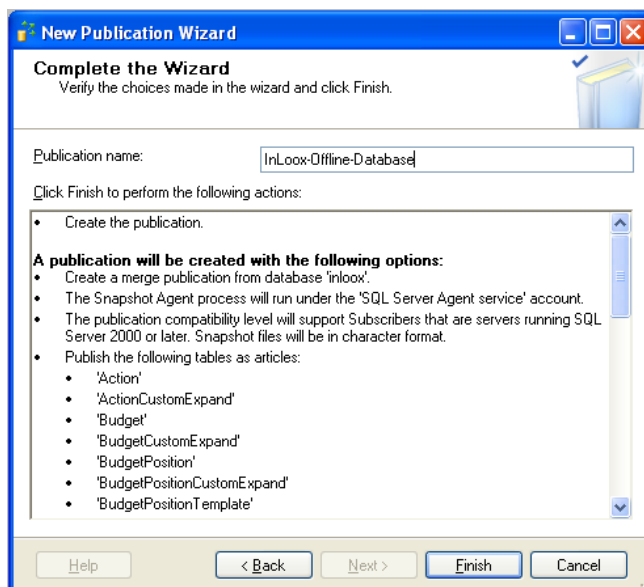
23. The window **Wizard Actions** appears



24. Click on **Next**

25. The window **Complete the Wizard** appears

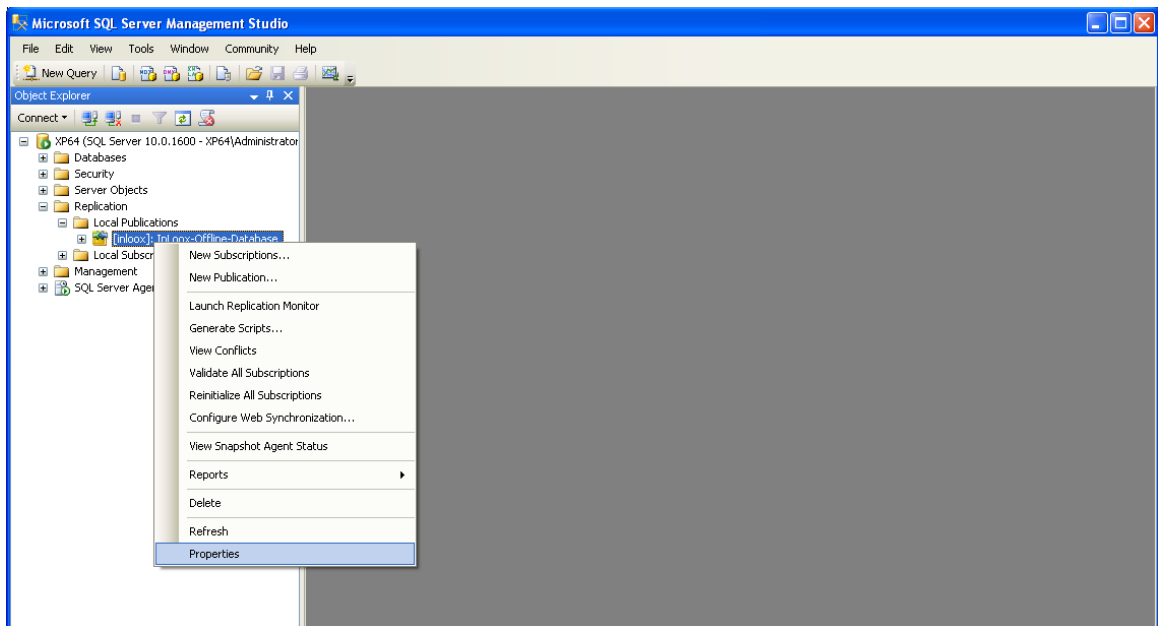
26. Enter a **Publication name**. It has been proved of value to use the database name



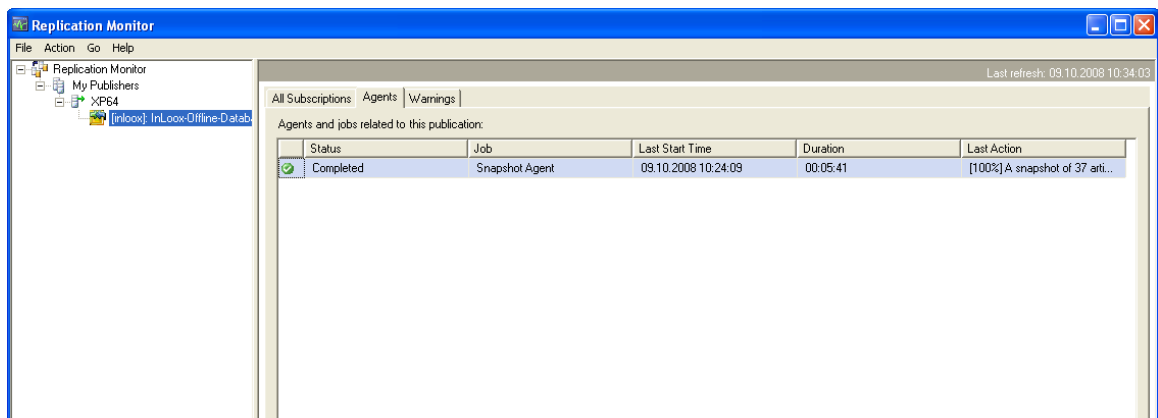
27. Click on **Finish** and the publication will be created

28. In the end a message is shown that the publication was crested successfully

29. You can examine your settings again, by pressing the **F5** key for updating the **Object Explorer**, extending the node **Replication >> Local Publications** and selecting **Properties**:



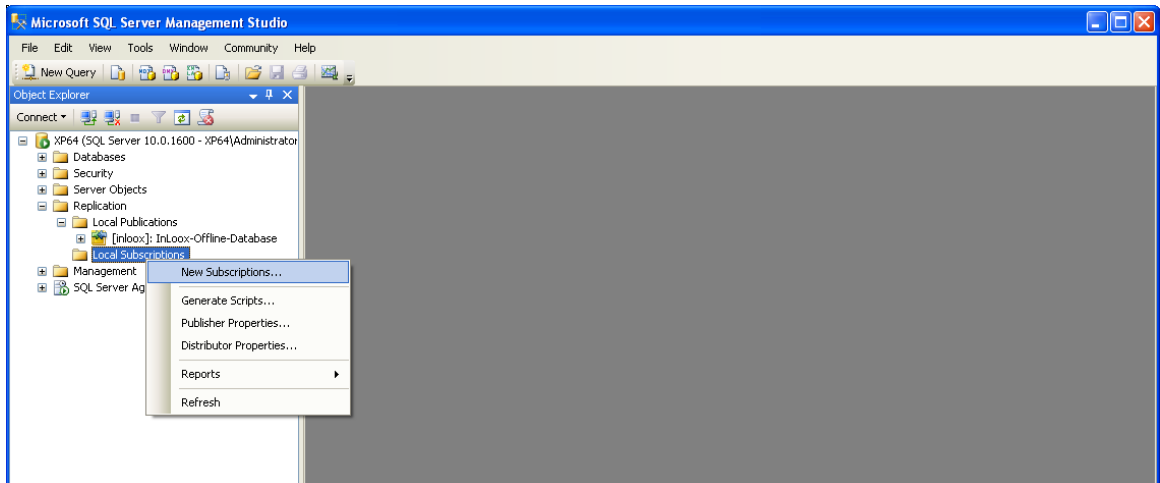
30. Use the **Replication Monitor** (available under the node **Management**), in order to examine whether errors occurred:



## Central subscription configuration on the server

You have created a distributor, a publisher as well as an article and a publication. Now you still have to create another subscriber and a schedule.

1. **Start SQL Server Management Studio**
2. Connect with the SQL Server, which was configured as publisher
3. Extend the **Replication** node
4. Right Click on **Local Subscriptions** in the **Object Explorer**

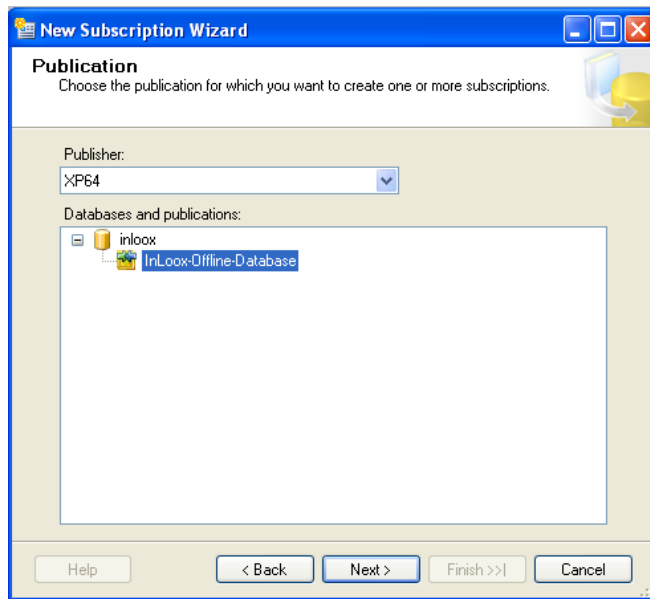


5. Click on **New Subscription...**  
Afterwards the **New Subscription Wizard** appears:



6. Click on **Next**

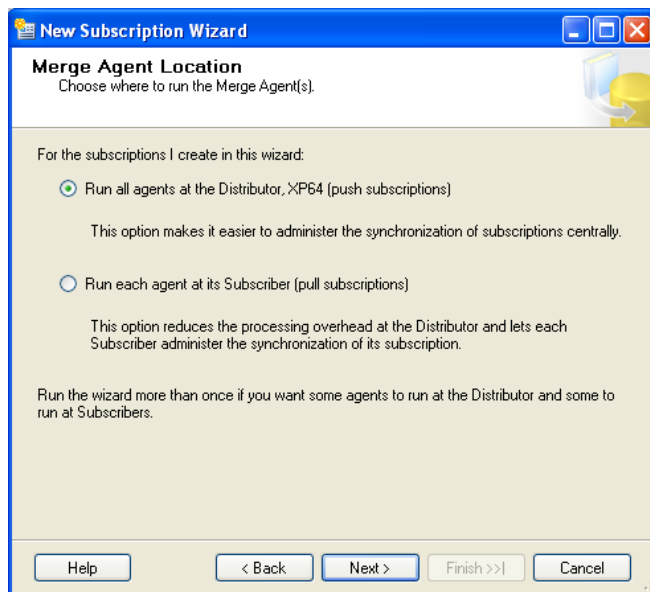
7. The window **Publication** appears



8. Select the appropriate Server (publisher) and the appropriate database

9. Click on **Next**

10. The **Merge Agent Location** window appears



Now you have to decide which way of publication you want to use. You have the choice between push and pull subscription.

**Subscription type comparison**

	Push subscription	Pull subscription
Advantages	<ul style="list-style-type: none"><li>- The agent runs on the server – no additional load on the client</li><li>- Data changes are pushed to the subscribers as soon as changes occur.</li><li>- Better security settings</li><li>- Manageable from the server</li></ul>	<ul style="list-style-type: none"><li>- Agent runs on the client - Less load on the server with many clients</li><li>- Sync status visible on the client</li></ul>
Disadvantages	<ul style="list-style-type: none"><li>- Higher load on the server with many clients</li><li>- More communication traffic</li></ul>	<ul style="list-style-type: none"><li>- Currently bad security concept (sysadmin role necessary)</li><li>- Sync cannot be started from server</li></ul>

We suggest using the push subscription.



