



# **Project management – integrated into Outlook**

InLoox 5.x off-line operation

An IQ medialab / OptCon Whitepaper

Published: January 2008

Author/copyright: © 2007-2008 Heinz Peter Bross, OptCon GmbH.

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

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

This White Paper is for informational purposes only. IQ MEDIALAB 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 IQ medialab GmbH.

© 2008 IQ medialab GmbH & Heinz-Peter Bross, OptCon GmbH. All rights reserve.

IQ medialab and InLoox are either registered trademarks or trademarks of IQ medialab 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
Alternative: Subscription configuration on the Client .....	30
Initialize a client created subscription.....	38
Changing InLoox database connection on the Client .....	40

## General information

### What is a replication?

Replication is a set of technologies for copying and distributing data and database objects from one database to another and synchronizing them afterwards, in order to guarantee the data consistency. With the data replication you can transfer data from a source database to one or more Microsoft SQL Servers. The data can be distributed to different locations, to remote users or mobile users over local networks and WANs (Wide AREA network), telecommunication connections, wireless connections or over the internet.

You can also copy and transform data over the replication and distribute user-defined data to several servers afterwards. In generally the replication is used, if you have to administer data on several servers periodically. If you only need a backup of the database or copy it only once, you do not need a replication. In this case use „Restoring of a database on a different location“, or simply run an import or export procedure. Usually an administrator of the database module handles the replication management. The administrator plans the daily procedures like availability of the system, monitoring and optimization, development, update, problem treatment and configuration. In small and medium enterprises maybe only one administrator is responsible, while the administrator role in large enterprises might be distributed on several persons.

### The replication-distribution-model

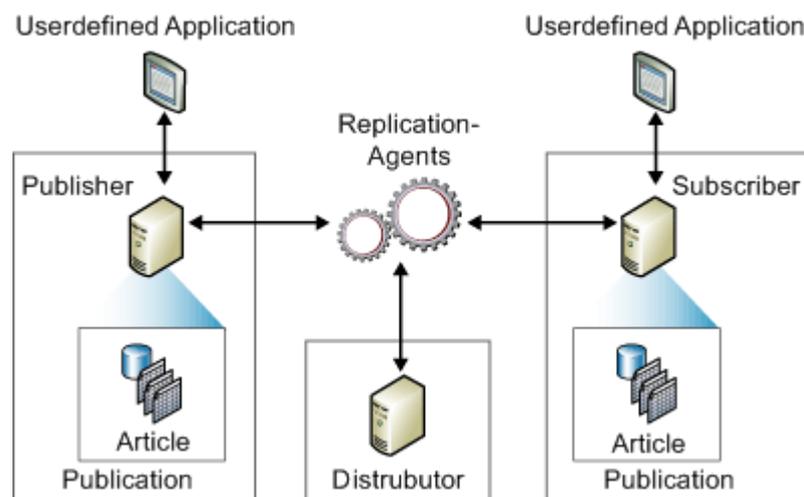
The replication architecture contains:

**Replication components**, that are the server and data components of the replication

**Replication-Agents**, that are applications, which support the replication process

**Replication-Variant**, that is the replication type, which can be configured.

Within SQL Server 2005 the replication is described with terms of the publishing industry. For the representation of the components the terms publisher, distributor, subscriber, publications, articles and subscription are used in the model of replication topology, which relies on processes from the publishing industry.



## 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.

## 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** preferred with service pack 2 and

**SQL Server 2005 Express Edition** preferred with service Pack2

#### SQL server

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

**SQL Server 2005 Workgroup, Standard or Enterprise Edition.** An Express Edition on the server is not sufficient, in order to be able to create a publication

#### SQL Server Management Studio

**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“

### 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.

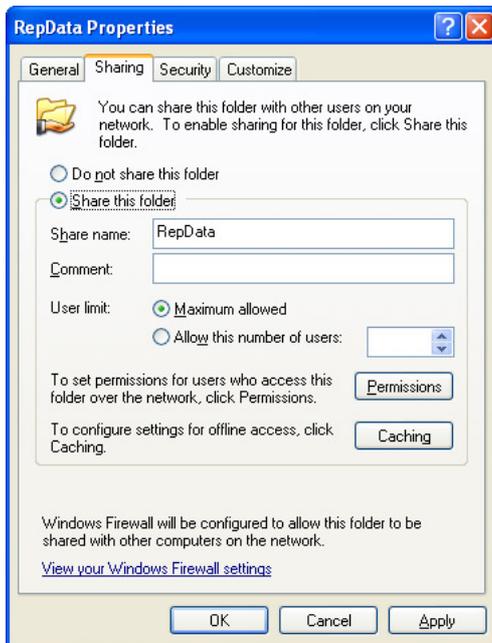
Windows Vista contains a new feature, the User Account Control (User Account Control, UAC), which administrators use to administer their extended user permissions (also called privileges). According to standard administrators do not use their administrator permissions on Windows Vista during execution. 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

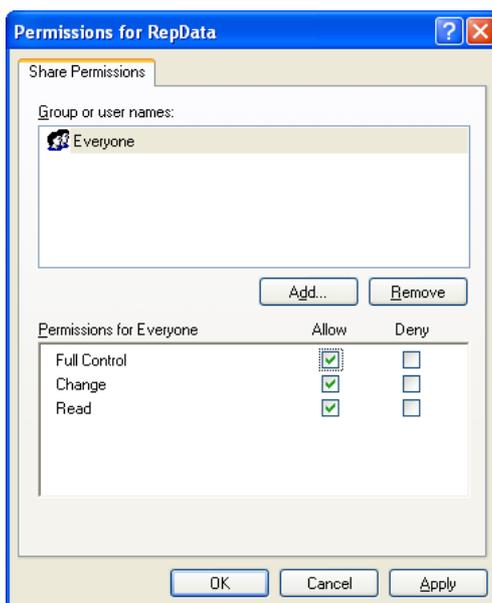
**X:\Programme\Microsoft SQL Server\<Instanzname>\MSSQL\RepData** is used as snapshot folder. If you use a **remote distributor** or a **pull subscription**, you must use a UNC-network-share (e.g. \\<Computername>\Snapshot) instead of a local path.

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**

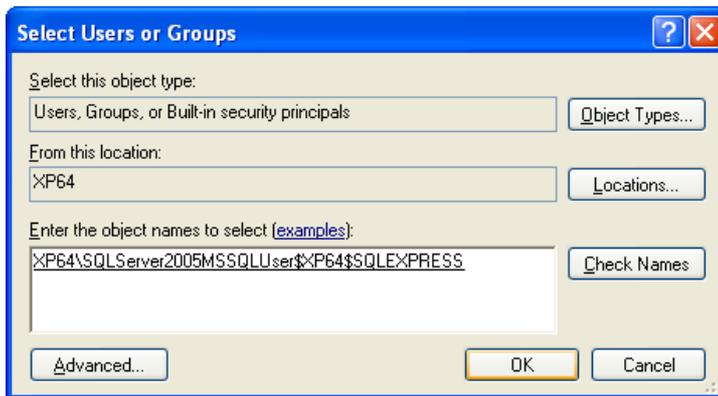


5. Click on **Permissions**
6. In the **Permissions for RepData** window click on **Add** if you like to add certain users, or select the group name of **Everyone** and check **Full Control**, if you would like to give full control to all users on the folder



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.

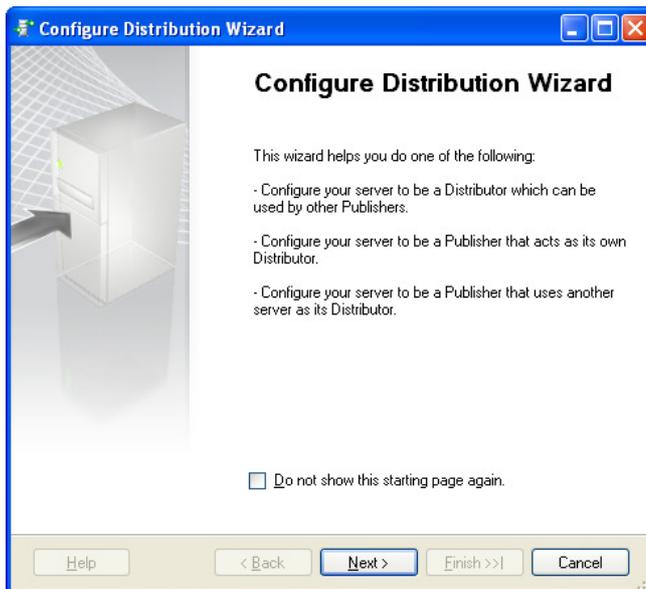
The distributor is a server, which contains the distribution database. In the database metadata and history data for all replications and transaction types of the transaction replication are stored. To setup the replication you have to configure a distributor. Each publisher can only be assigned to one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

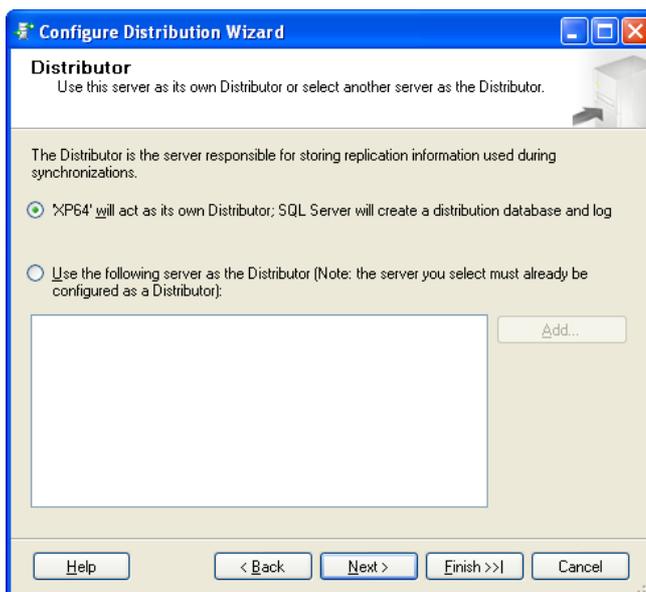


6. The **Configure Distribution Wizard** is started:



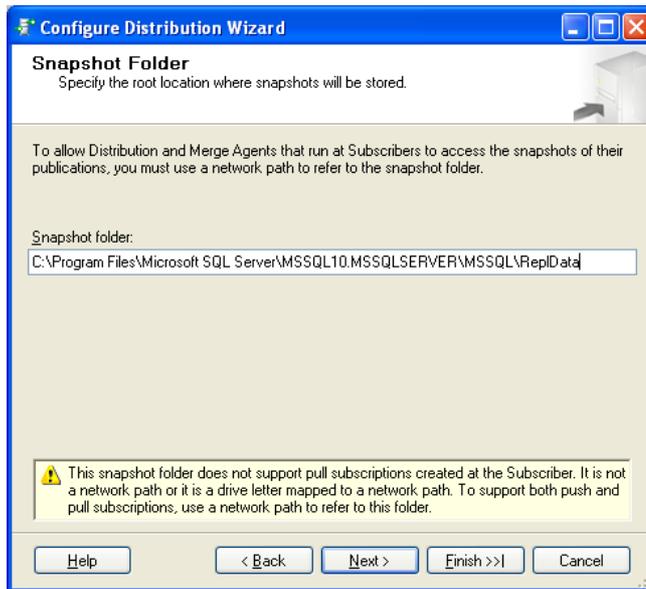
7. Click on **Next**

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



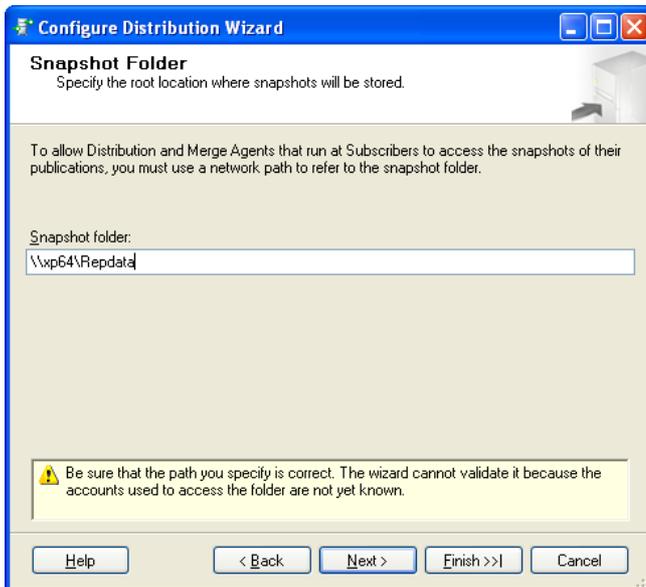
14. Independently of their type each publication needs a base snapshot. This Snapshot is stored in a folder, which is used together by all replication agents.

