Setup Snapshot replication: step by step guide

Alexander Chigrik
chigrik@mssqlcity.com

Introduction

In this article, I want to tell you about some general Microsoft SQL Server replication's topics: replication topologies, replication types, replication agents; and about Snapshot replication: how to check necessary conditions for this replication type, how to backup and restore databases participated in this replication scenario, and how to setting up Snapshot replication step by step.

Because it's only test example, I used only one server to replicate data: Publisher, Subscriber and Distributor databases were resided on the same machine.

General concepts

Replication is the process whereby data is copied between databases on the same server or different servers connected by LANs, WANs, or the Internet.

Microsoft SQL Server replication uses a **publisher**, **distributor** and **subscriber** metaphor.

Publisher is the server or database that sends its data to another server or database.

Subscriber is the server or database that receives data from another server or database.

Distributor is the server that manages the flow of data through the replication system. This server contains the **distribution database**.

Publisher contains **publication/publications**. **Publication** is a collection of one or more articles that is sent to a subscriber server or database.

Article is the basic unit of replication and can be a table or a subset of a table.

Subscription is the group of data that a server or database will receive.

There are **push** and **pull subscriptions**. **Push subscription** is subscription when the publishing server will periodically push transactions out to the subscribing server or database.

Pull subscription is subscription when the subscribing server will periodically connect to the **distribution database** and pull information.

The **Distribution database** is a system database, which is stored on the Distributor and does not contain any user tables. This database is used to store snapshot jobs and

all transactions waiting to be distributed to Subscribers.

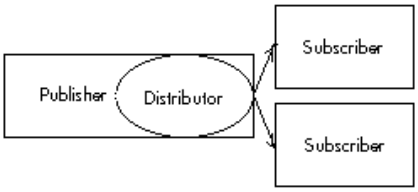
Replication topologies

Microsoft SQL Server supports the following replication topologies:

- Central publisher
- Central subscriber
- Central publisher with remote distributor
- Central distributor
- Publishing subscriber

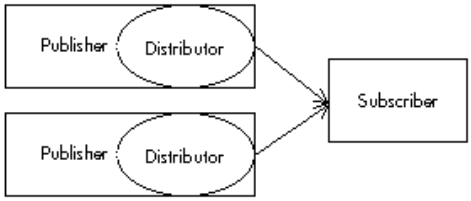
Central publisher

This is one of the most used replication topologies. In this scenario, one server is configured as Publisher and Distributor and another server/servers is/are configured as Subscriber/Subscribers.



Central subscriber

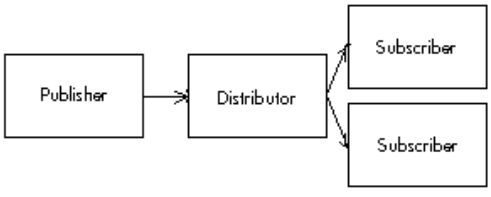
This is a common topology in data warehousing. Many servers or databases replicate their data to a single central server in one or more databases.



Central publisher with remote distributor

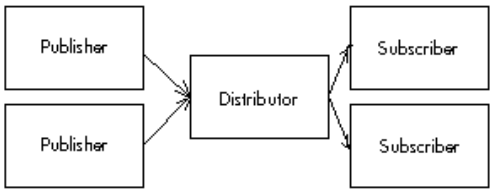
In this topology, distribution database resides on another server than publisher. This topology uses for performance reasons when the level of replication activity increases or the server or network resources become constrained. It reduces Publisher loading, but it increases overall network traffic.

This topology requires separate Microsoft SQL Server installations, one for the Publisher and one for the Distributor.



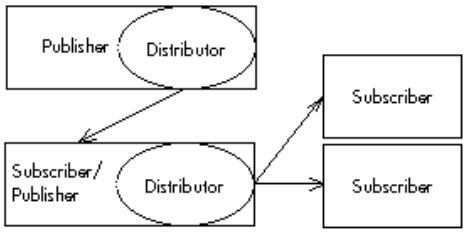
Central distributor

In this topology, several publishers use only one distributor, which resides on another server than publishers. This is one of the most unused replication topologies, because it has only single point of failure (on the single server with central distributor), and if distributor's server will fail, entire replication scenario will be destroyed.