## General Prerequisites for subscriptions:

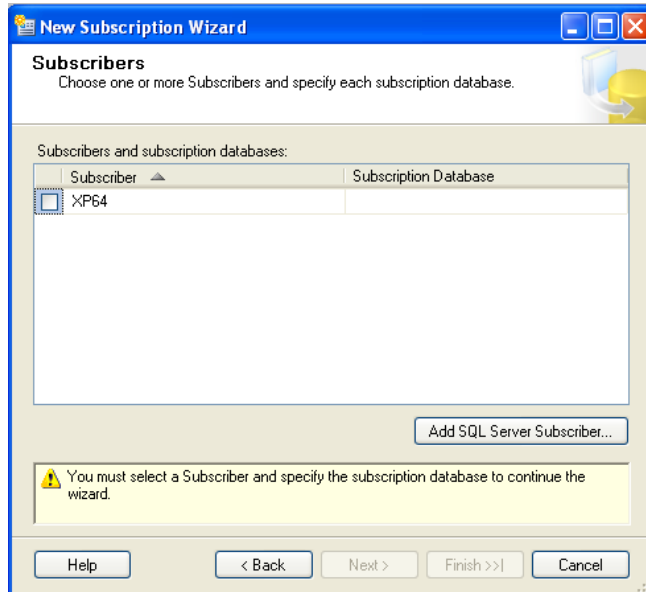
Verify that the client is reachable from the server.

1. The client pc and the installed version of SQL Server Express have to be switched on and is connected to the network.
2. Go to the Windows command prompt and perform a ping  
*ping ClientPCName*
3. If the PC does not answer check your firewall (Note: Windows 7 suppresses ping request by default)
4. Open the SQL Management Studio on the Server
5. Click on Open Connection
6. Fill in the details for the client database:  
Server: ClientPCName\SQLEXPRESS  
SQL Authentication with username and password.
7. You should be able to connect to the client database

## Alternative 1: Create a push subscription

A subscription can be created from the main server side and from the client side. The steps on the client side are very similar. To simplify matters, the following guide for push subscription will describe the steps from the server side, the pull subscription will describe it from the client side.

1. **Select whether the merge agent should be run at the distributor or at the subscriber. Select push subscription here**
2. Click on **Next** and the **Subscribers** window appears with a selection of the subscribers and subscription databases.

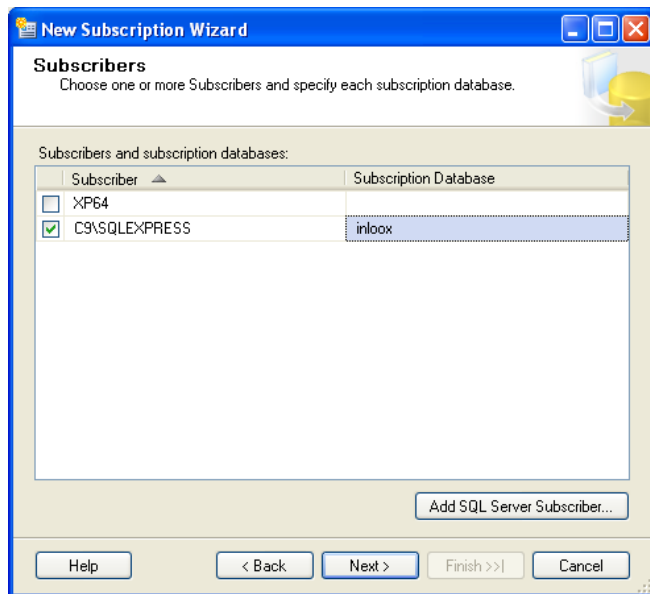


3. Click on **Add SQL Server Subscriber...** and connect with the appropriate subscriber, in this case with your client or your notebook.

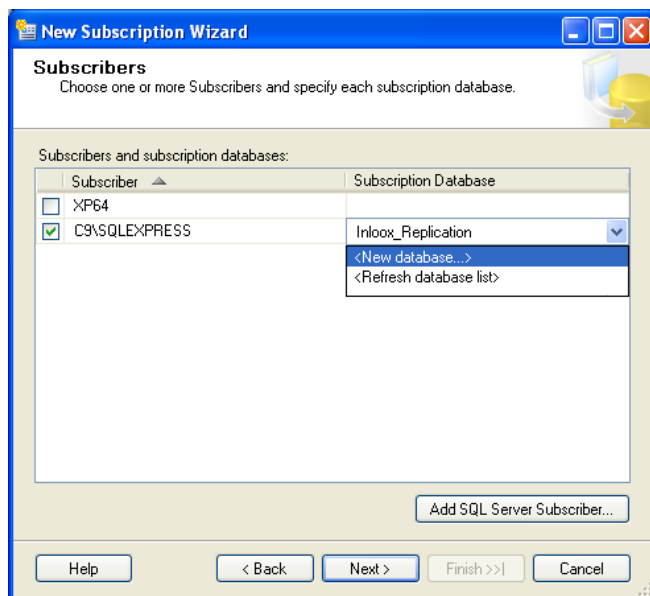


4. Click on **Connect**

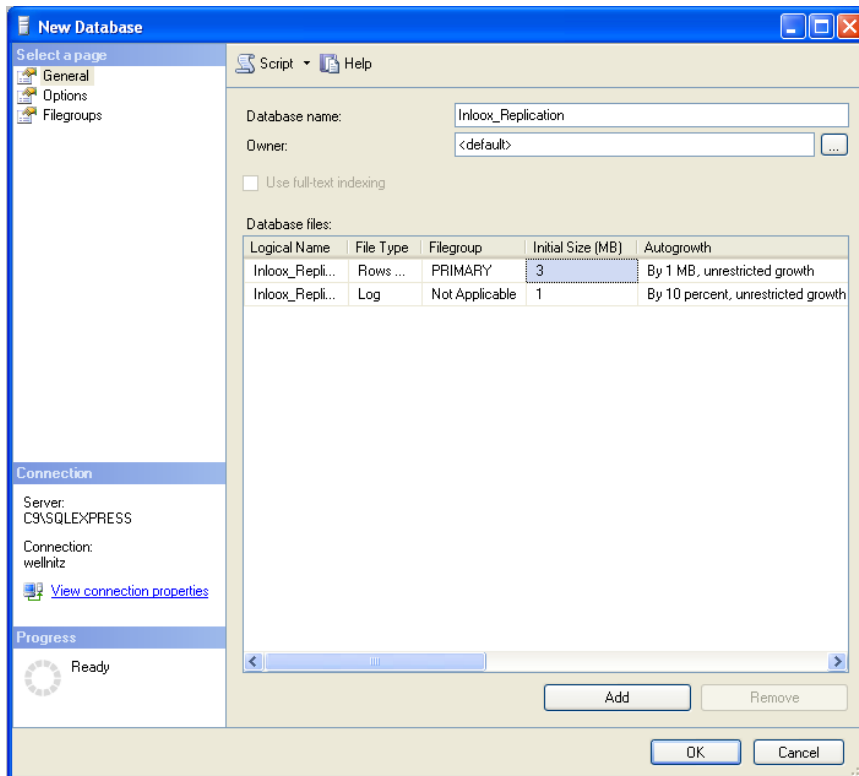
5. The **Subscribers** window appears again
6. Open the list under the column **Subscription Database**



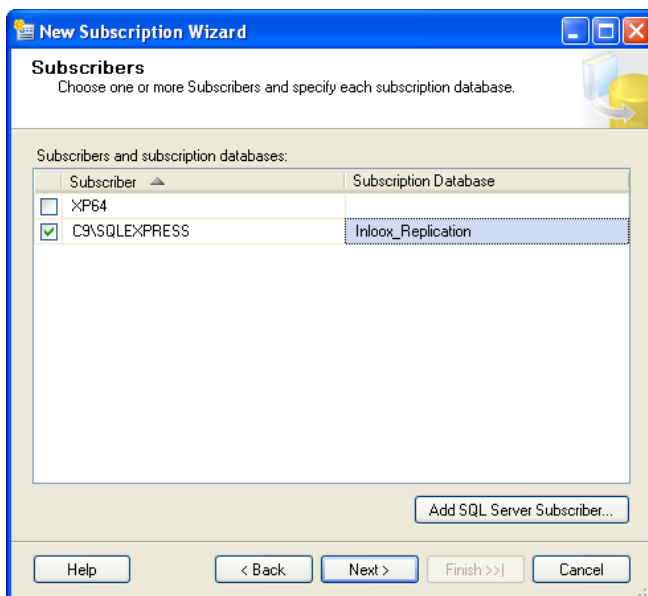
7. Select the **New Database...** entry



8. Assign a database name in the window **New Database** e.g. **InLoox\_Replication**

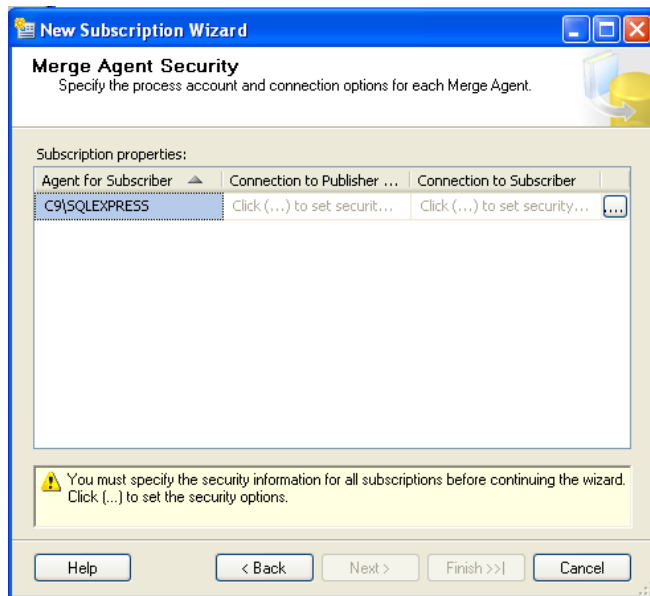


9. Click on **OK**
10. The **Subscriber** window appears again, but this time with the selected database **InLoox\_Replication**