The standard path is **C:\Program Files\Microsoft SQL Server\MSSQL.1\MSSQL\ReplData** 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\repdata** at first



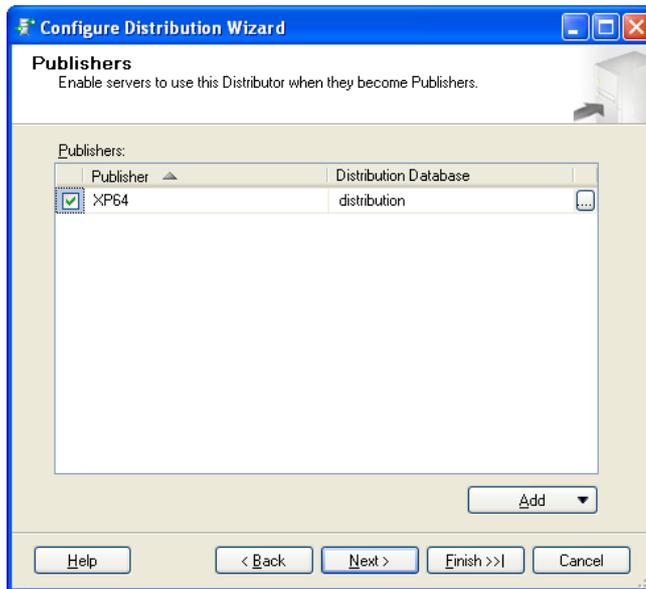
17. Click on **Next**

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



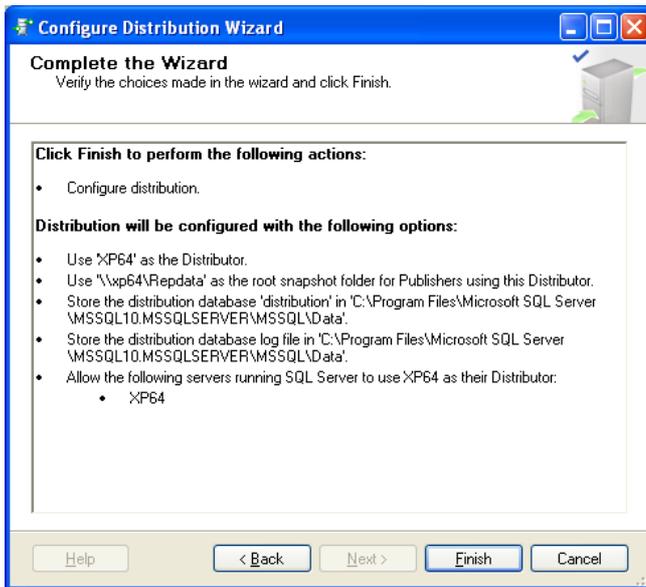
21. Here you see all servers on the network and you are able to select the publishers, who use this system as distributor. If you like to add a server, so that it can be used as distributor, as soon as it is changed to a publisher, click on **Add** and select one of the offered possibilities afterwards

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



23. Click on **Next**

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



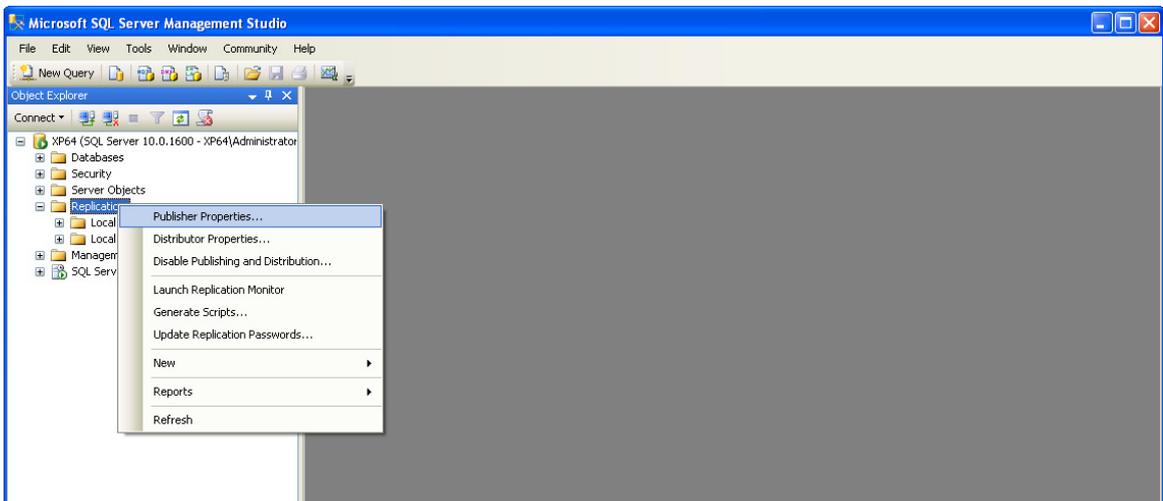
25. Click on **Finish**

The publisher will be installed

26. Click on **Close**

The distributor is established and operational

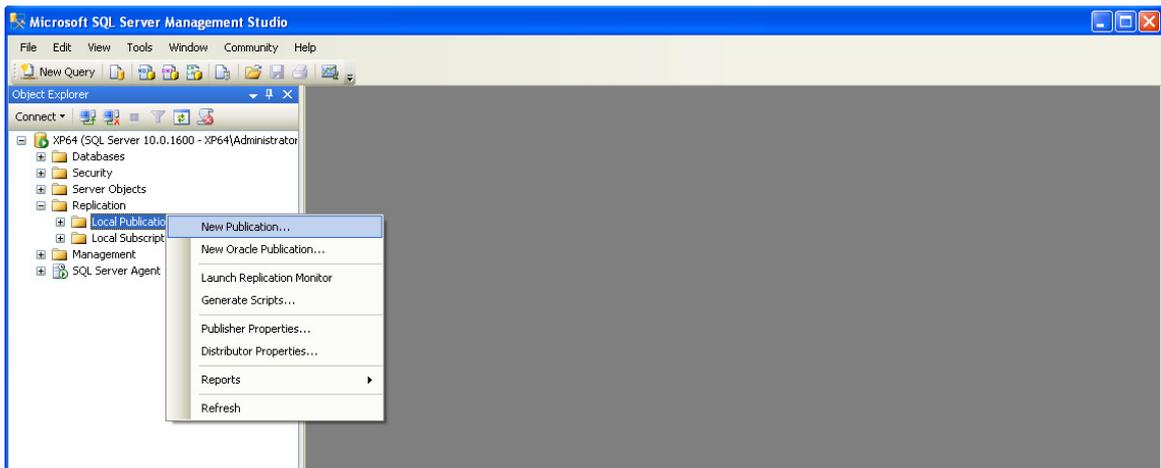
27. You can verify the settings status and if the publisher is set up correctly, by right clicking on the **Replication** node



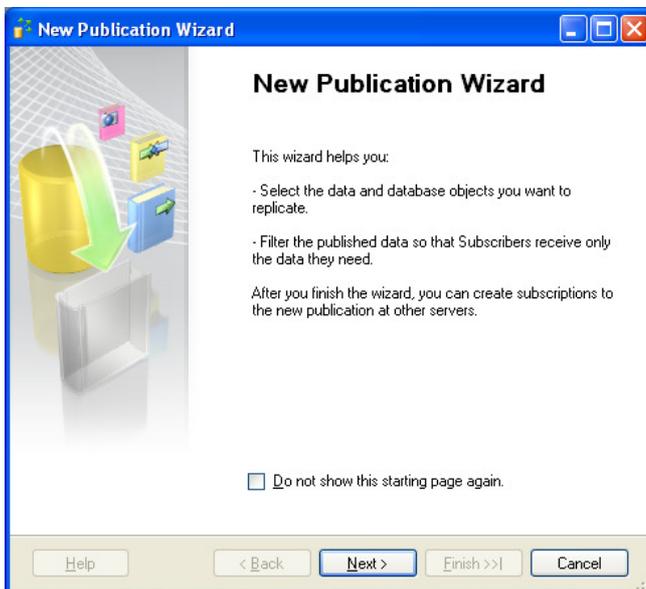
28. Here you can examine the **Publisher Properties**

## 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...**

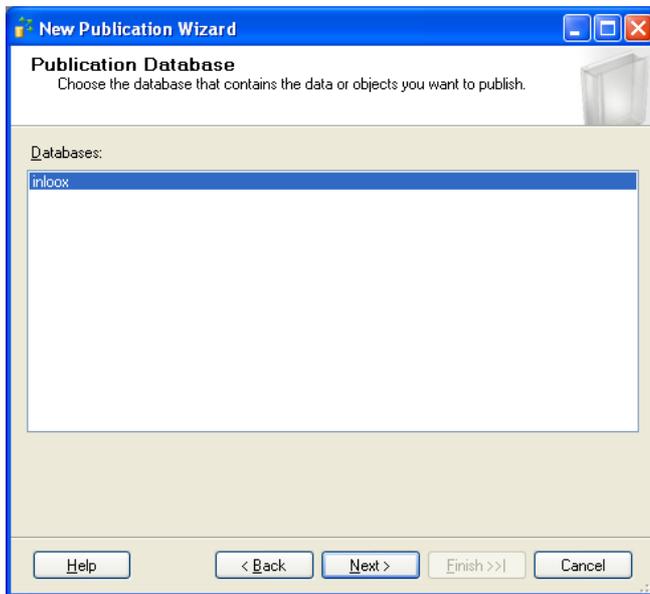


6. The window **New Publication Wizard** appears



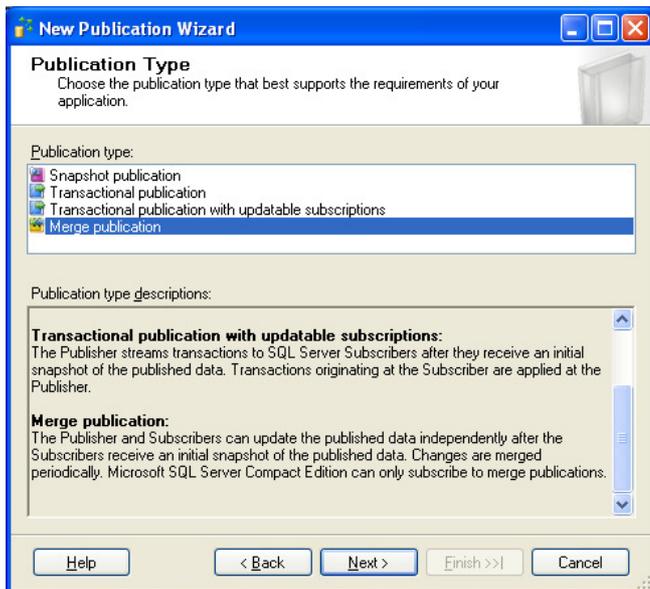
7. Click on **Next**

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.