Publishing subscriber

This is a dual role topology. In this topology, two servers publish the same data. One publishing server sends data to subscriber, and then this subscriber publish data to any number of other subscribers. This is useful when a Publisher should send data to Subscribers over a slow or expensive communications link.



Replication types

Microsoft SQL Server 7.0/2000 supports the following replication types:

- Snapshot
- Transactional
- Merge

Snapshot replication is a simplest type of replication. With this kind of replication, all replicated data (replica) will be copied from the Publisher database to Subscriber/Subscribers database on a periodic basis. Snapshot replication is best used as a method for replicating data that changes infrequently and when the size of replicated data is not very large.

With Transactional replication, SQL Server captures all changes that were made in the articles and stores INSERT, UPDATE, and DELETE statements in the distribution database. This changes then sent to subscribers from the distribution database and applied in the same order. Transactional replication is best used when the replicated data changes frequently or when the size of replicated data is not small and is not necessary to support autonomous changes the replicated data on the Publisher and on the Subscriber.

Merge replication is a most difficult replication type. It allows making autonomous changes to replicated data on the Publisher and on the Subscriber. With Merge replication, SQL Server captures all incremental data changes in the source and in the target databases, and reconciles conflicts according to rules you configure or using a custom resolver you create. Merge replication is best used when you want to support autonomous changes the replicated data on the Publisher and on the Subscriber.

Replication agents

Microsoft SQL Server 7.0/2000 supports the following replication agents:

- Snapshot Agent
- Log Reader Agent
- Distribution Agent
- Merge Agent

The Snapshot Agent is a replication agent that makes snapshot files, stores the snapshot on the Distributor, and records information about the synchronization status in the distribution database. The Snapshot Agent is used in all replication types (Snapshot, Transactional, and Merge replications), and can be administered by using SQL Server Enterprise Manager.

The Log Reader Agent is a replication agent that moves transactions marked for replication from the transaction log on the Publisher to the distribution database. This replication agent is not used in Snapshot replication.

The Distribution Agent is a replication agent that moves the snapshot jobs from the distribution database to Subscribers, and moves all transactions waiting to be distributed to Subscribers. The Distribution Agent is used in Snapshot and Transactional replications, and can be administered by using SQL Server Enterprise Manager.

The Merge Agent is a replication agent that applies initial snapshot jobs from the publication database tables to Subscribers, and merges incremental data changes that have occurred since the initial snapshot was created. The Merge Agent is used only in Merge replication.

Checking necessary conditions

Check the following before setting up Snapshot replication:

1. The Localsystem account has no access to shares on the network as it isn't an authenticated network account. So, if you want to setting up replication you must change the account the MSSQLServer and SQLServerAgent services runs under to a account with the Windows NT/Windows 2000 administrator's rights. If your Microsoft SQL Server runs on Windows NT or Windows 2000, you can create Windows NT/Windows 2000 account and include it into local Administrators group, into Domain Users group, and set *Log in as a service* permission for this account.

Because Windows 9x does not support Windows NT services, so if your Microsoft SQL Server runs on Windows 9x, you do not need to create SQL account.

2. Only members of the sysadmin server role can setting up and configure replication, so if you have not these rights, you cannot setting up replication.

3. Don't forgive to start SQLServerAgent service (and MSSQLServer service, of course).

4. You should allocate adequate disk space in the snapshot folder.

5. You should allocate adequate disk space for the distribution database.

6. You should ensure the server that is being replicated to, is defined as a remote server.

Step by step example

In this example, I will use only one server to replicate data: Publisher, Subscriber and Distributor databases will be resided on the same machine. I will use Snapshot replication with push subscription. To setting up Snapshot replication, you can use GUI interface (from the SQL Server Enterprise Manager), or you can run SQL Server system stored procedures. The first way is much easy and much understandably, so I will use it.