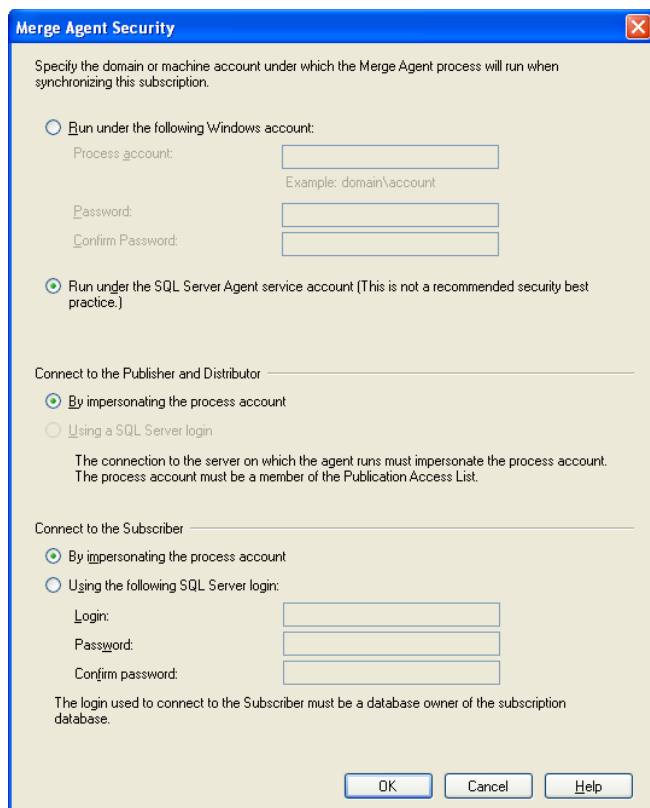


11. Click on **Next**

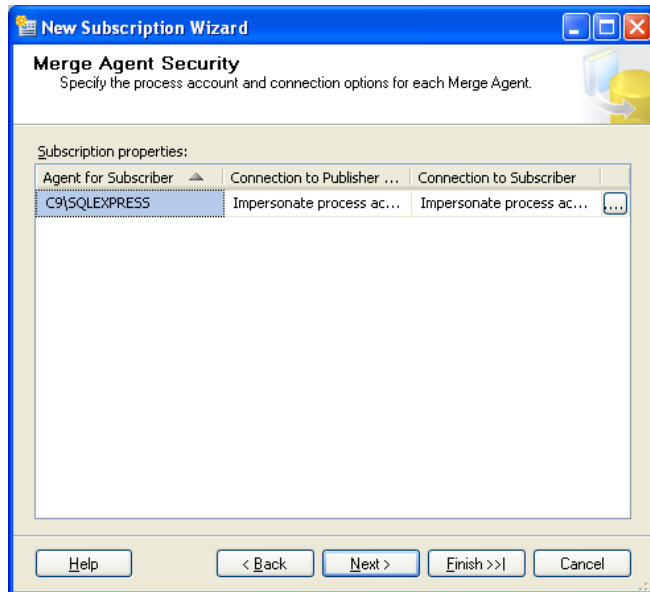
12. The **Merge Agent Security** window appears



13. You have to set the security information for all subscription before the wizard can be continued. Click on (...), in order to configure the security settings.
14. Select the **Run under the SQL Server Agent service account (This is not a recommended security best practice)** option in the **Merge Agent Security** window



15. Click on **OK**
16. The following inputs will appear in the **Merge Agent Security** window

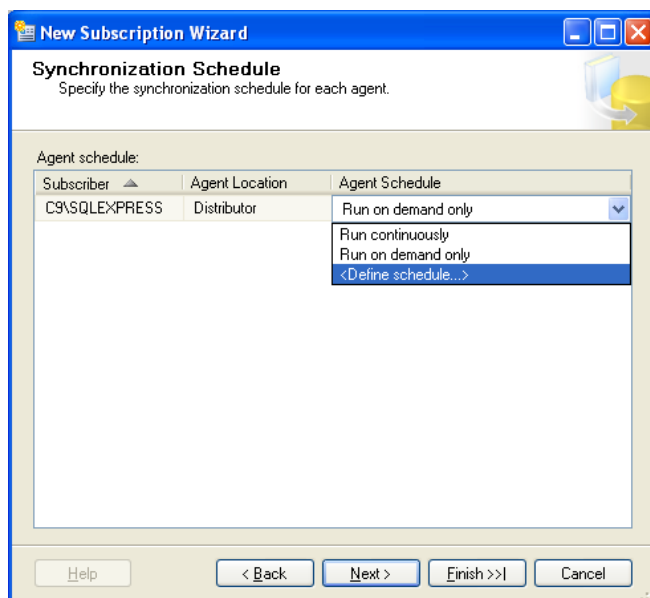


17. Click on **Next**

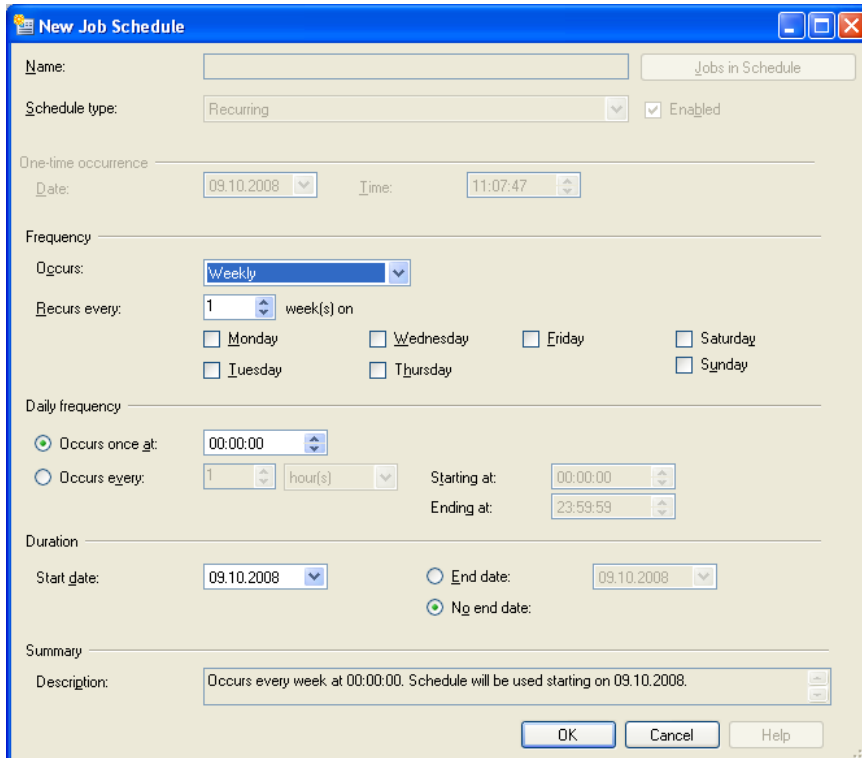
The **Synchronization Schedule** window appears. Choose an option for the synchronization.

Type	Meaning	Pro/Contra
Run continuously	The SQL-Agent checks every 60 seconds the connection to the clients and if changes have to be synchronized	Pro: Changes are nearly immediately synched  Contra: Higher network traffic, especially with many clients
Run on demand only	The databases are replicated, but are only synchronized if it is manually started on the server	Pro: Only small amount of traffic  Contra: Not useful for frequent changes. Has to be started by an admin on the main server
Define schedule	The synchronization is started as defined in the schedule.  You can choose days, times and frequency.	Pro: Synchronization can be configured for the individual need.  Contra: Has to be configured

We recommend using continuously or defining schedule (e.g. every 10 minutes) for InLoox.

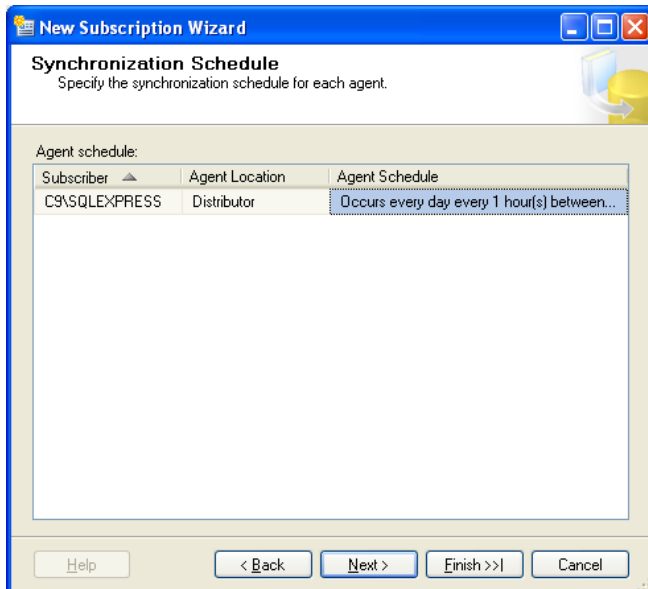


18. The **New Job Schedule** window appears automatically



The **New Job Schedule** dialog box is shown. It has a blue title bar with standard window controls. The main area is light beige. At the top, there's a 'Name' field and a 'Jobs in Schedule' button. Below that, 'Schedule type' is set to 'Recurring' with a dropdown arrow, and 'Enabled' is checked. A section for 'One-time occurrence' has 'Date' set to '09.10.2008' and 'Time' set to '11:07:47'. The 'Frequency' section shows 'Occurs' set to 'Weekly' with a dropdown, and 'Recur every' set to '1' week(s) on. Below this are checkboxes for days of the week: Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, and Sunday. The 'Daily frequency' section has two options: 'Occurs once at' (selected) with a time set to '00:00:00', and 'Occurs every' with '1' hour(s). It also has 'Starting at' and 'Ending at' time fields. The 'Duration' section has 'Start date' set to '09.10.2008', and options for 'End date' (set to '09.10.2008') or 'No end date' (selected). A 'Summary' section at the bottom has a 'Description' field containing the text: 'Occurs every week at 00:00:00. Schedule will be used starting on 09.10.2008.' At the bottom right are 'OK', 'Cancel', and 'Help' buttons.

19. Enter the desired data and dates, e.g. all 5 min. daily. Pay attention on your network load, particularly with a large number of users. Click on **OK** thereafter
20. The **Synchronization Schedule** window appears again, filled with the selected data under the **Agent Schedule** column.

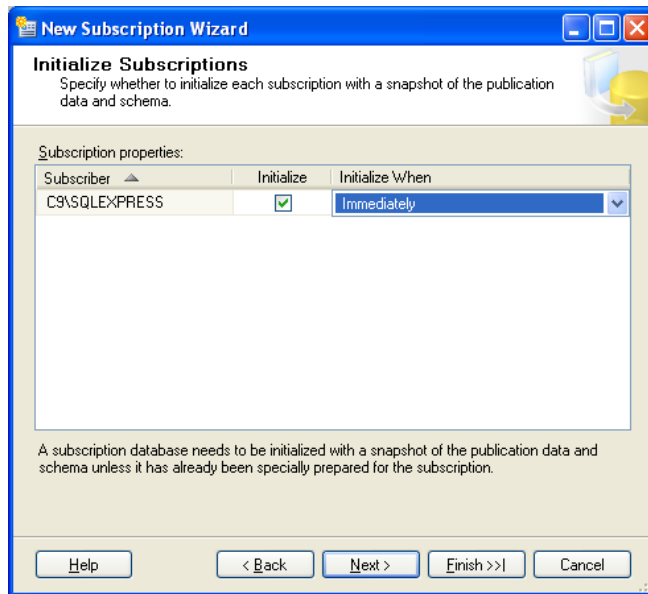


The **New Subscription Wizard - Synchronization Schedule** dialog box is shown. It has a blue title bar. The main area is light beige. At the top, it says 'Specify the synchronization schedule for each agent.' Below this is a table with three columns: 'Subscriber', 'Agent Location', and 'Agent Schedule'. The first row shows 'C9\SQLXPRESS' under 'Subscriber', 'Distributor' under 'Agent Location', and 'Occurs every day every 1 hour(s) between...' under 'Agent Schedule'. Below the table is a large empty text area. At the bottom are buttons for '< Back', 'Next >', 'Finish >>', 'Cancel', and a 'Help' button on the left.