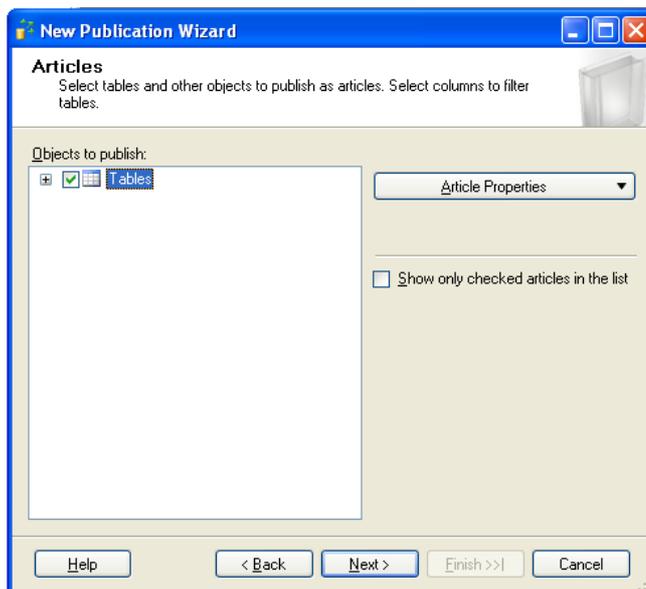
12. Click on **Next**

13. The window **Subscriber Types** appears



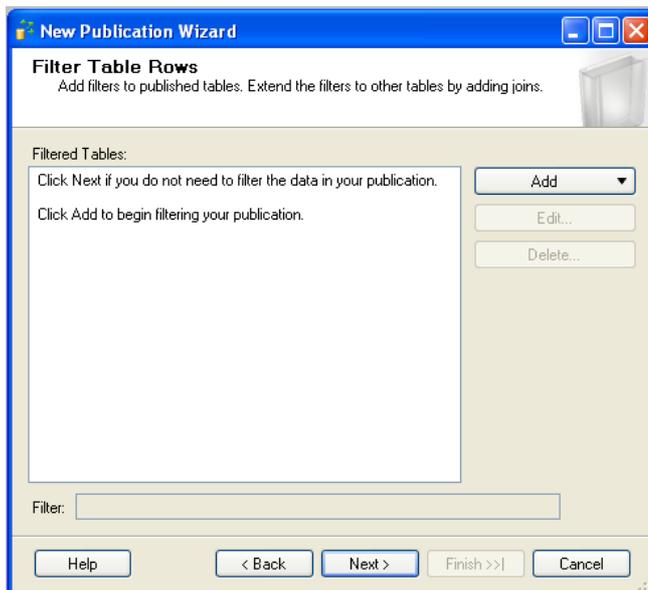
14. Activate all options here and click you on **Next**

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:

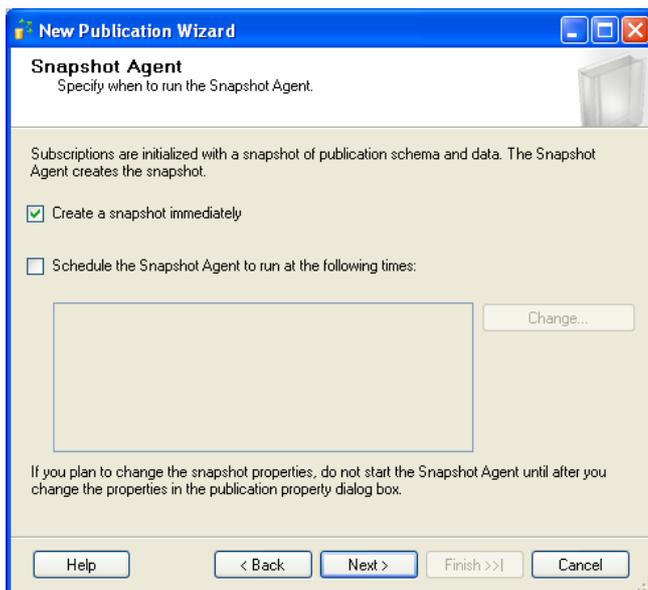


16. Click on **Next** afterwards

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

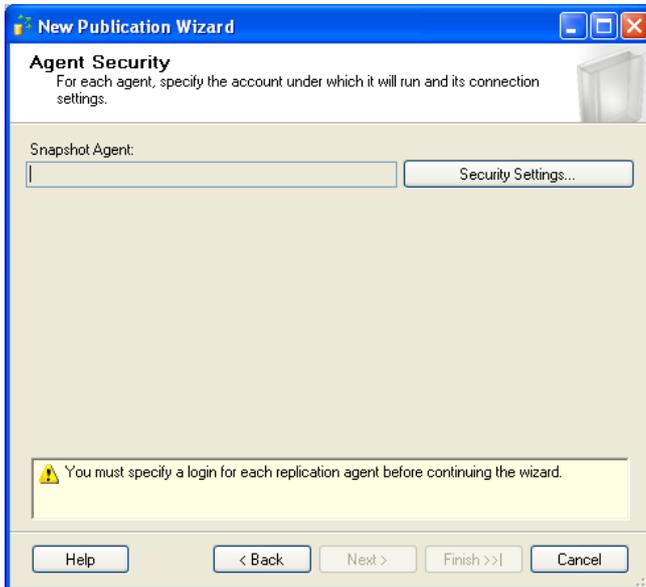


18. Specify the base snapshot of the data for immediate or after a schedule creation in the **Snapshot Agent** window



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.

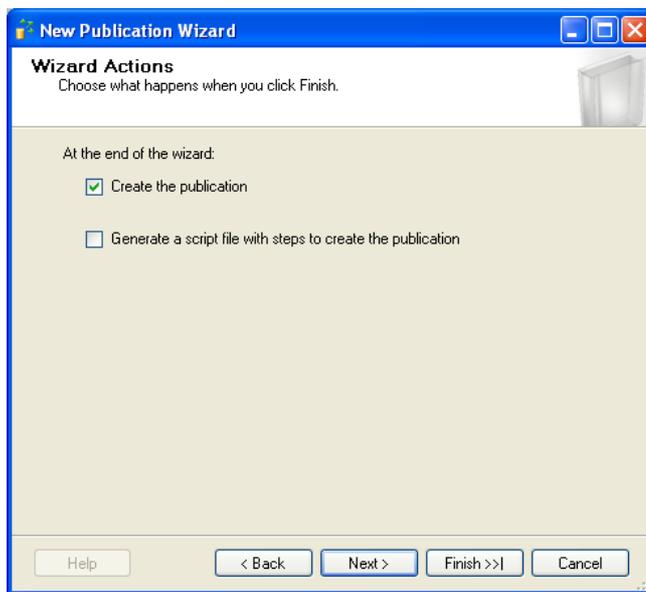


21. With the button **Security Settings...** you can specify a Windows or an SQL Server account. If you have created an account for the replication procedure, you can specify it here. Please remind that it needs access to **all systems** (Servers/all clients with off-line availability) as well as the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



22. Click on **OK** afterwards

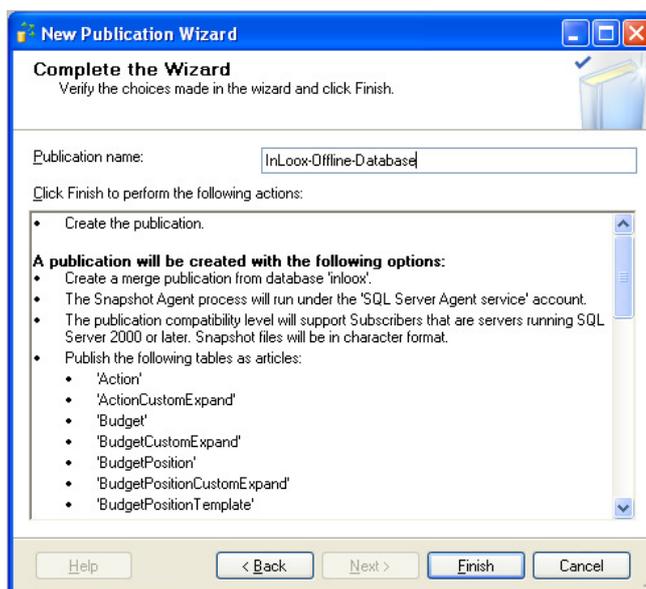
23. The window **Wizard Actions** appears



24. Click on **Next**

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

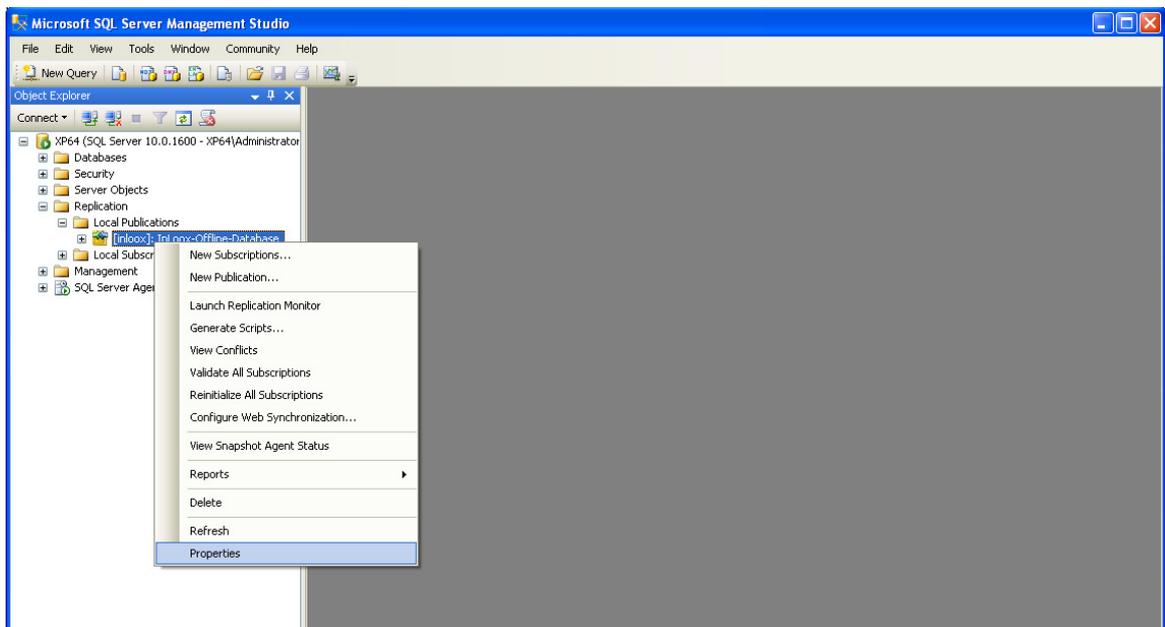
26. Enter a **Publication name**. It has been proved of value to use the database name



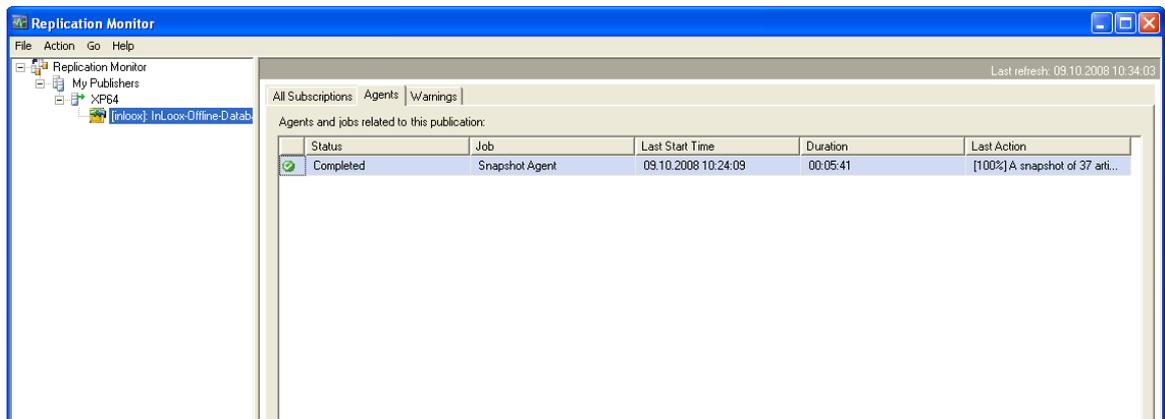
27. Click on **Finish** and the publication will be created

28. In the end a message is shown that the publication was created 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**:



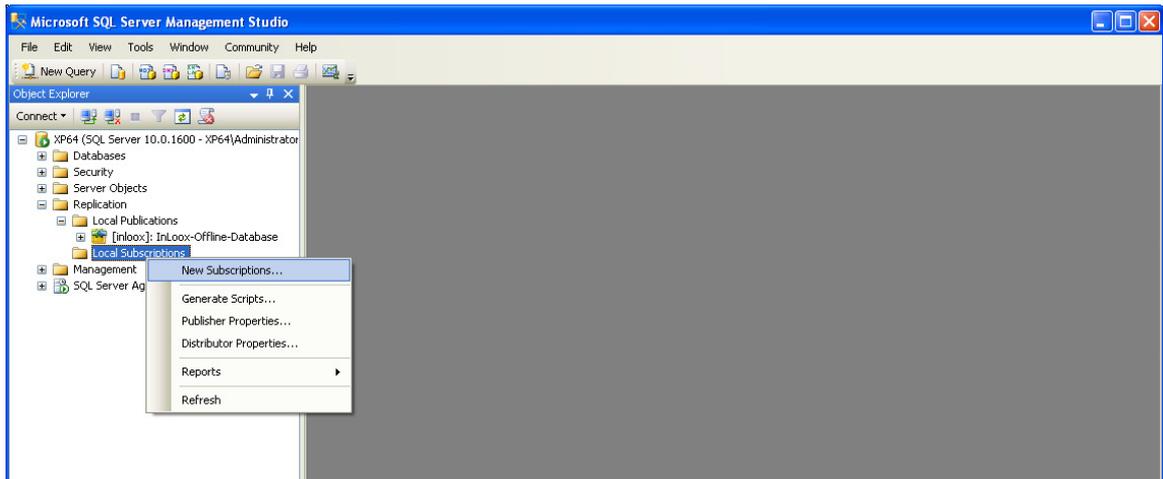
30. Use the **Replication Monitor** (available under the node **Management**), in order to examine whether errors occurred:



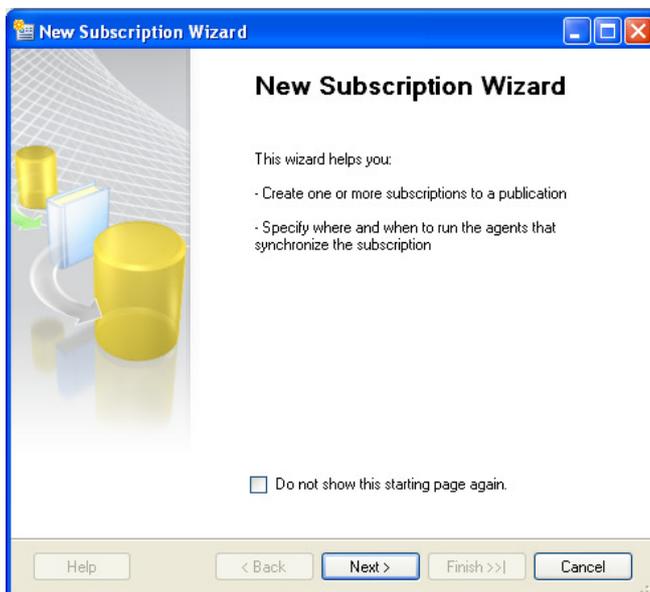
## Central subscription configuration on the server

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**

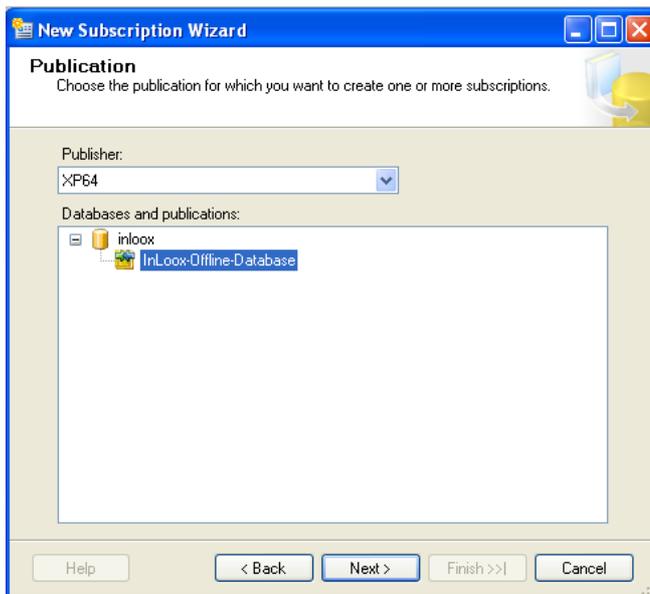


5. Click on **New Subscription...**  
Afterwards the **New Subscription Wizard** appears:



6. Click on **Next**

7. The window **Publication** appears



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

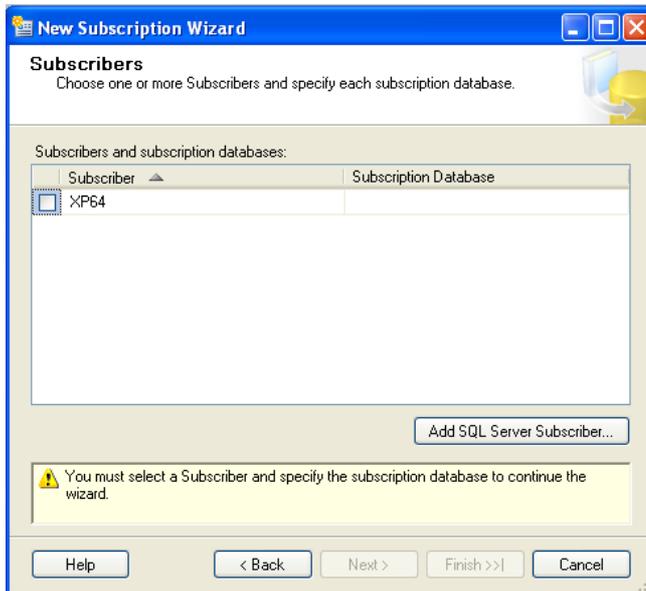
9. Click on **Next**

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



11. Select whether the merge agent should be run at the distributor or at the subscriber. Select **push subscription** here

- Click on **Next** and the **Subscribers** window appears with a selection of the subscribers and subscription databases.

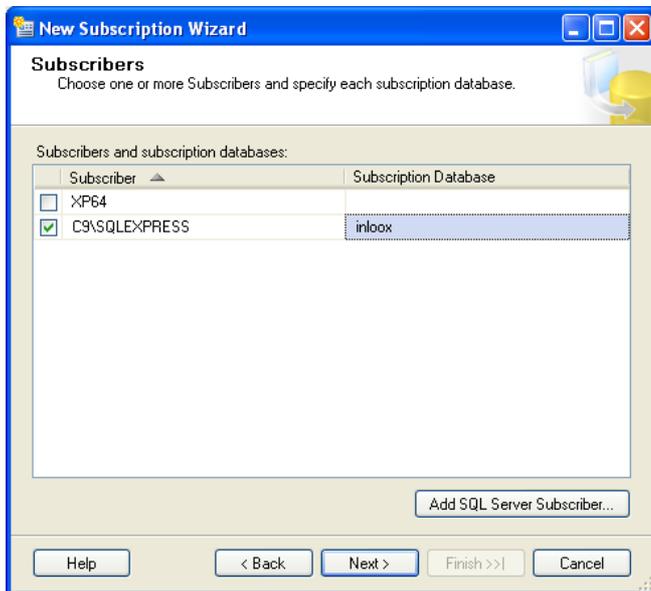


- Click on **Add SQL Server Subscriber...** and connect with the appropriate subscriber, in this case with your client or your notebook.

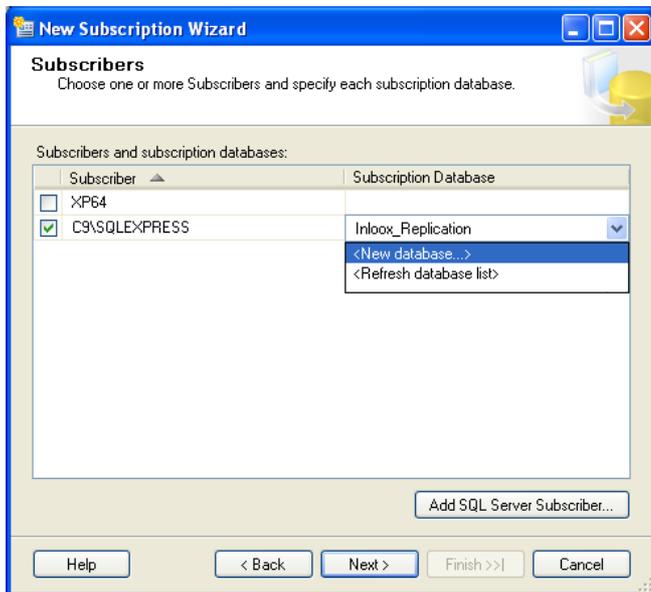


- Click on **Connect**

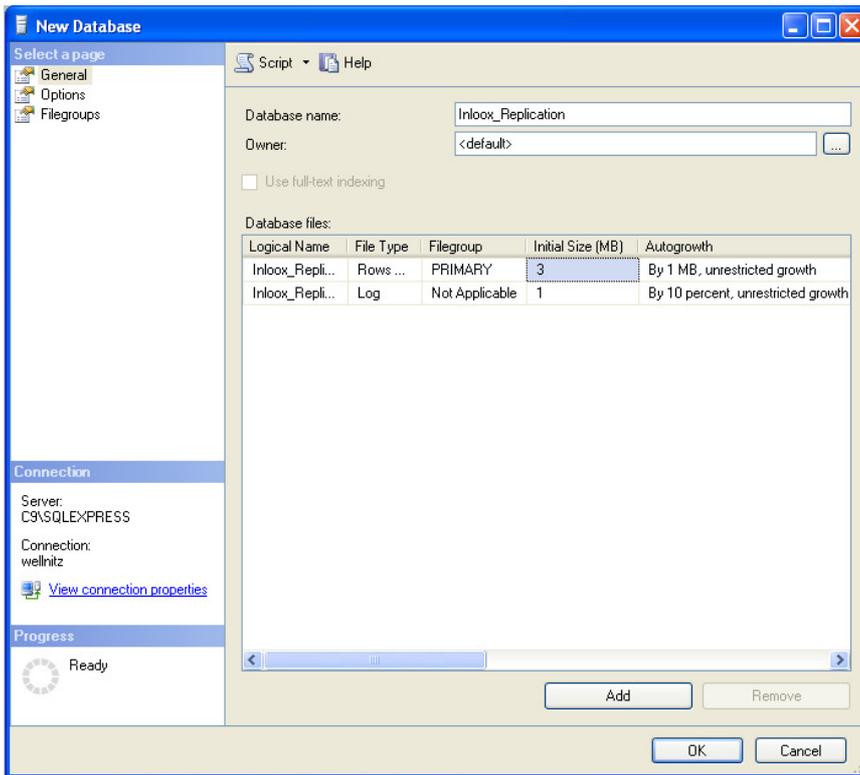
15. The **Subscribers** window appears again
16. Open the list under the column **Subscription Database**