First of all, you should register the new remote server in which will be replicated. Because, I use only one server to replicate data, I don't need to make this step. Figure 1 shows remote server name (the same as the local server name, in this case).

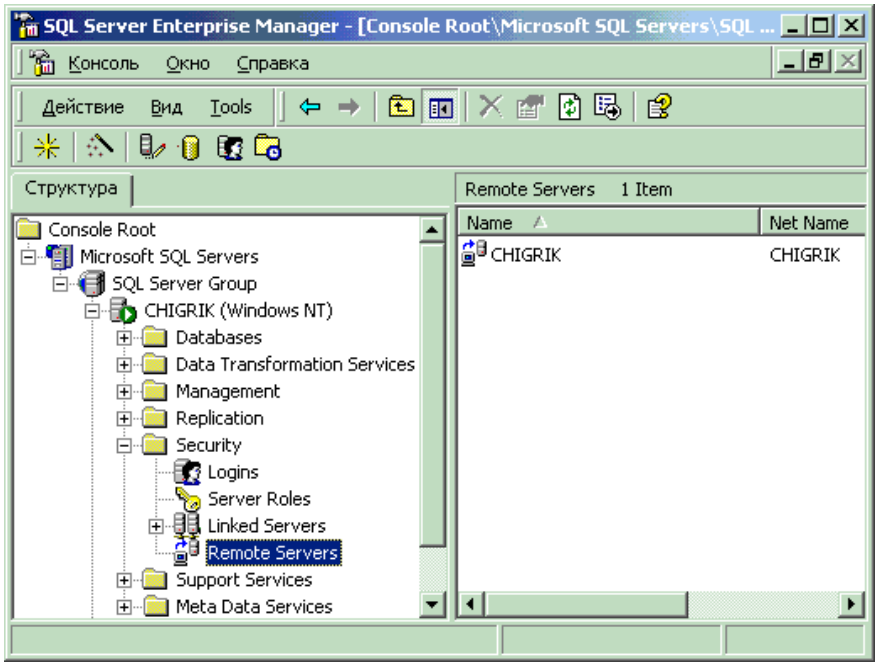


Figure 1.

In this example, I will replicate data from the pubs database into pubs_copy database.

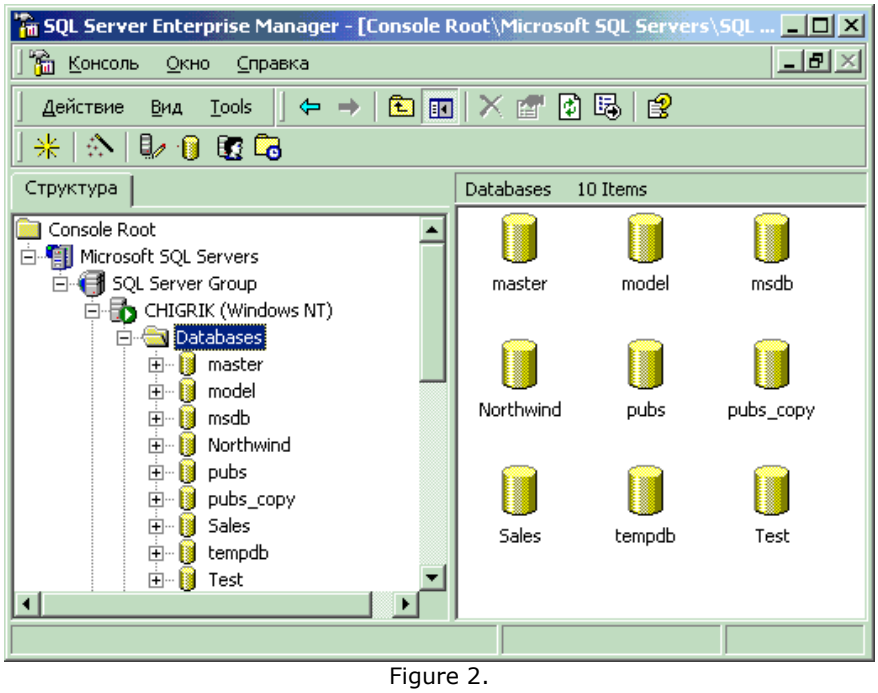


Figure 2.