21. Click on **Next**



22. The **Initialize Subscriptions** window appears. Change the **initialization time** to **Immediately**



23. Click on **Next**

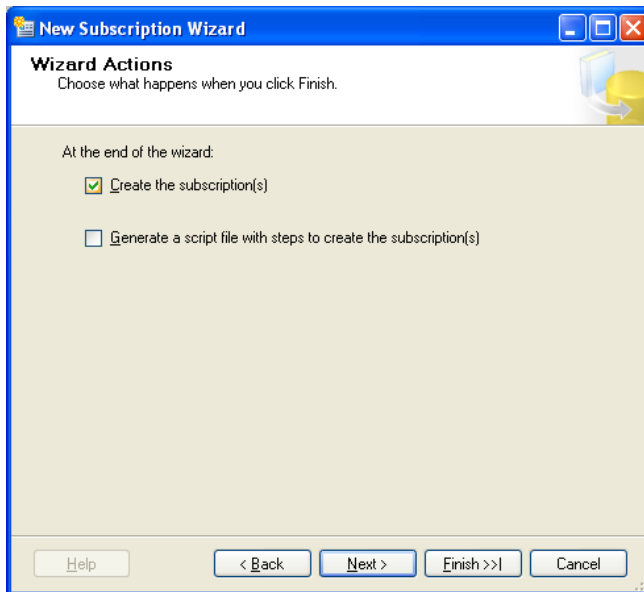
24. The **Subscription Type** window appears



25. Here you determine, what happens to a conflict, whether the **client** or the **server** gets priority. Here select e.g. **First to Publisher wins**

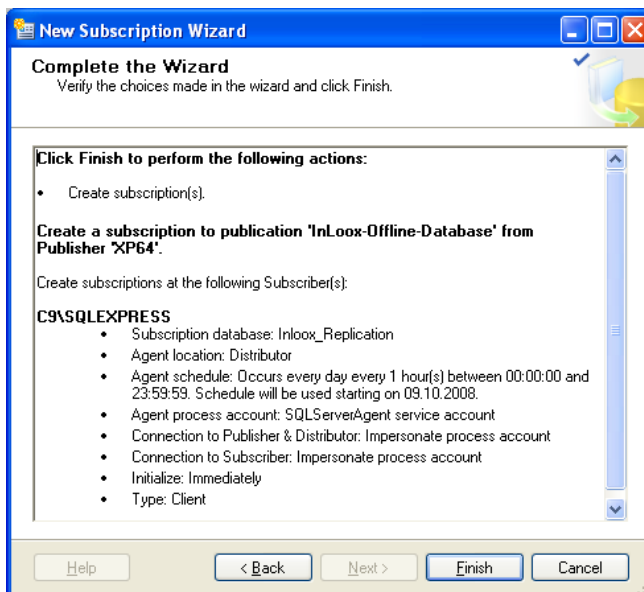
26. Click on **Next**

27. The **Wizard Actions** window appears



28. Click on **Next**

29. The **Complete the Wizard** window appears

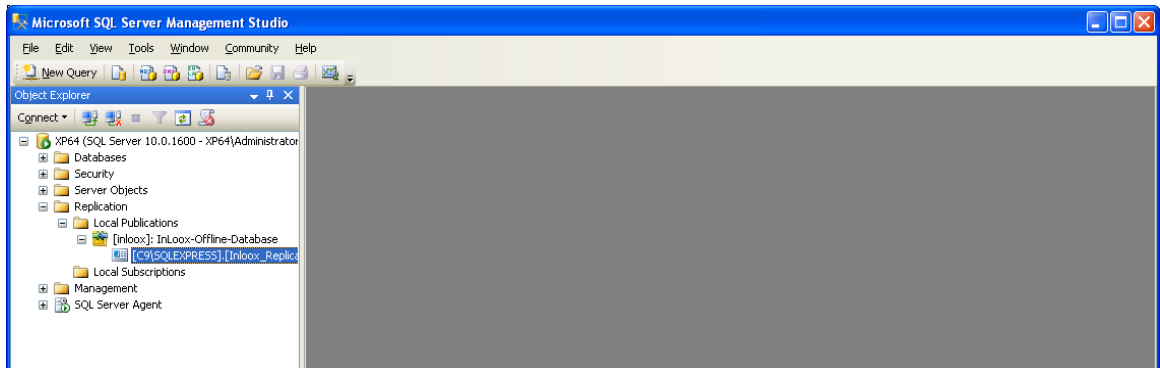


30. Click on **Finish**. The subscription is created.

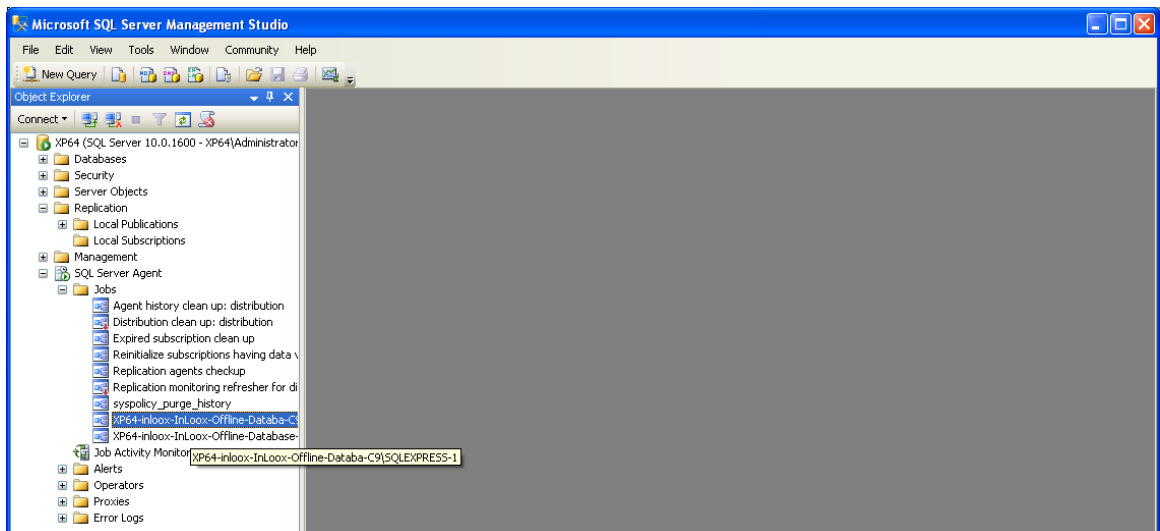
31. At the end click on **Close**

Verify the Subscription by performing the following steps:

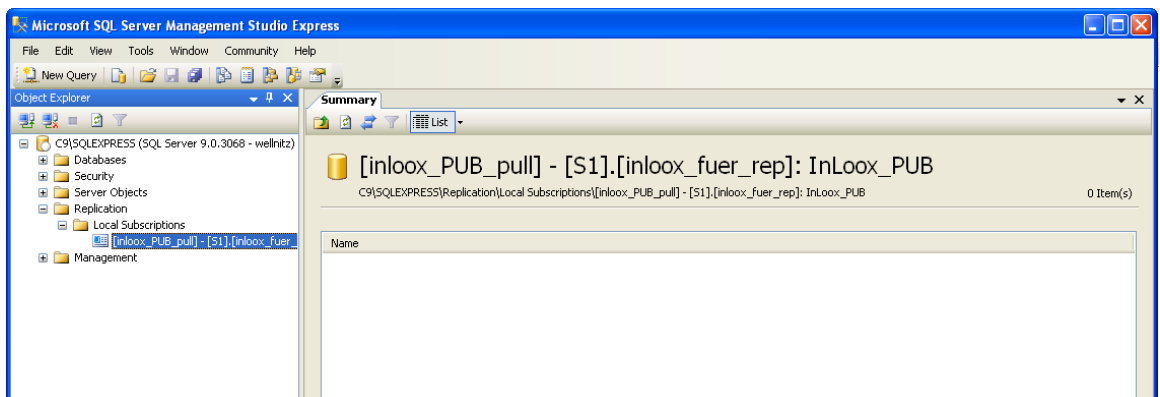
32. You can see the created Replication under the **Replication >> Local Publications** node



33. You can see the first created job under the **SQL Server Agent >> Jobs** node



34. You see, that a local subscription was created on the client under the **Replication >> Local subscription** node. Perhaps you have to press **F5** for updating the list.



## Alternative 2: Create a pull subscription

The following guide will describe the creation of the subscription from the client side. This task can also be performed in similar steps from the Server side.

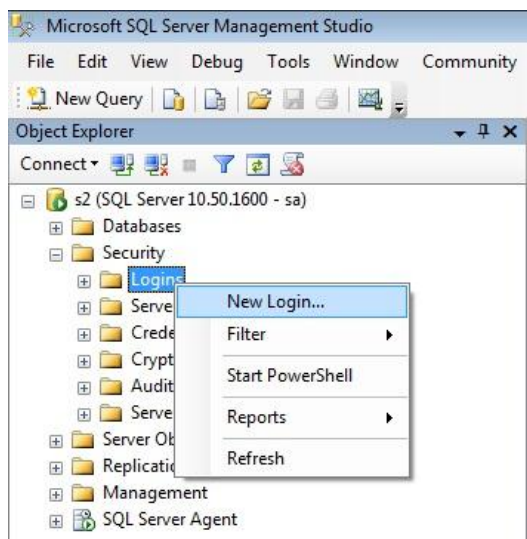
First create a user account on the main database and give appropriate permissions to this account. The account has to use a domain Windows logon account with integrated security. This permission role “sysadmin” has to be set for this account.

This setting is the main disadvantage of the pull subscription from a security point of view, since it allows the user to do nearly everything.

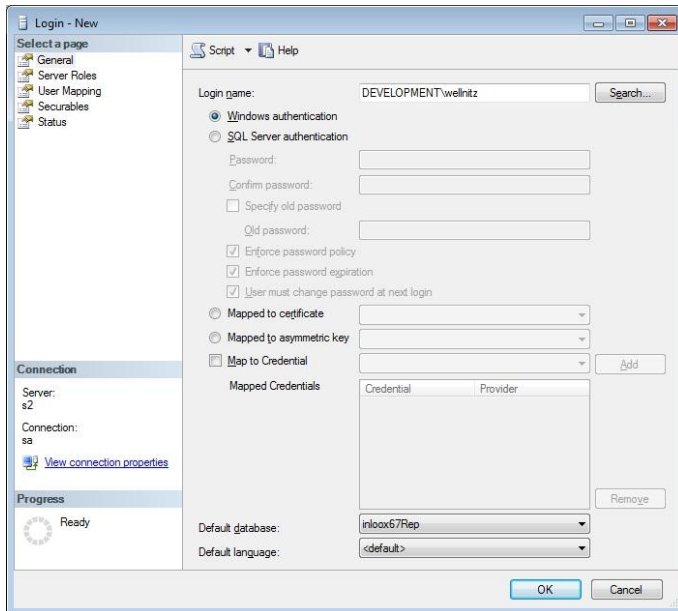
1. Connect to the main server Database with an account which has all privileges.



2. Expand the “Security” node and right click on “Logins”. Choose “New Login...”



3. Search the domain login for the user and choose “Windows authentication”. Select the replication Database as “Default database”.



Login - New

Select a page

- General
- Server Roles
- User Mapping
- Securables
- Status

Connection

Server: s2

Connection: sa

[View connection properties](#)

Progress

Ready

Script Help

Login name: DEVELOPMENT\wellnitz Search...