17. Select the **New Database...entry**

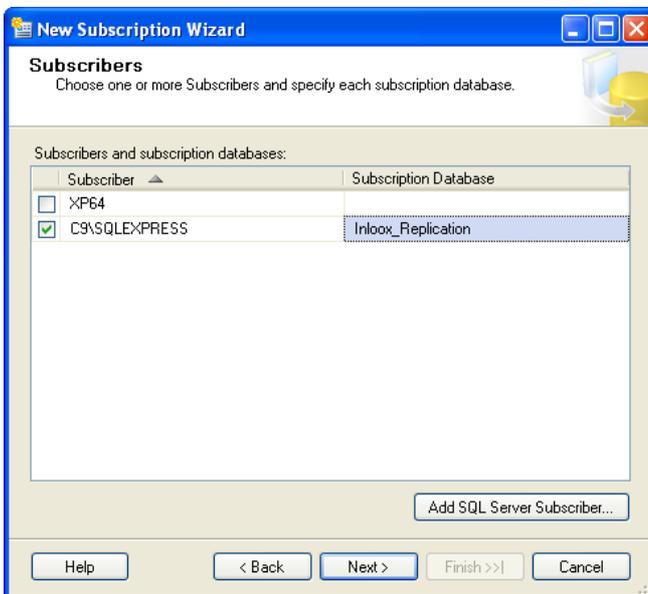


18. Assign a database name in the window **New Database** e.g. **InLoox\_Replication**



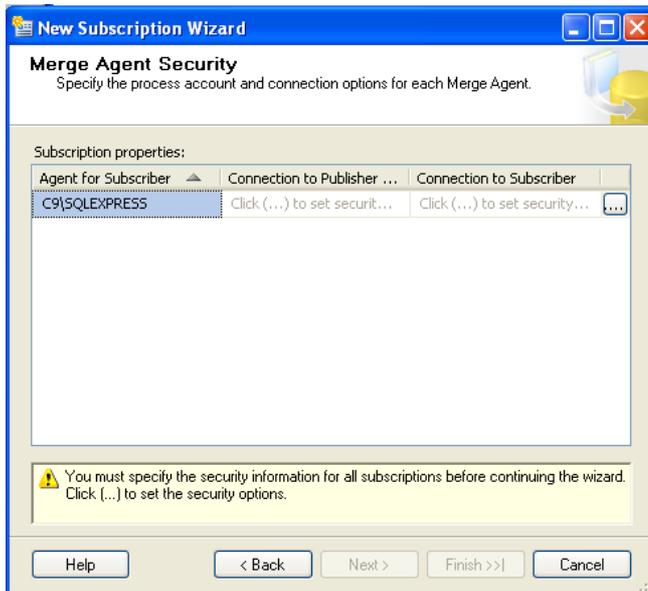
19. Click on **OK**

20. The **Subscriber** window appears again, but this time with the selected database **InLoox\_Replication**



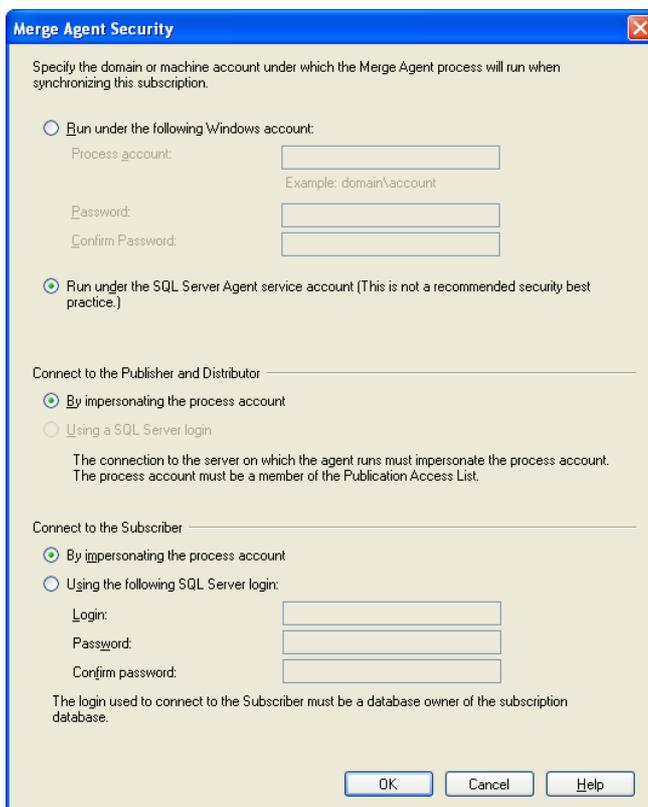
21. Click on **Next**

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

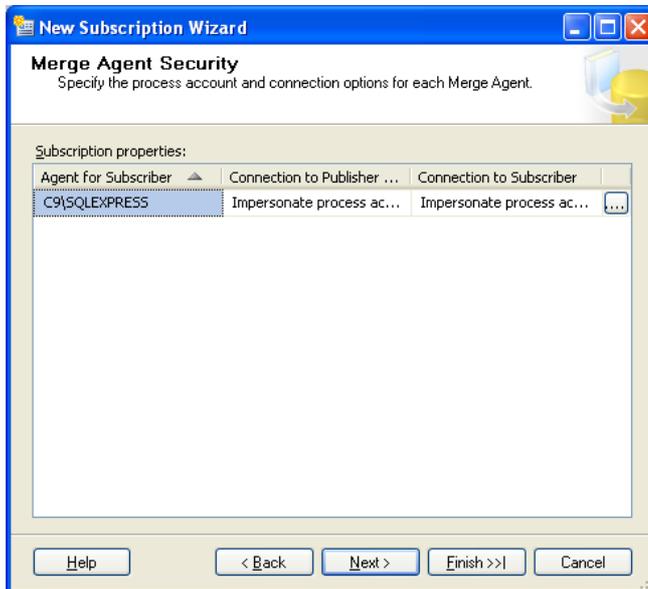


23. You have to set the security information for all subscription before the wizard can be continued. Click on (...), in order to configure the security settings.

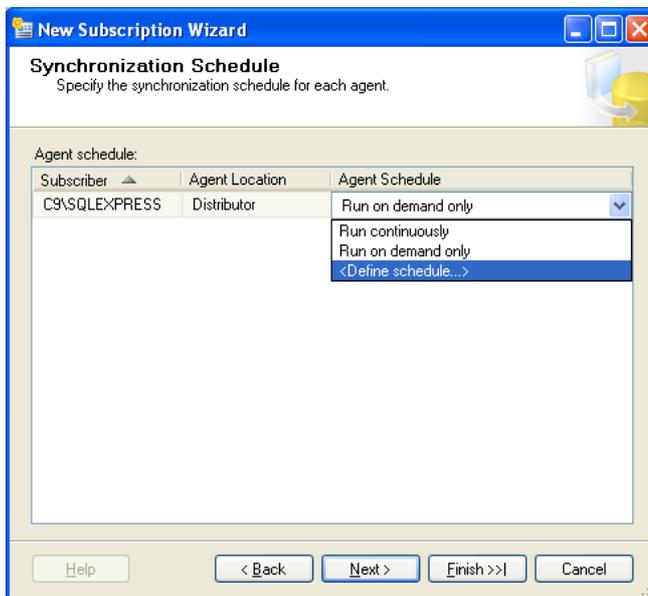
24. Select the **Run under the SQL Server Agent service account (This is not a recommended security best practice)** option in the **Merge Agent Security** window



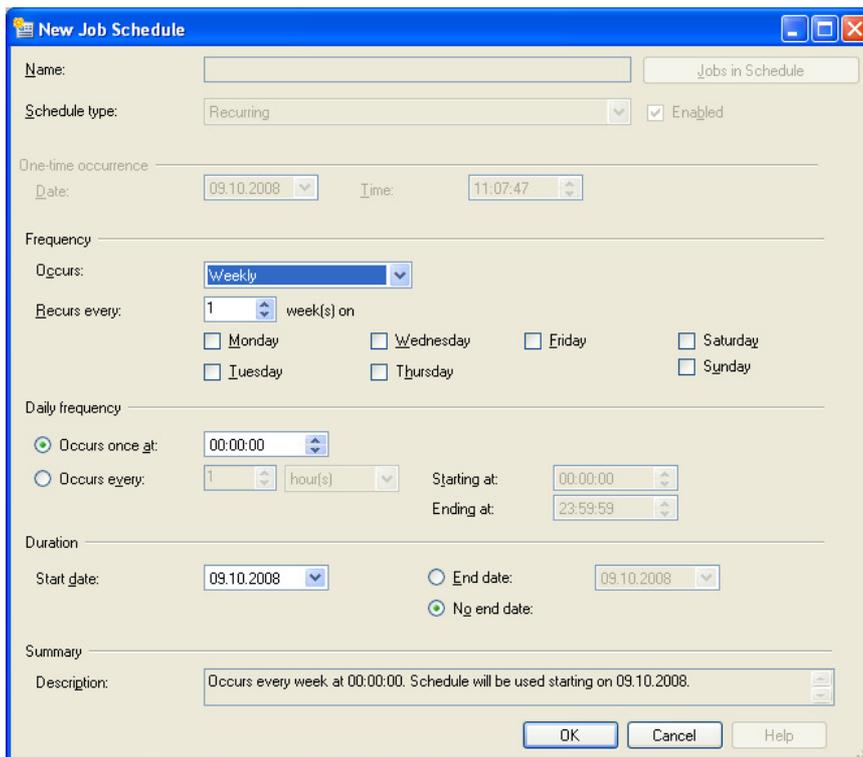
25. Click on **OK**
26. The following inputs will appear in the **Merge Agent Security** window



27. Click on **Next**
28. The **Synchronization Schedule** window appears. Select **Define Schedule...** under the **Agent Schedule** column

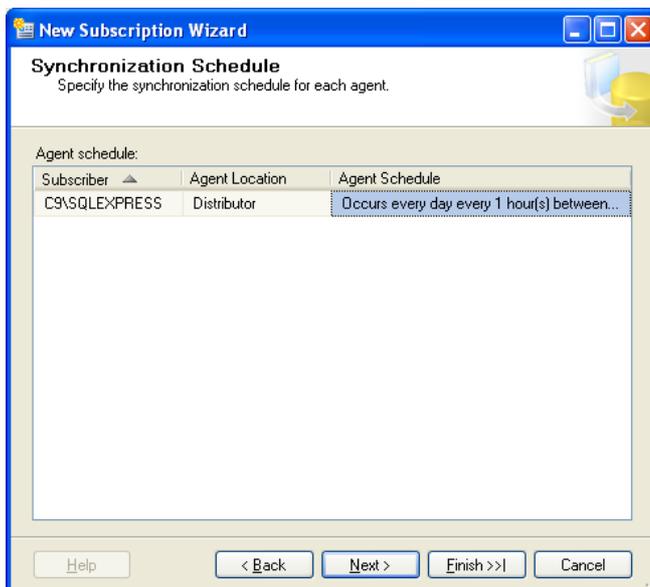


29. The **New Job Schedule** window appears automatically



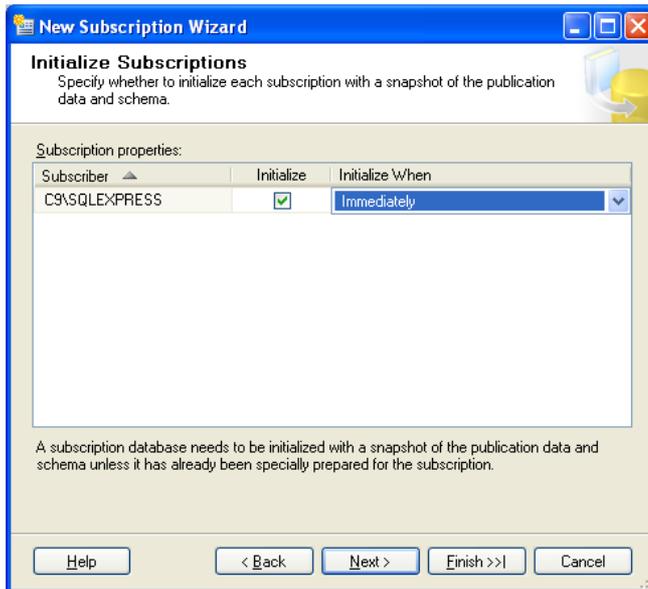
30. Enter the desired data and dates, e.g. all 5 min. daily. Pay attention on your network load, particularly with a large number of users. Click on **OK** thereafter

31. The **Synchronization Schedule** window appears again, filled with the selected data under the **Agent Schedule** column.