Select

Tools =>
Replication =>
Configure Publishing, Subscribers, and Distribution...

as shown in Figure 3.

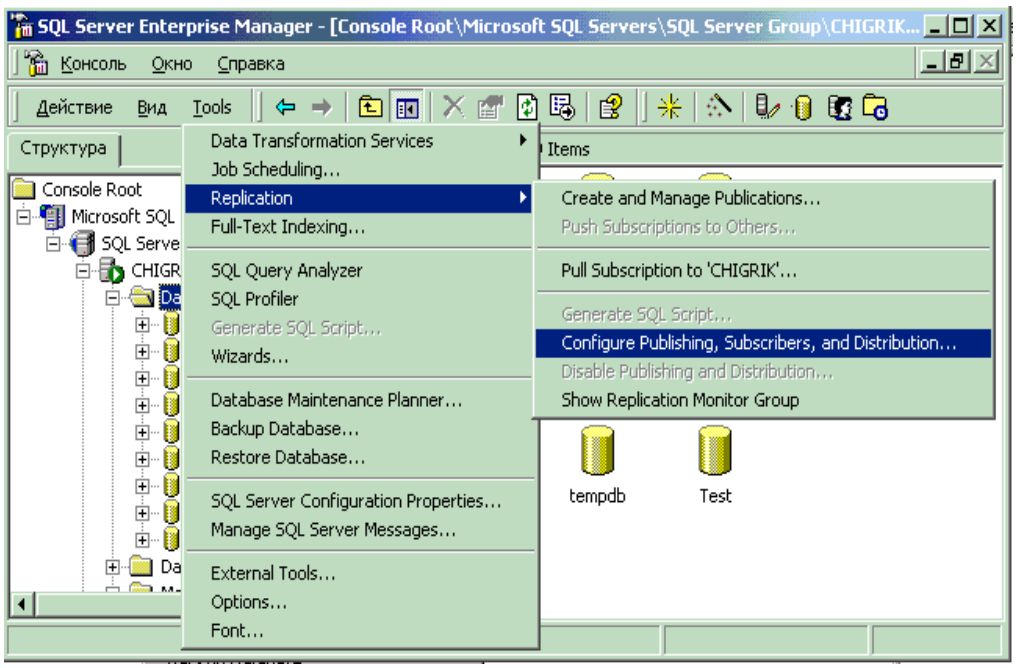


Figure 3.

This will launch the Configure Publishing and Distribution Wizard, as shown in Figure 4.