☒ Windows authentication

☐ SQL Server authentication

Password:

Confirm password:

☐ Specify old password

Old password:

☒ Enforce password policy

☒ Enforce password expiration

☒ User must change password at next login

☐ Mapped to certificate

☐ Mapped to asymmetric key

☐ Map to Credential

Map to Credential Add

Mapped Credentials

Credential	Provider
------------	----------

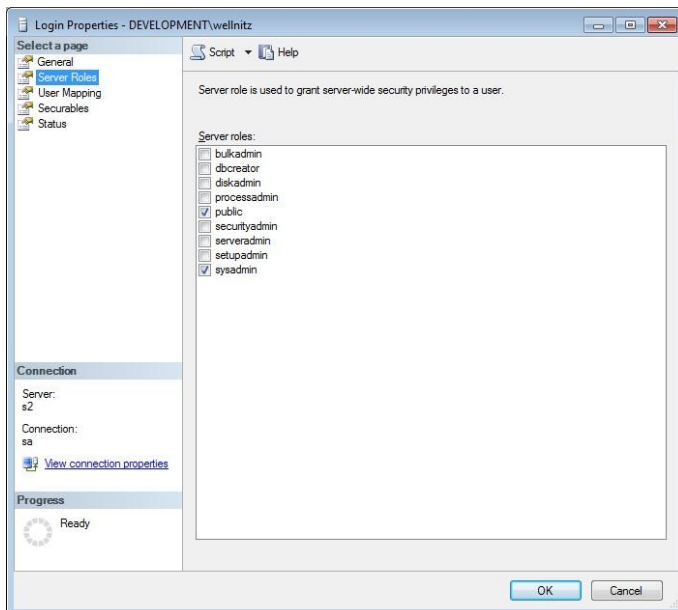
Remove

Default database: inloox67Rep

Default language: <default>

OK Cancel

4. Switch to “Server Roles” and check “sysadmin”



Login Properties - DEVELOPMENT\wellnitz

Select a page

- General
- Server Roles
- User Mapping
- Securables
- Status

Connection

Server: s2

Connection: sa

[View connection properties](#)

Progress

Ready

Script Help

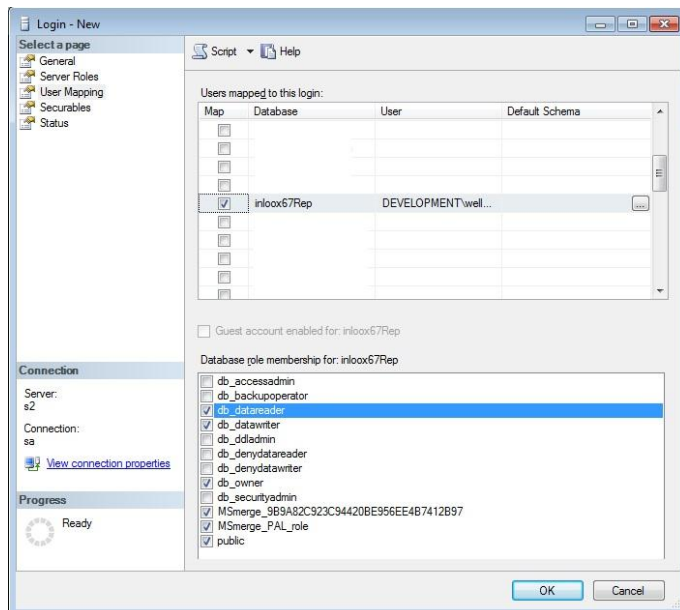
Server role is used to grant server-wide security privileges to a user.

Server roles:

- ☐ bulkadmin
- ☐ dbcreator
- ☐ diskadmin
- ☐ processadmin
- ☐ public
- ☒ securityadmin
- ☐ serveradmin
- ☐ setupadmin
- ☒ sysadmin

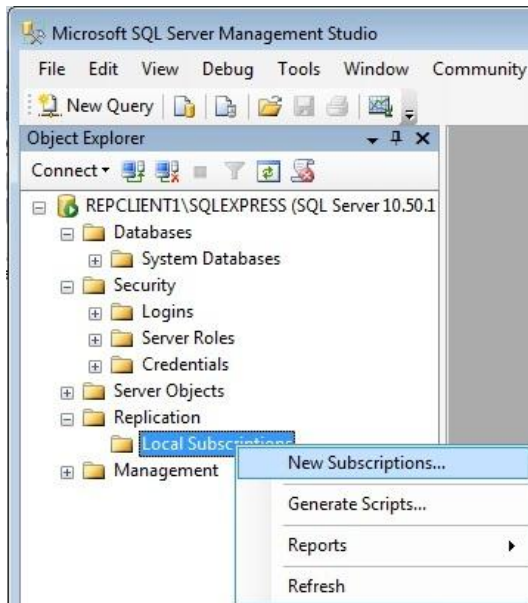
OK Cancel

5. Switch to “User Mapping” and select the replication database. Set the following role membership: db\_datareader, db\_datawriter, db\_owner and the MSmerge roles

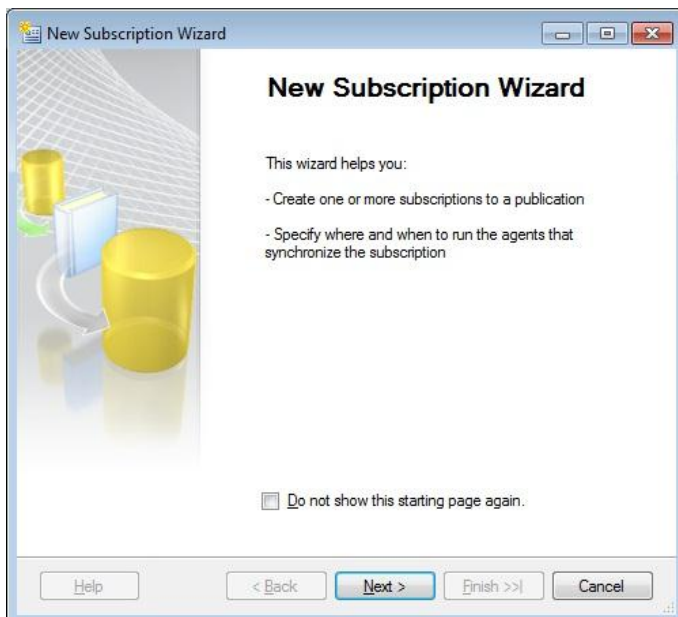


Now all account settings for this user are done on the main server. You can proceed with the subscription setup.

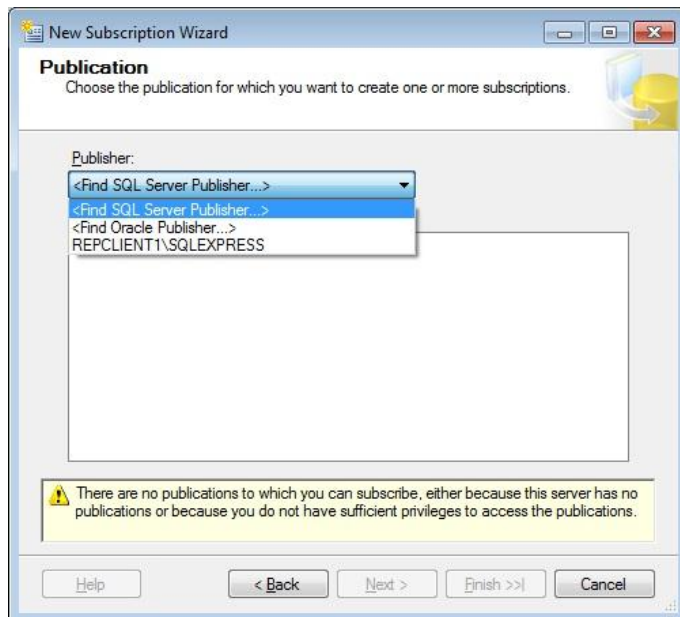
1. Open the SQL Management Studio on the client and connect to the installed SQL Express database.
2. Expand the node “Replication” and right click on “Local Subscription”. Click on “New Subscriptions”



3. The “New Subscription Wizard” is launched.



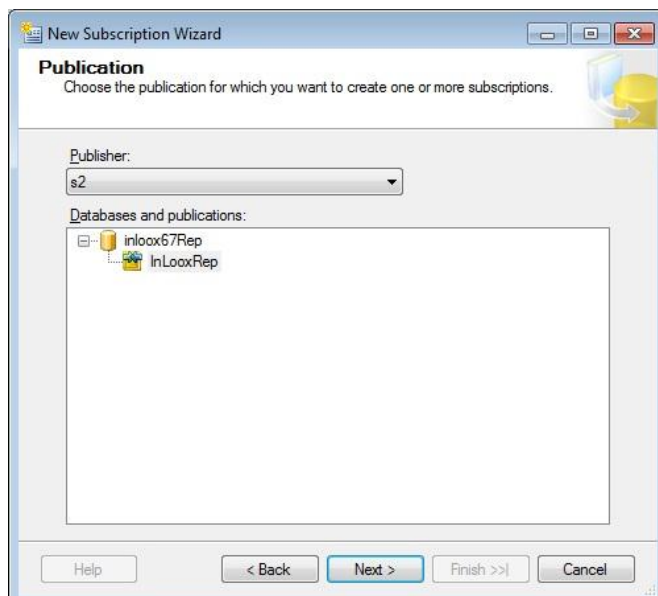
- Click on the drop down to search for the publisher database. Click on "<Find SQL Server Publisher...>".



- Fill in the server connection information. Make sure to use integrated security here.

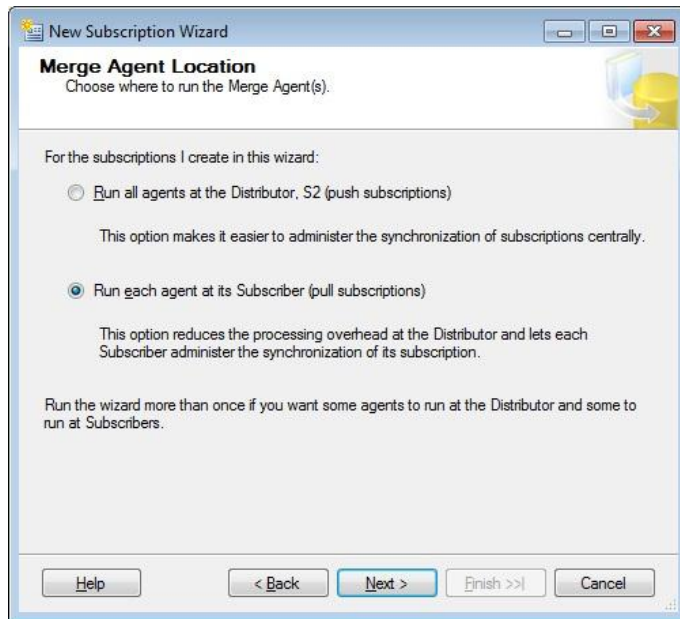


- Choose the publication and click on "Next".