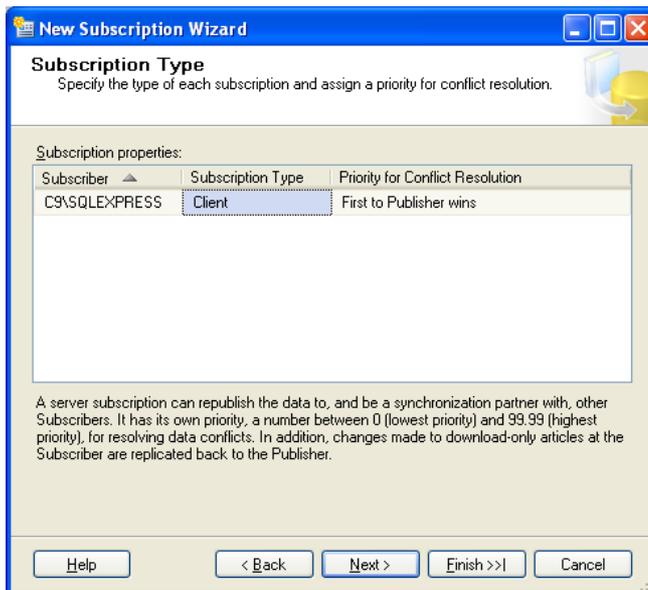
32. Click on **Next**

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



34. Click on **Next**

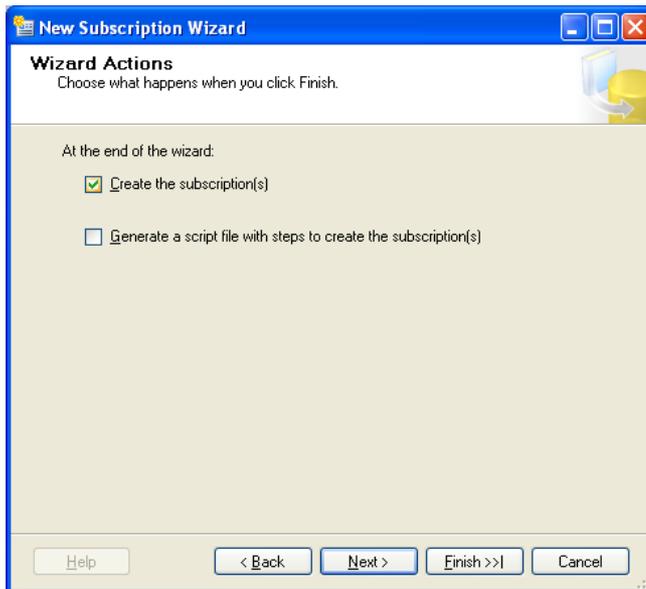
35. The **Subscription Type** window appears



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

37. Click on **Next**

38. The **Wizard Actions** window appears



39. Click on **Next**

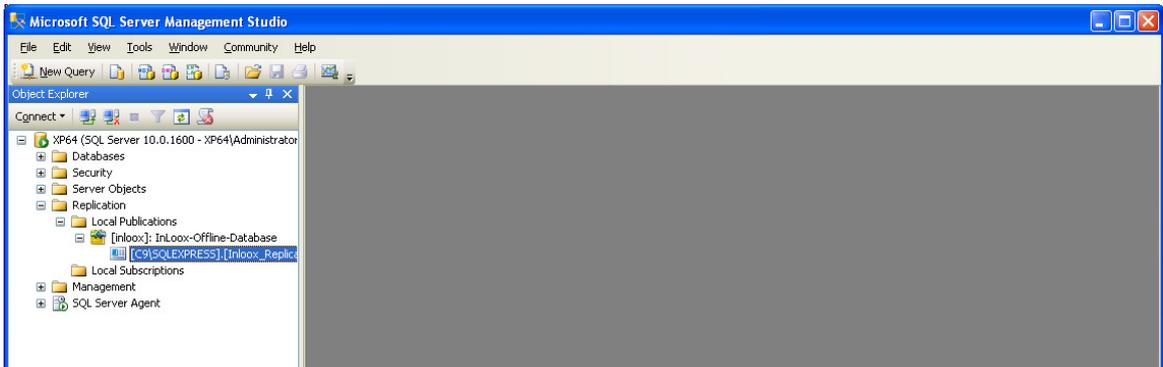
40. The **Complete the Wizard** window appears



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

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

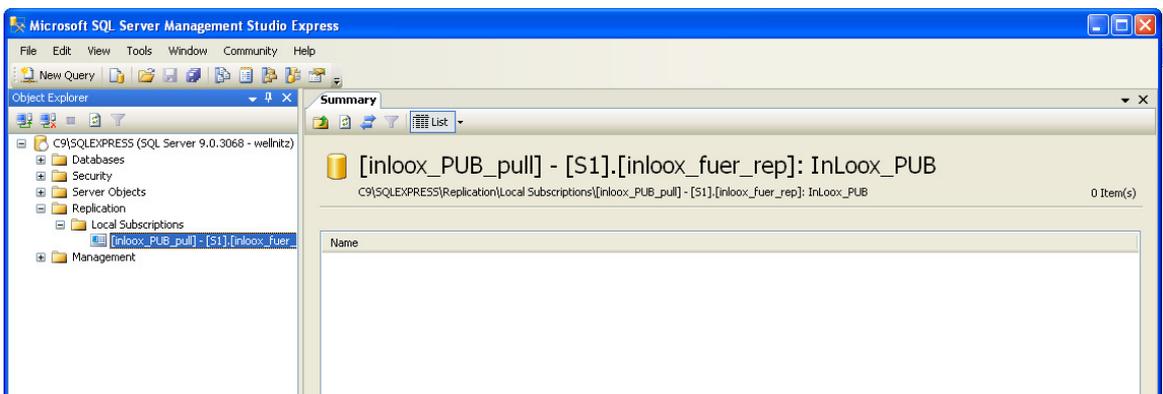
43. You can see the created Replication under the **Replication >> Local Publications** node



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



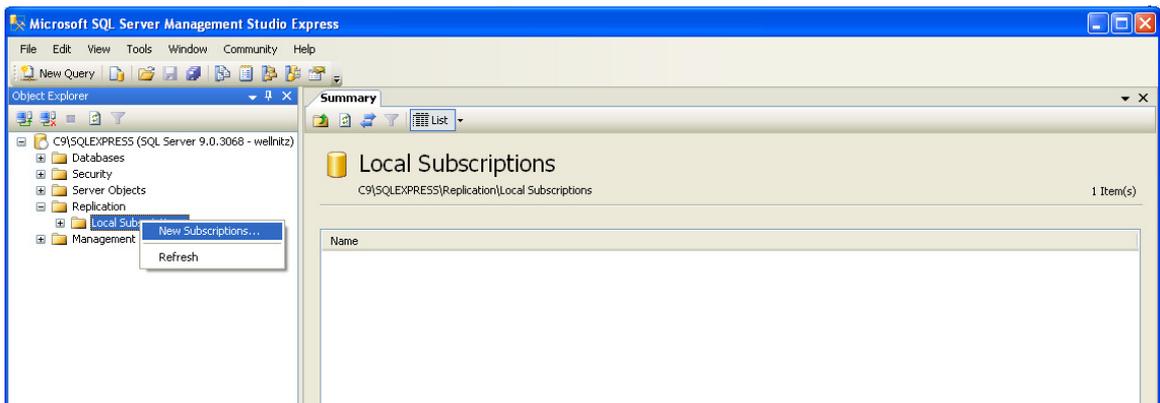
45. You see, that a local subscription was created on the client under the **Replication >> Local subscription** node. Perhaps you have to press **F5** for updating the list.



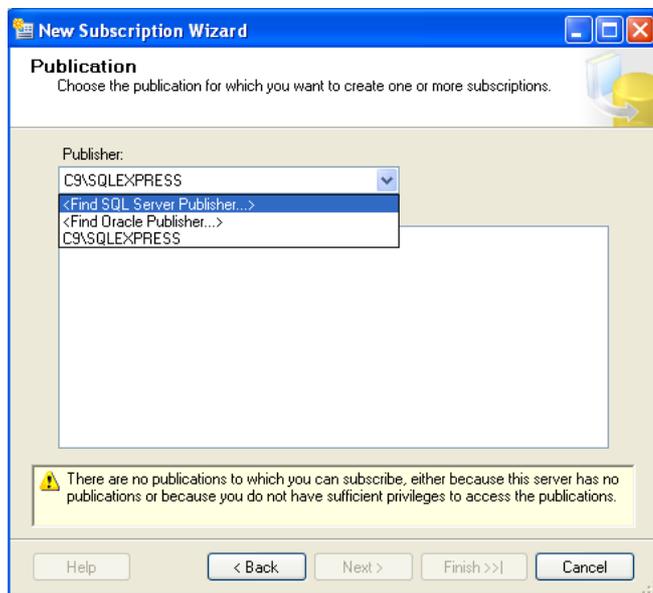
## Alternative: Subscription configuration on the Client

Alternatively you also can perform this subscriber creation on the Client. This can be unfavorable, since you have to configure all clients separately. Proceed as follows:

1. Start **SQL Server Management Studio** on the client
2. Connect with the local **Microsoft SQL Server Express Edition**
3. Extend the **Replication** node
4. Right Click on **Local subscription** in the **Object Explorer**



5. Click on **New Subscription...**
6. The **Publication** window appears

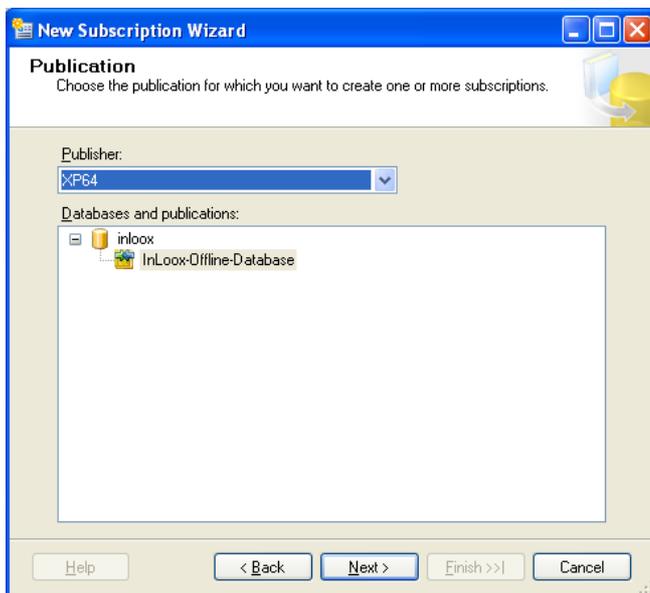


7. Select the SQL Server which you defined as publisher below under **Publisher**,

8. Connect to the server

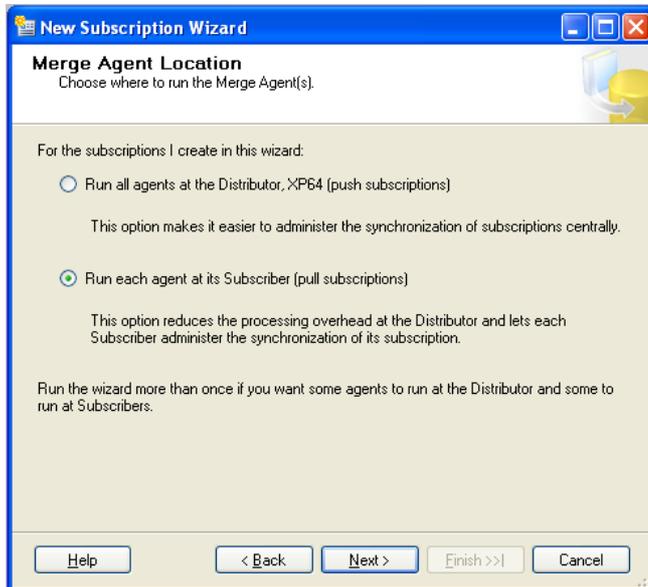


9. After clicking on **Connect** the **Publication** window appears



10. The database and the publication are already selected
11. Click on **Next**

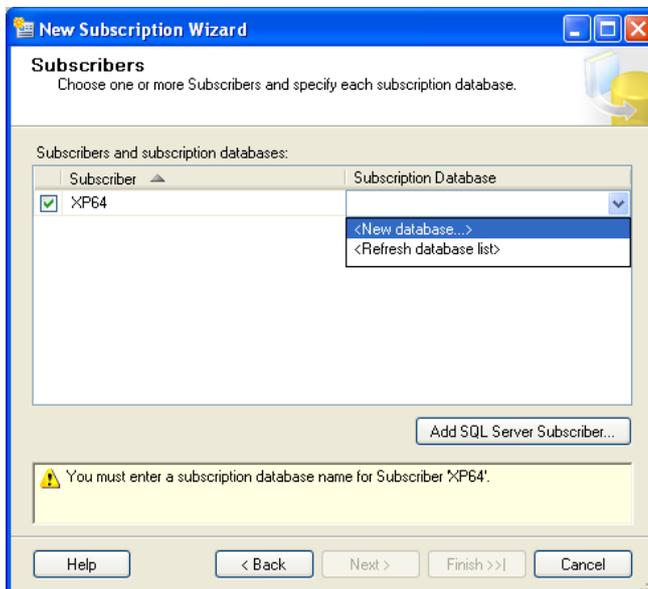
12. The **Merge Agent Location** window appears