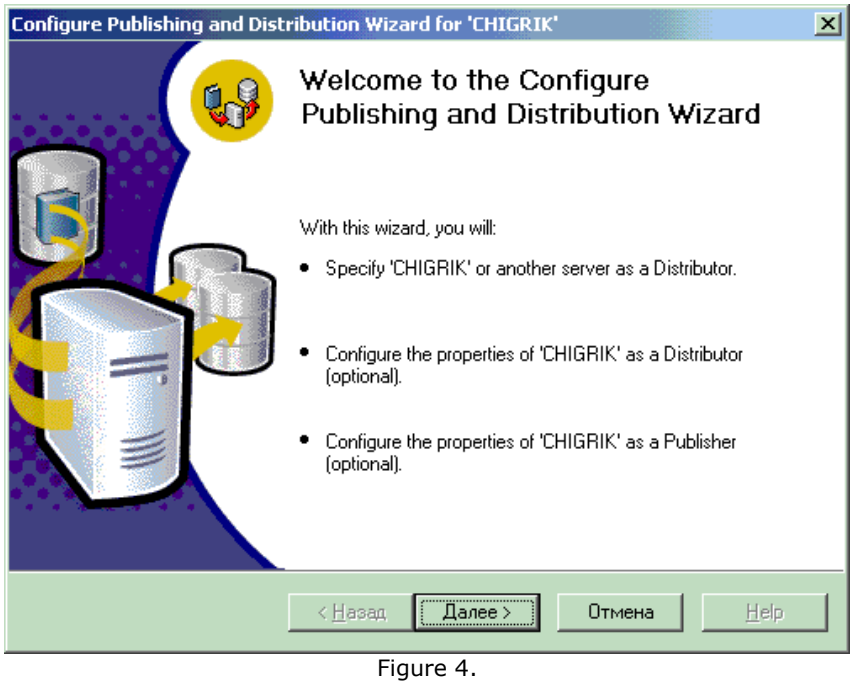


Figure 4.

Select the Next button to create the Distributor, as shown in Figure 5.

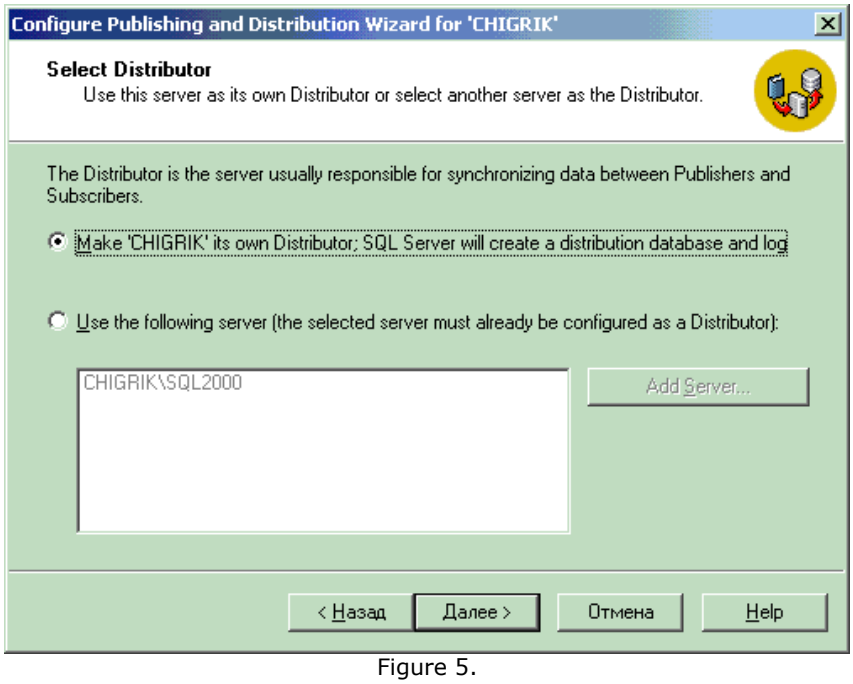


Figure 5.

Here you can configure SQLServerAgent service to start automatically when the computer is started. Check
Yes, configure the SQL Server Agent service to start automatically
and click the Next button, as shown in Figure 6.