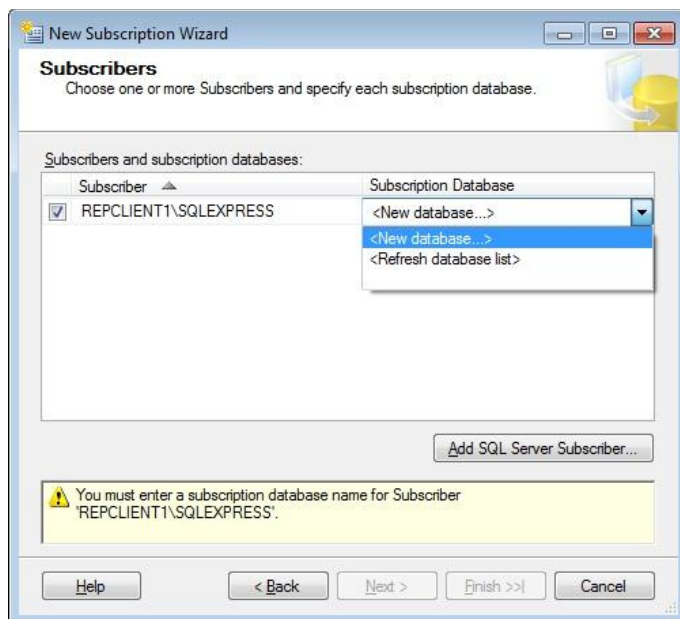




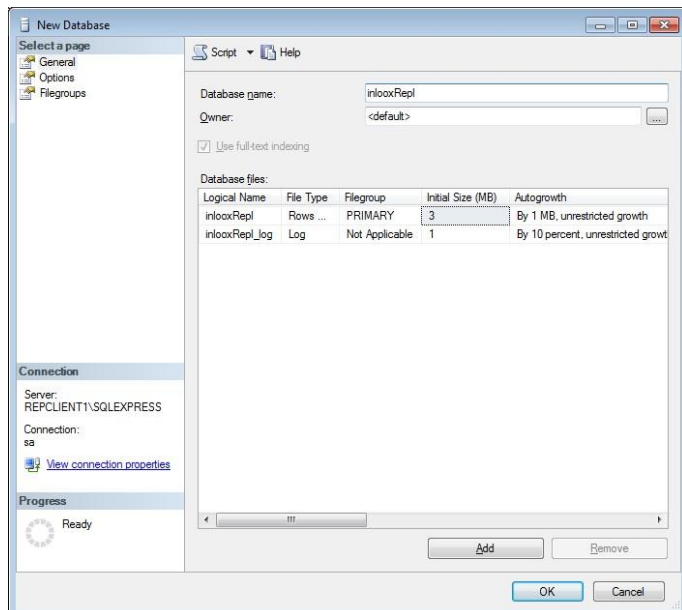
7. Choose "Run each agent at its subscriber (pull subscription)" and click on "Next".



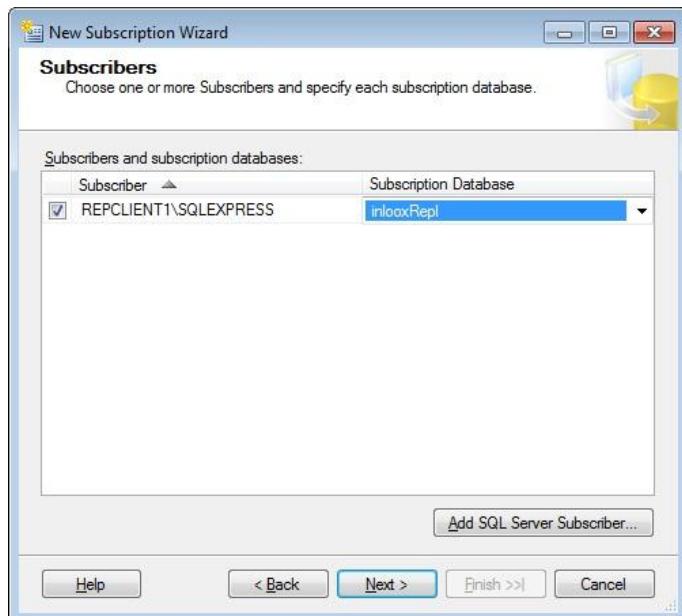
8. The local SQL Server is selected. Select "<New database...>" from the drop down.



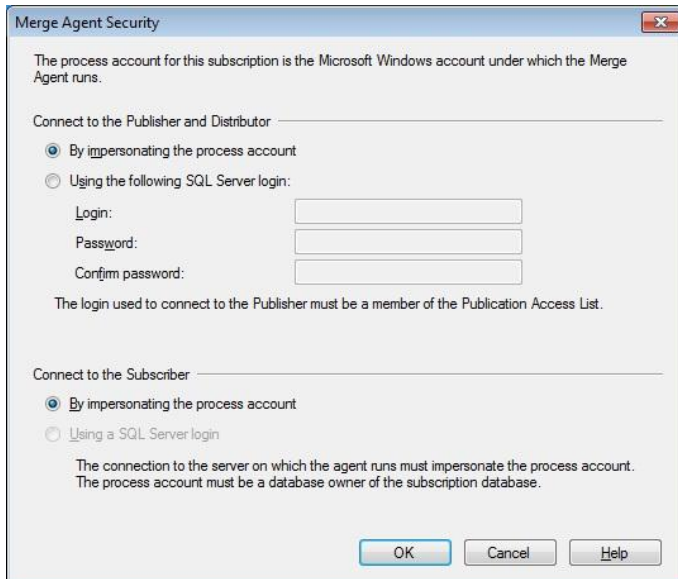
9. Fill in a database name. It is useful to have a name like “inlooxRepl” to directly show that it is a replicated database. Click “OK”.



10. The newly created database is automatically selected. Click on “Next”.



11. Now you have to set the security for the merge agent. Although you have the possibility to fill in “SQL Server login” this does currently not work and is a known issue. You have to choose “By impersonating the process account”. Click “OK”.



The process account for this subscription is the Microsoft Windows account under which the Merge Agent runs.

Connect to the Publisher and Distributor

☒ By impersonating the process account

☐ Using the following SQL Server login:

Login:

Password:

Confirm password:

The login used to connect to the Publisher must be a member of the Publication Access List.

Connect to the Subscriber

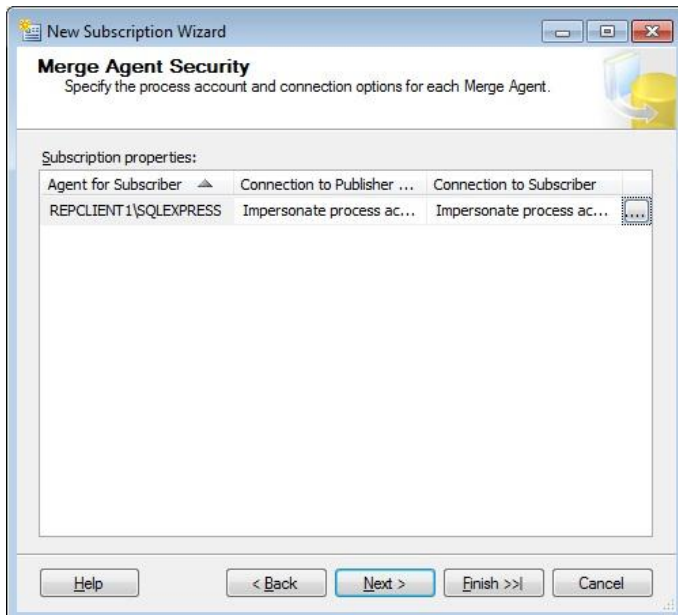
☒ By impersonating the process account

☐ Using a SQL Server login

The connection to the server on which the agent runs must impersonate the process account.  
The process account must be a database owner of the subscription database.

OK Cancel Help

12. Click “OK” on this summary screen.



New Subscription Wizard

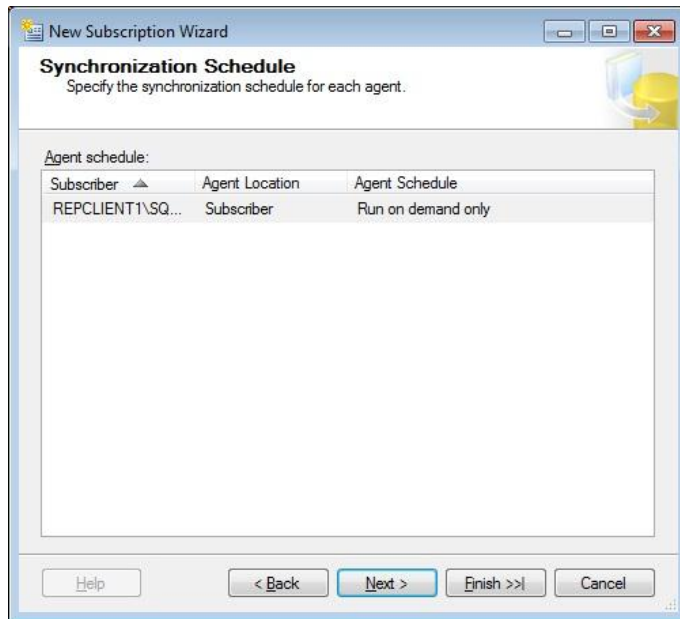
**Merge Agent Security**  
Specify the process account and connection options for each Merge Agent.

Subscription properties:

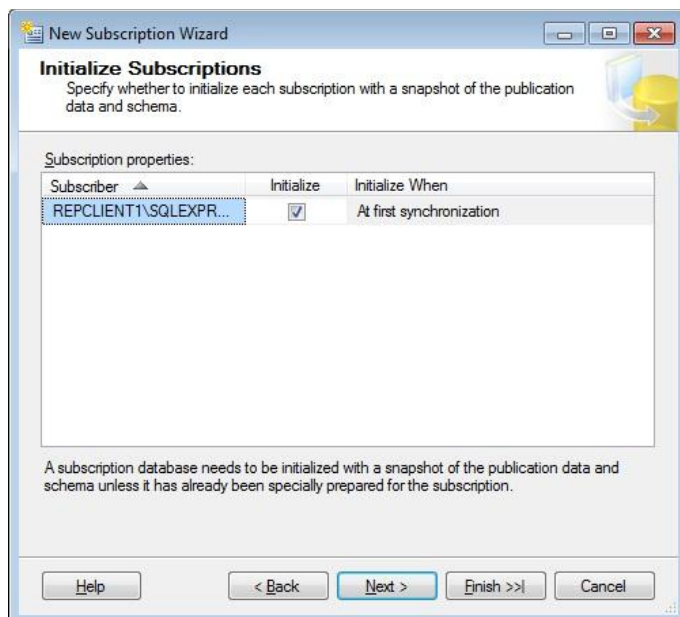
Agent for Subscriber	Connection to Publisher ...	Connection to Subscriber
REPLICENT1\SQLEXPRESS	Impersonate process ac...	Impersonate process ac...

Help < Back Next > Finish >> Cancel

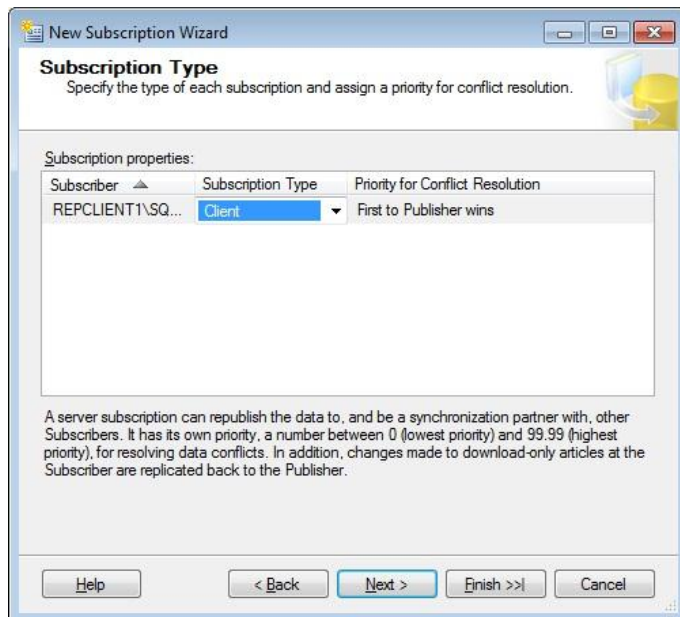
13. Just proceed with “OK”.