13. Click on **Run each agent at its Subscriber (pull subscriptions)**

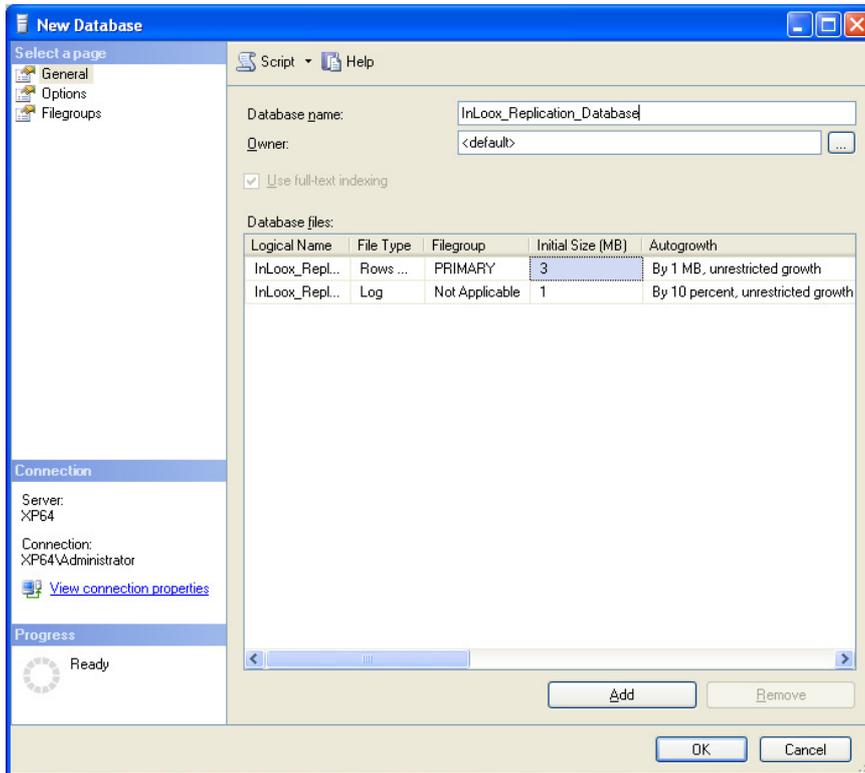
14. Click on **Next**

15. The **Subscribers** window for the selection appears

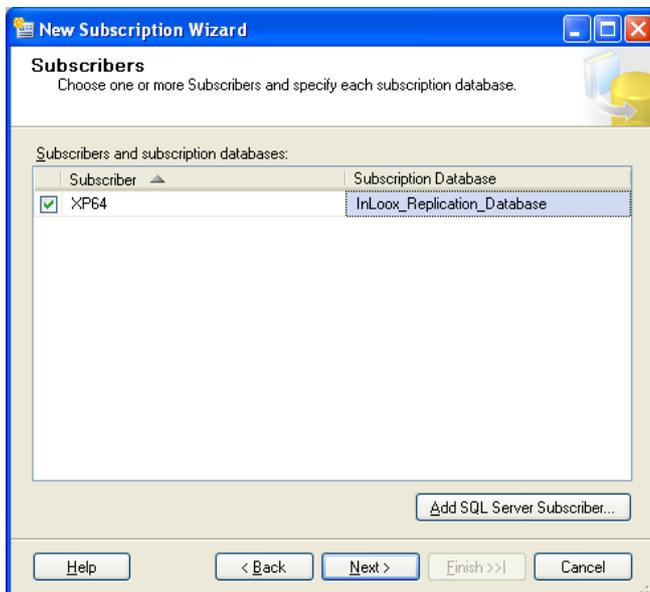


16. Select the **New Database...**entry in the subscription database column

17. Enter a name in order to create a new database. Afterwards click on **OK**

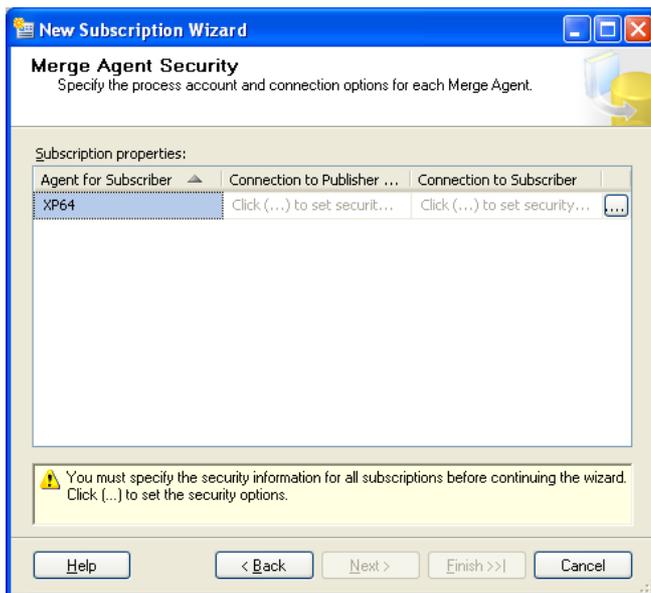


18. Now the **subscription database** is selected

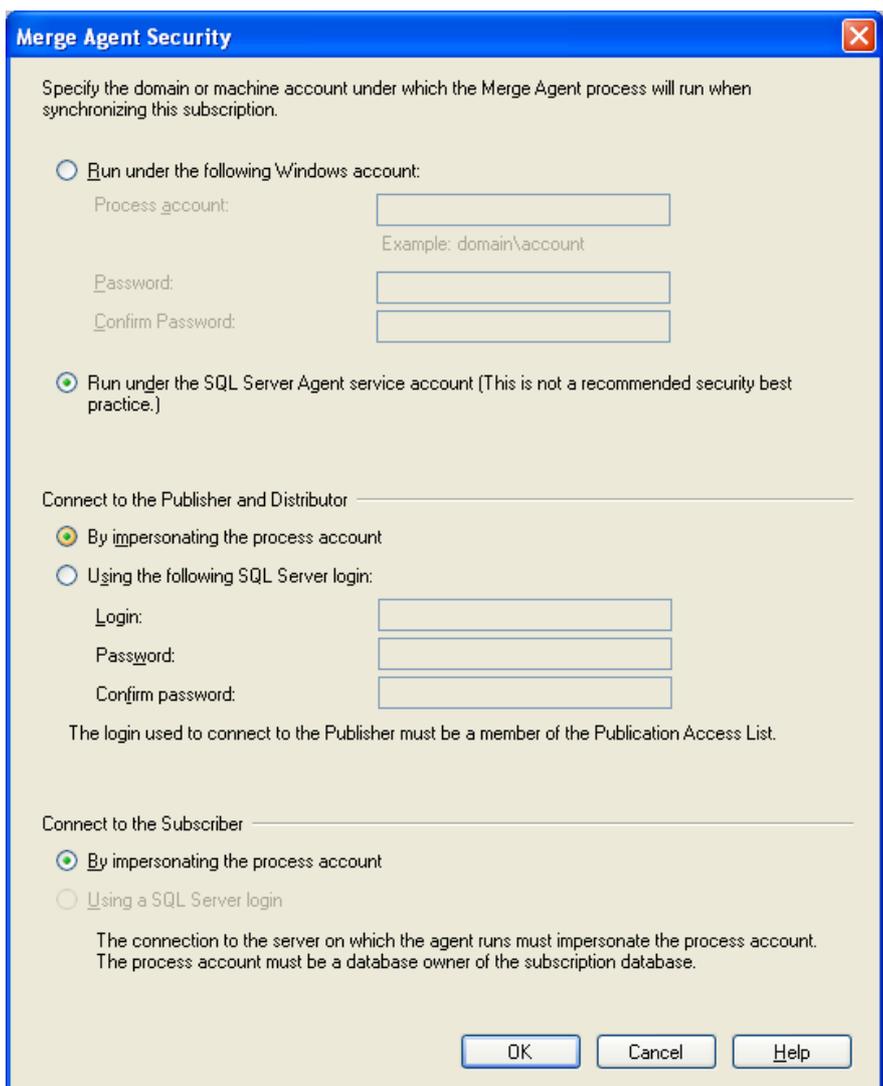


19. Click on **Next**

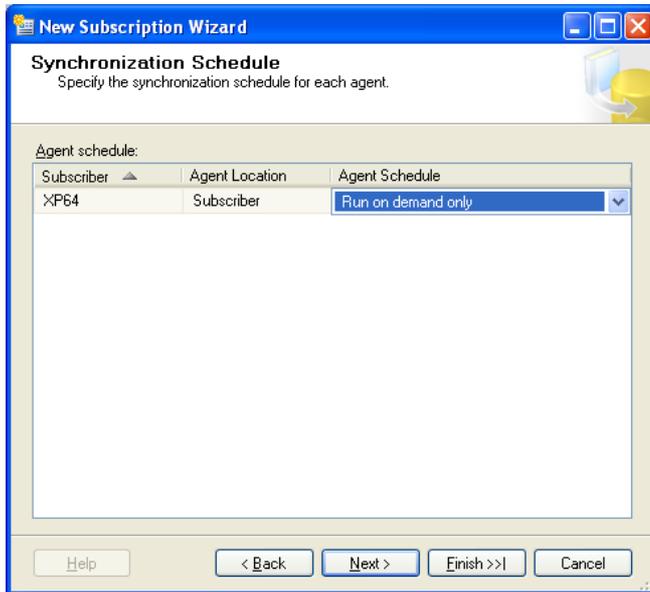
20. The **Merge Agent Security** window appears



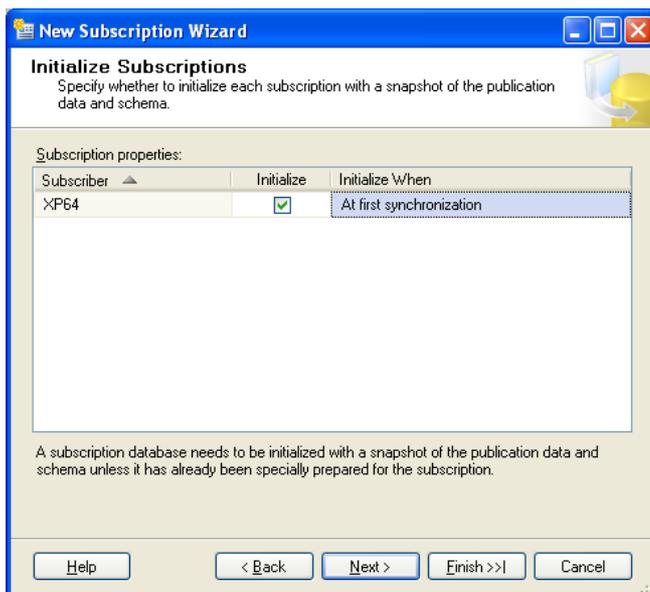
21. The **Merge Agent Security selection** dialog appears



22. Click on **OK** and then on **Next**
23. The **Synchronization Schedule** window appears

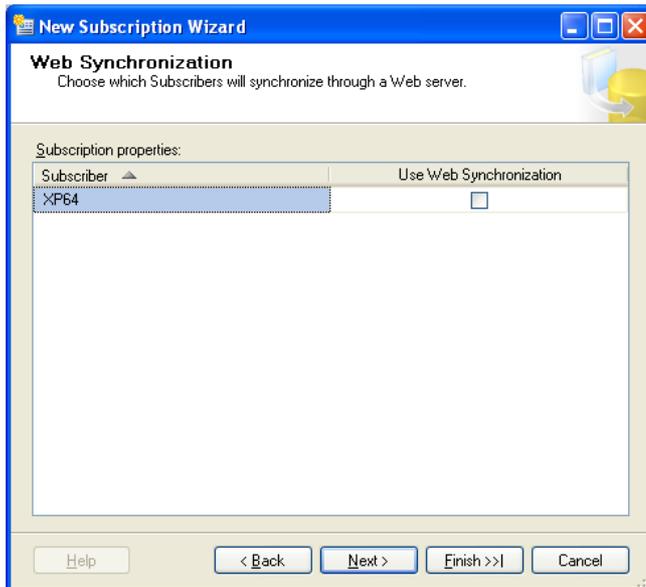


24. Click on **Next** and then the **Initialize Subscriptions** window appear



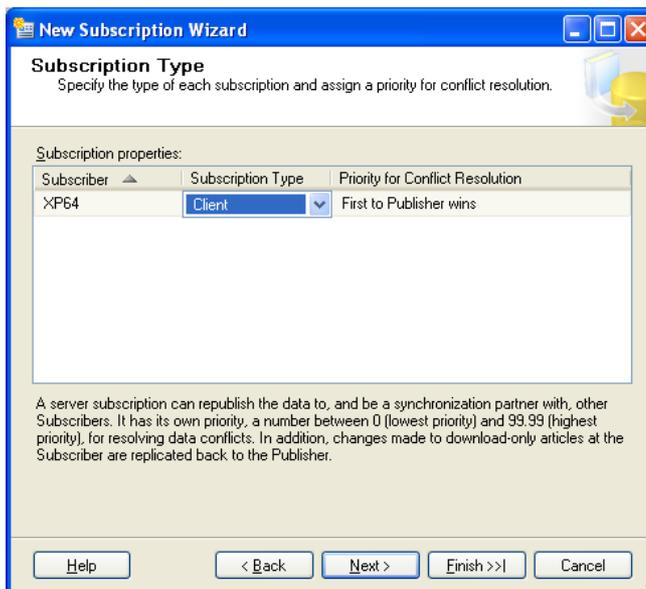
25. Leave the standard settings and click on **Next**