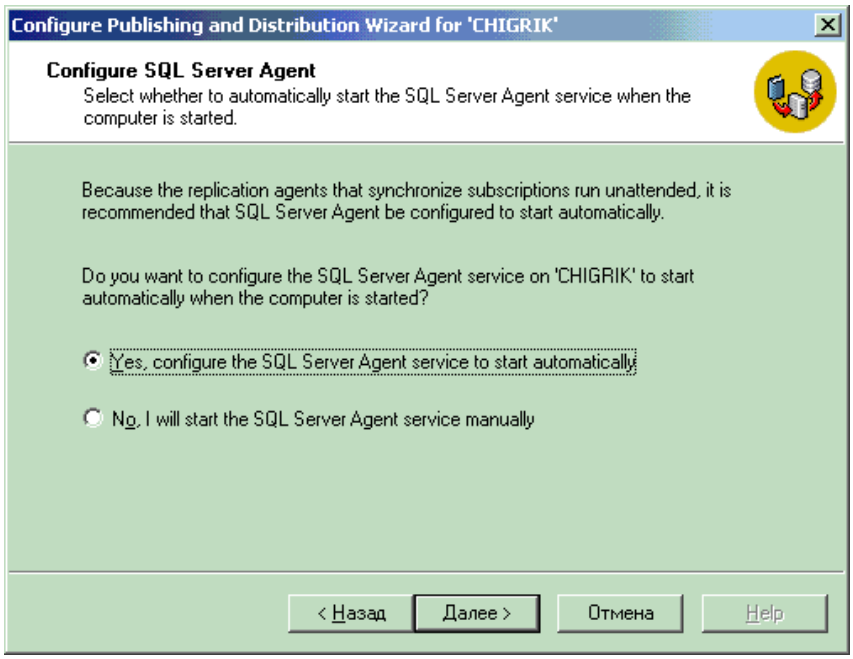


Figure 6.

Specify snapshot folder using a network path and click the Next button, as shown in Figure 7.

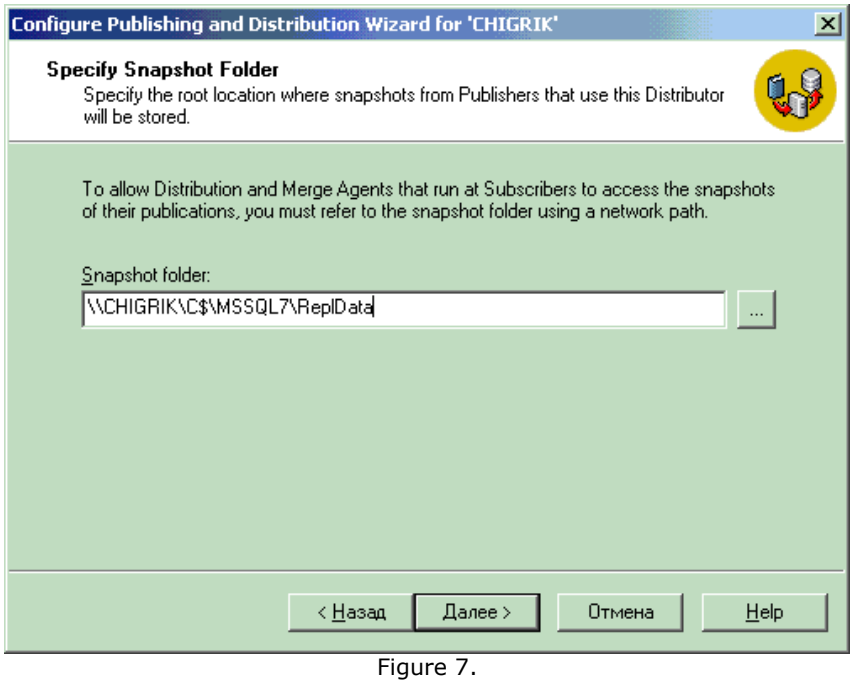


Figure 7.

Now you can customize the publishing and distribution settings, or you can choose the default settings. Check *No, use the following default settings* and click the Next button, as shown in Figure 8.