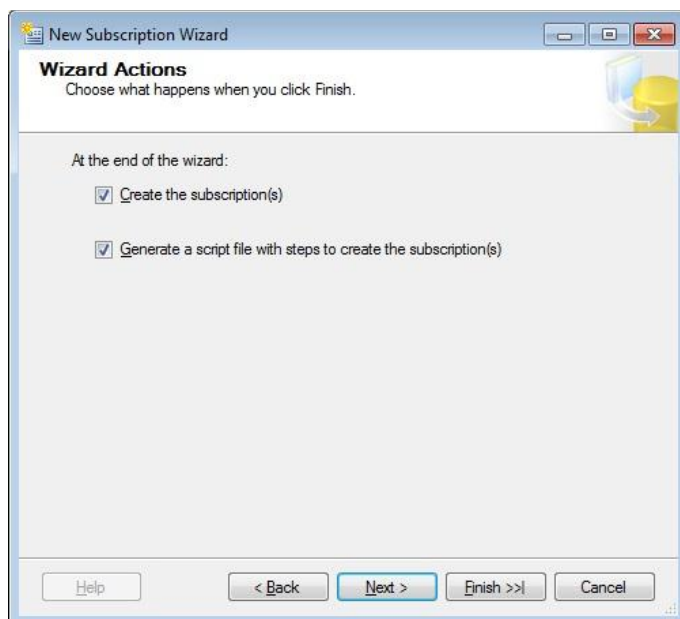
14. Just proceed with “OK”.



15. Specify the subscription type "Client" to setup what should happen if there are data conflicts. The first data on the main server will win.



16. Click on "Next" and on finish to create the subscription.



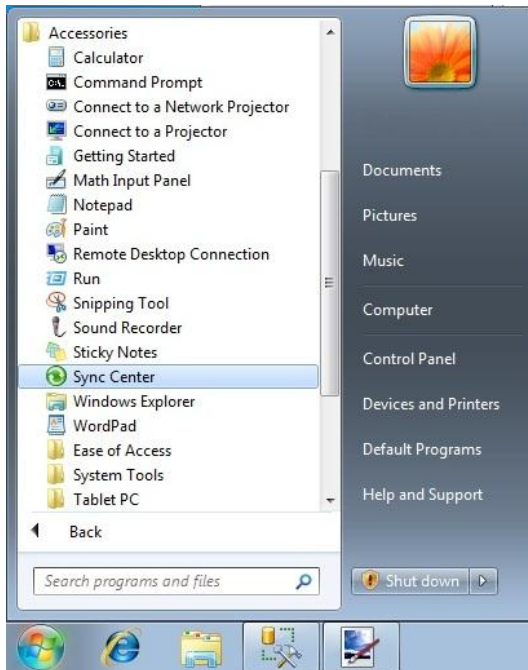
Now the general setup is finished and the synchronization has to be configured.

## Configure the synchronization for a pull subscription.

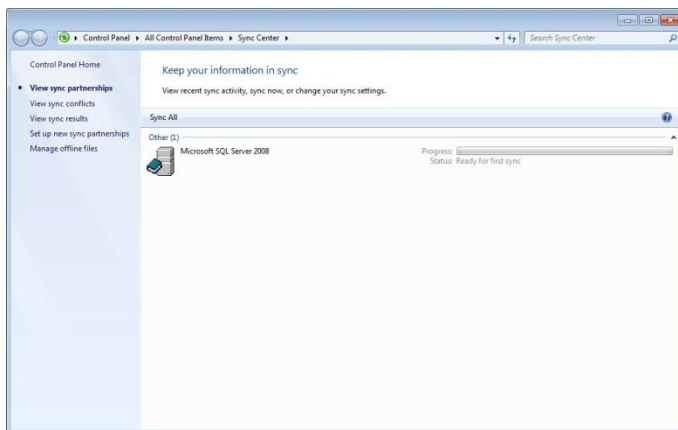
Microsoft SQL Express databases do not offer an SQL agent service like the full version. (SQL Server Express R2 installs this service but it is not executable).

With the Express editions the synchronization has to be performed by the “Microsoft Sync Center”.

1. You can find this tool via “Start”->”All Programs”->”Accessories”->”Sync Center”



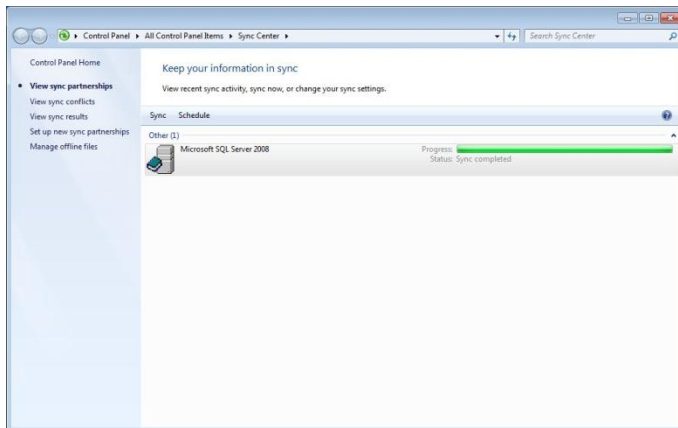
2. Open the “Sync Center” and you should see your SQL Server at “View sync partnerships”.



Please note that it might take a few moments till the entry appears in the list.

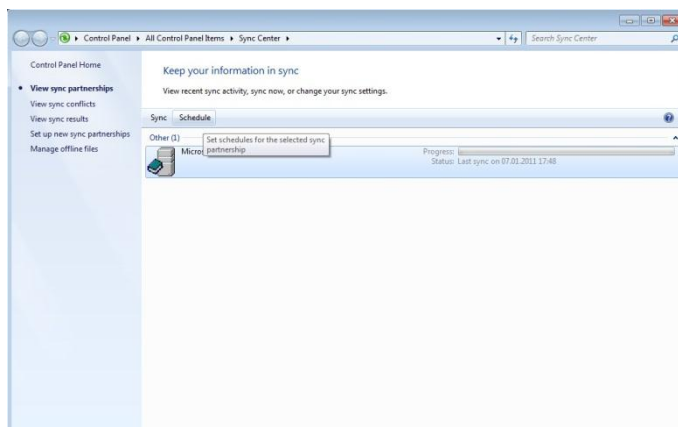
Close and reopen “Sync Center” if it does not appear after some minutes.

3. Click on “Sync”. If all login parameters are correctly configured it synchronizes with the main database. This might take several minutes since all tables have to be transferred.

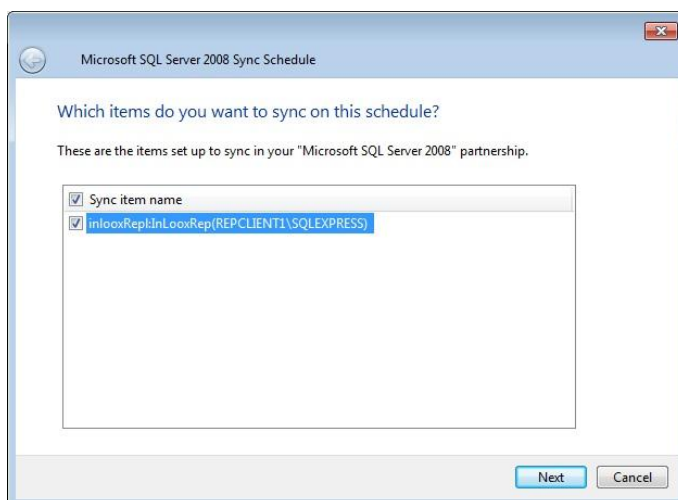


## Setup a sync schedule

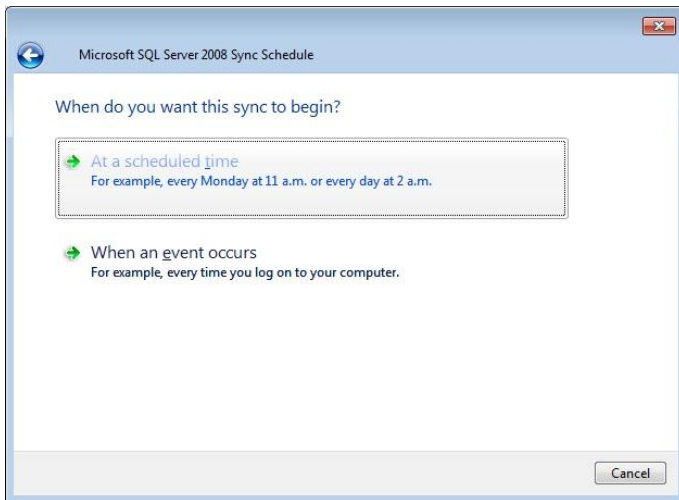
1. Click on “Schedule”.



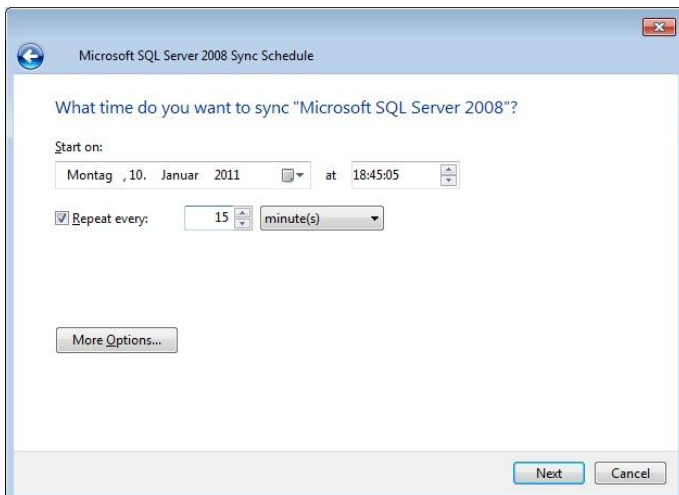
2. Choose your replication and click on “Next”.



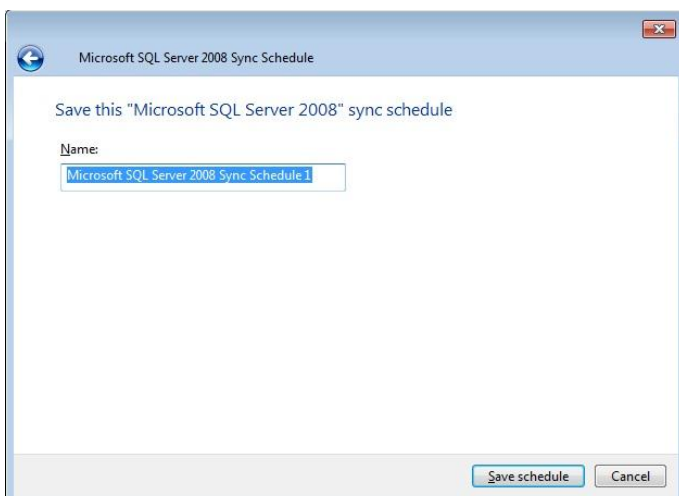
3. Click on "At a scheduled time".