26. The **Web Synchronization** window appears



27. Click on **Next**

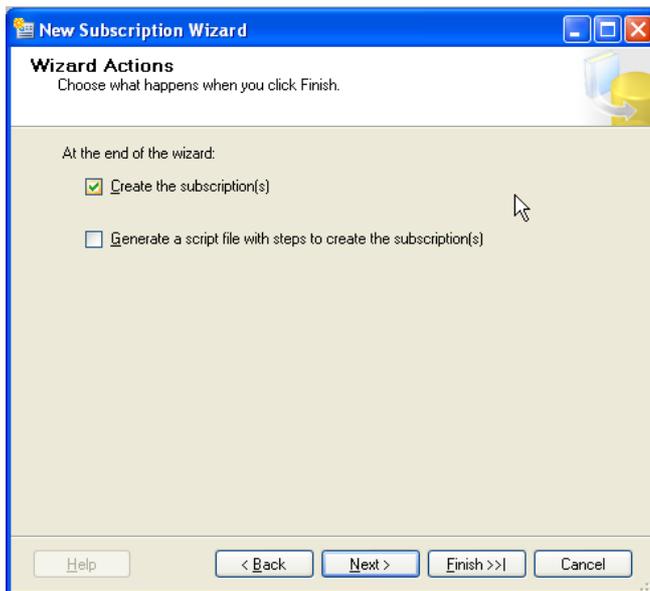
28. The **Subscription Type** window appears



29. Select the entry **client** below the **Subscription Type** column

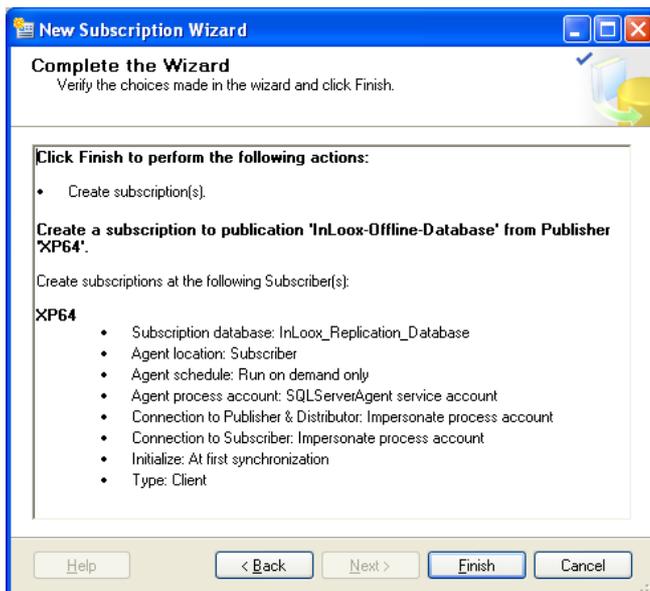
30. Click on **Next**

31. The **Wizard Actions** window appears



32. Click on **Next**

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



34. Click on the **Finish** button. **Subscription was created** window appears

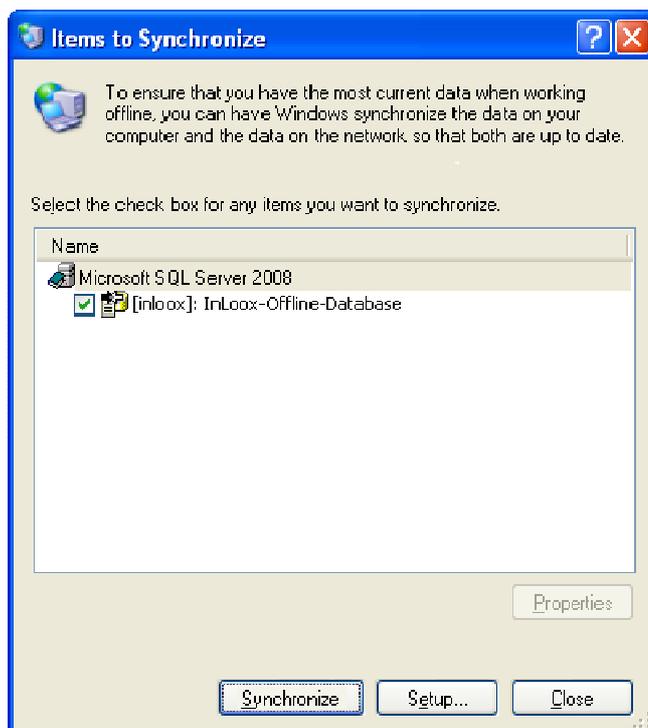
35. Click on **Close**

36. The subscription is created. Now perform the following steps for the **Initialization**

## Initialize a client created subscription

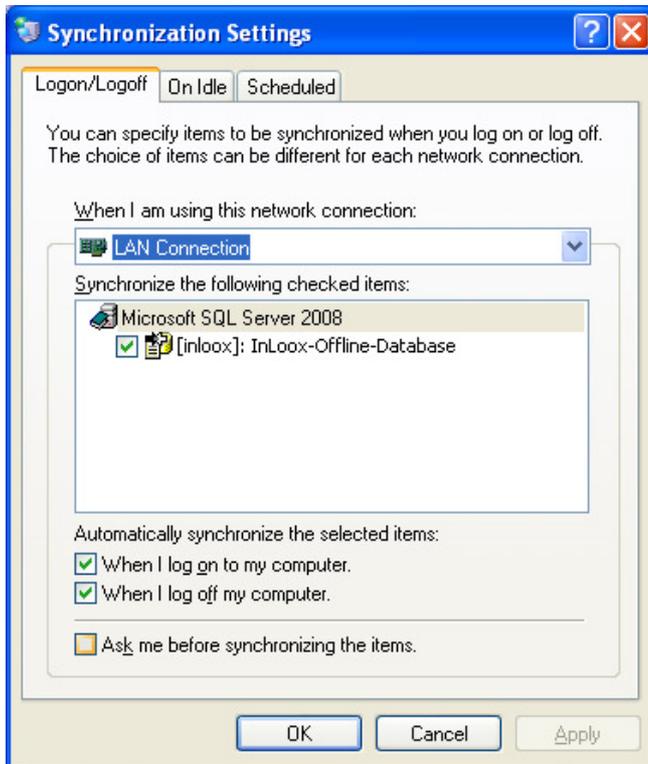
Proceed as follows:

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

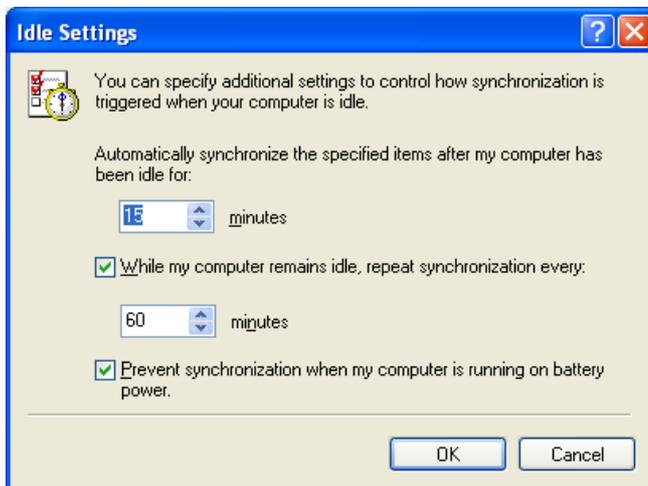


4. If you have to synchronize only one database, you can click immediately on **synchronizing**, otherwise select the desired object.

5. 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



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

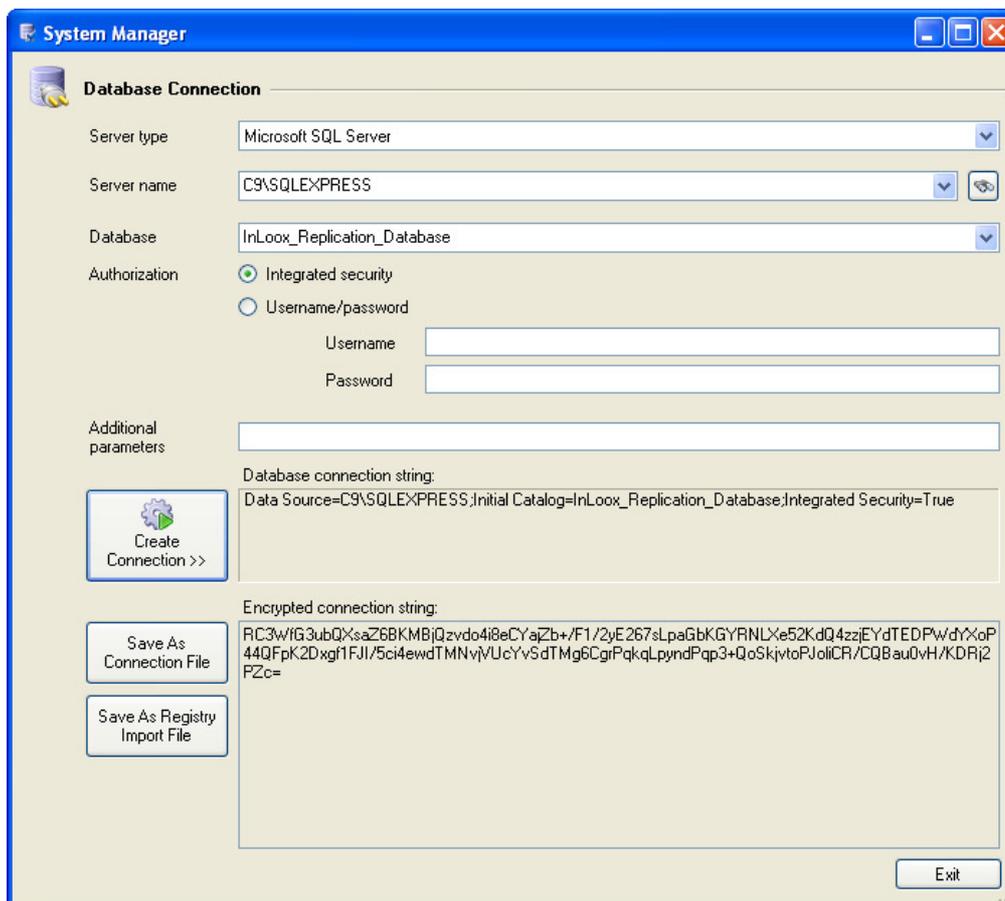


7. Furthermore you can automate the synchronization over **Planned Tasks**

## Changing InLoox database connection on the Client

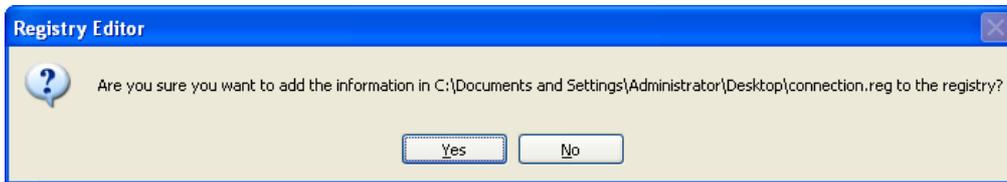
Finally, you have to change the database connection of the InLoox clients. The connection data is stored in the Windows Registry. You have to change the database pointer, so that in future, InLoox works with the local database replication. Proceed as follows:

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**