4. Specify a schedule according to your needs. We suggest to start immediately and sync every 10 to 15 minutes. Syncing too often will put a lot of load on the client, too less might lead to conflicts and data loss.



5. Specify a name and save the schedule.

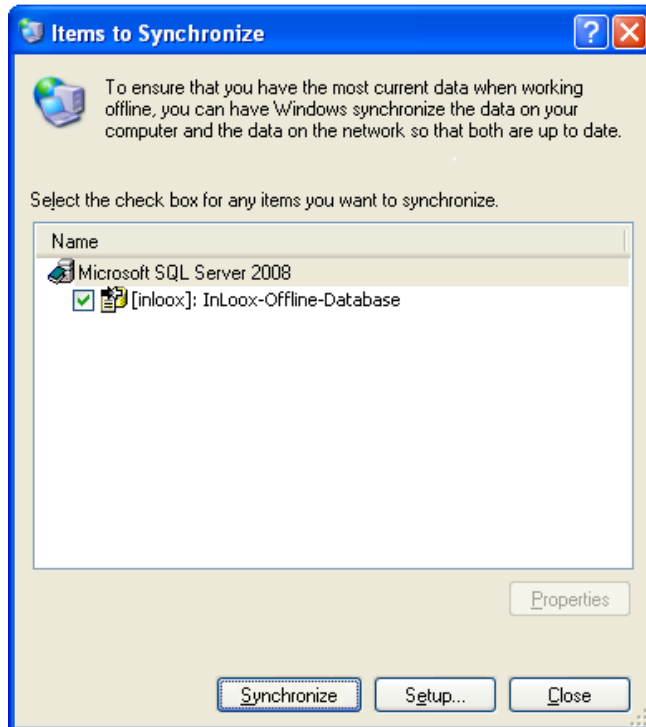




## Initialize a client created subscription

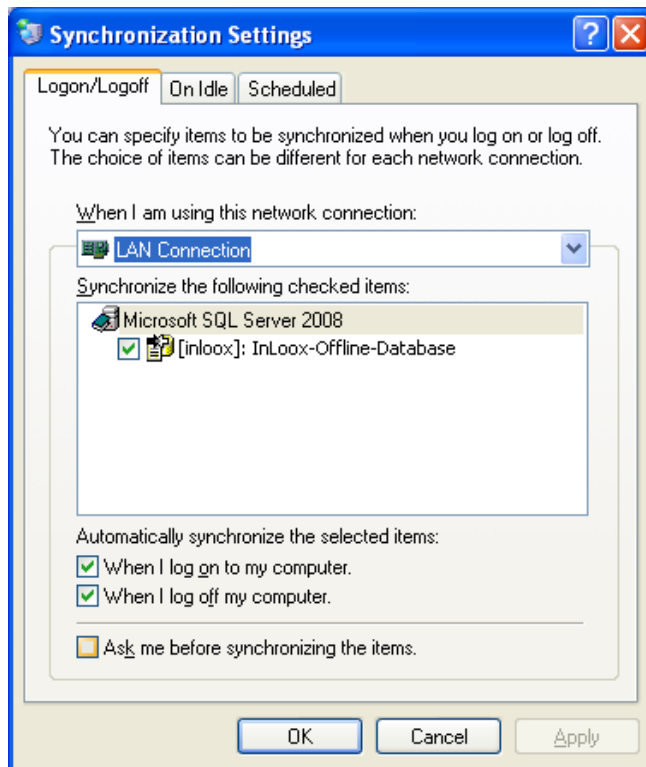
Proceed as follows:

1. Open the **Windows Explorer** on your Client
2. Click on **Synchronizing** in the **Extra** menu
3. The **Items to Synchronize** window appears

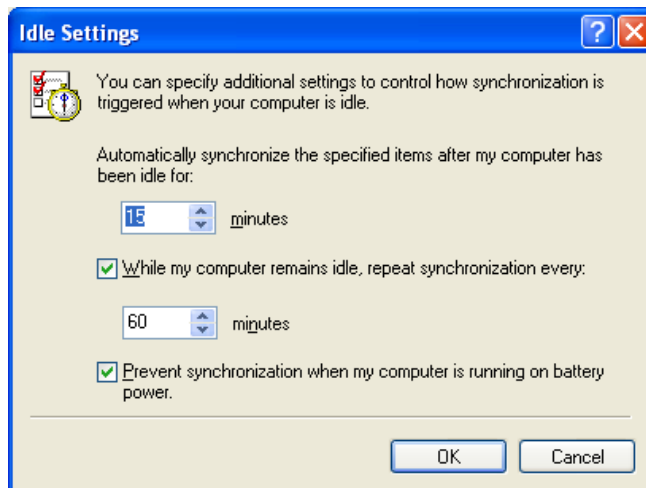


4. If you have to synchronize only one database, you can click immediately on **synchronizing**, otherwise select the desired object.

- During the synchronization, you can specify with a click on the **Setup** Button, when the replication should be started, e.g. with the login or logoff on the network



- You can specify **On Idle** settings over the register e.g. that the synchronization runs in the idle time of the computer

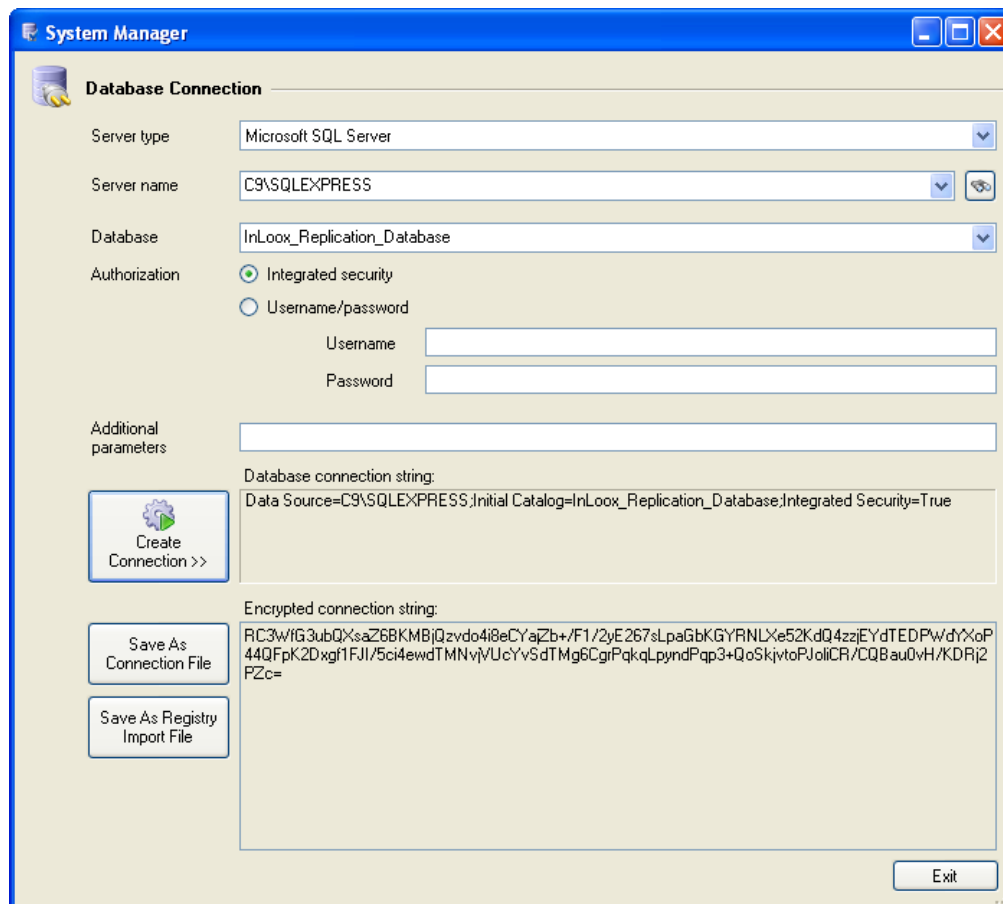


- Furthermore you can automate the synchronization over **Planned Tasks**

## Changing InLoox database connection on the Client

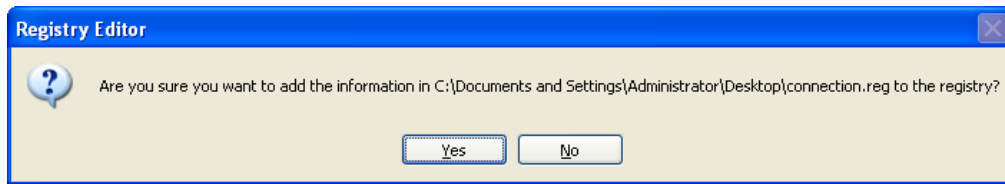
Finally, you have to change the database connection of the InLoox PM clients. The connection data is stored in the Windows Registry. You have to change the database pointer, so that in future, InLoox PM works with the local database replication. Proceed as follows:

1. Download the System Manager archive from the [download area](http://www.inloox.com/download/release/InLooxSystemManager.zip) of the InLoox homepage  
<http://www.inloox.com/download/release/InLooxSystemManager.zip>  
Extract all files of the archives to a separate Folder  
e.g. C:\Program Files\IQ medialab\InLoox\systemmanager
2. Directly start the file **InlooxSystemManager.exe** from this Folder
3. Afterwards the **System Manager** window opens
4. Click on the binoculars symbol at the right edge of the **Server Name** line
5. After a few seconds the **Server Name (local)** is selected
6. Enter the name of the replicated database, in our example **InLoox\_Replication** in the field **Database**
7. Click on **Create Connection**



8. An encrypted connection character is created
9. Store this connection character sequence in a folder of your choice by pressing the **Save As Registry Import File** button e.g. on the **Desktop**
10. Assign a meaningful name for this file e.g. **InLoox\_Offlinedatabase** and click **Save** afterwards. If all clients use the same connection sequence, you can rollout the stored file via software distribution, in order to automate the change on several Clients

11. Close the **InLoox System Manager** and **Microsoft Outlook**
12. Start the created file on the **Desktop** with a double-click
13. A registration editor message appears:



14. Click on Yes and confirm the change of the database  
**Important:** At least you need **main user permission** for changing the database connection
15. Start **Microsoft Outlook**. Now you can work off-line with **InLoox PM**.

## Replication – Upgrade the InLoox database

All clients should be synced with the main database before it is updated, to prevent data loss.

You can update the publisher database with the setup like non replicated databases. We recommend to run the setup on the server to prevent issues with network connections.

But after this is finished, additional steps have to be done so the replication takes care about new tables, columns and schema changes.

The upgrade adds tables and columns which have to be added to the replication.

1. **Start the SQL Server Management Studio**
2. Connect to the SQL-Server, which is the **Publisher**
3. Expand the node **Replication** and the node **Locale Publications**
4. Right click on the **publication** and select **Properties** from the context menu
5. Change to the tab **Article**, and deactivate „**Only show selected articles**“

The list now shows all objects. Verify that all objects are checked.

Attention!

Due to an error in the Management Studio the deselection and selection of the highest hierarchy does not necessarily lead to a selection of all sub objects!

6. You receive the message, that after adding articles, a new snapshot has to be generated. Click on **Yes**
7. After adding all objects, you can leave the dialog with **OK**
8. Create a new snapshot, to distribute all changes
9. Click on Start to create the new snapshot
10. After that resync all clients

All clients should now have the new database schema and are able to work with InLoox PM 7