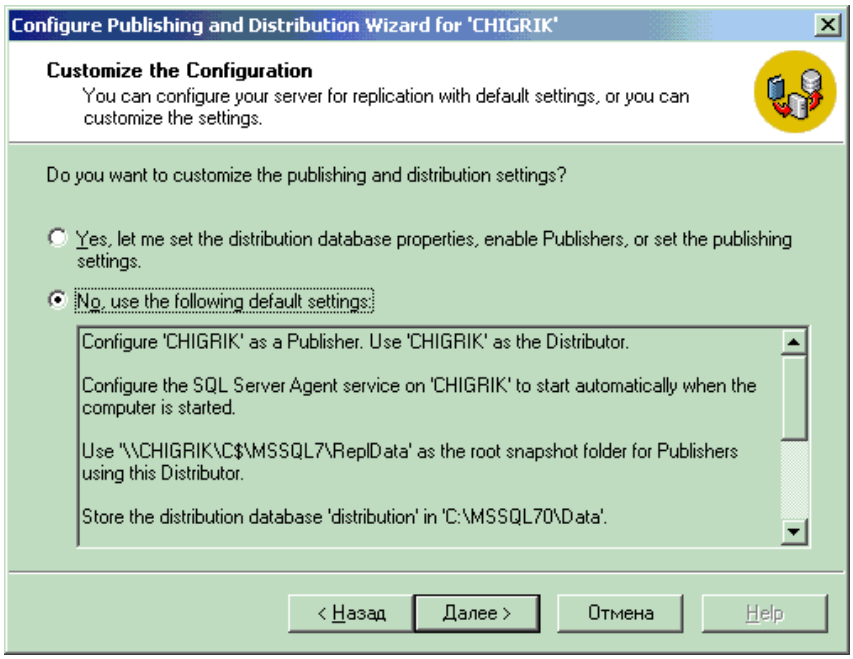


Figure 8.

Click the Finish button, as shown in Figure 9.

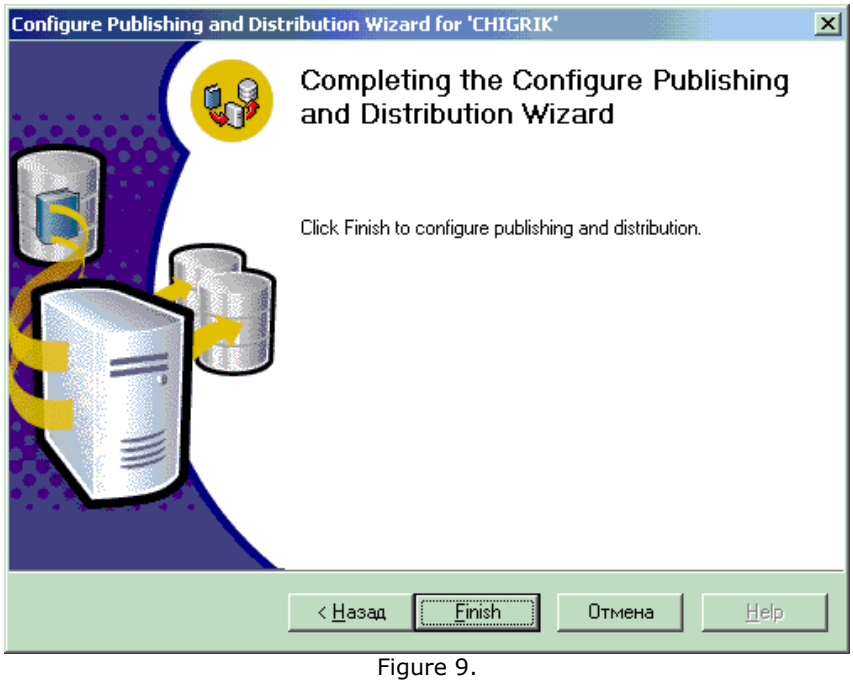


Figure 9.

Microsoft SQL Server created the distribution database, enabled publishing, and installed the distributor. Once completed, you should see Figure 10.

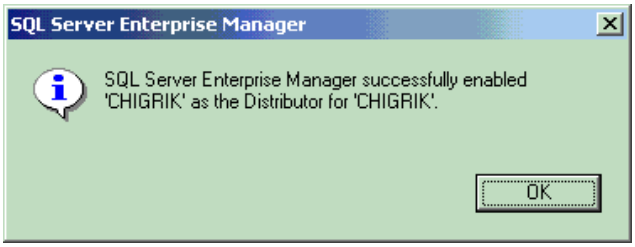


Figure 10.

Click OK button and see Figure 11. As we installed CHIGRIK as Distributor, so Replication monitor has been added to the console tree on CHIGRIK server. Click Close button.

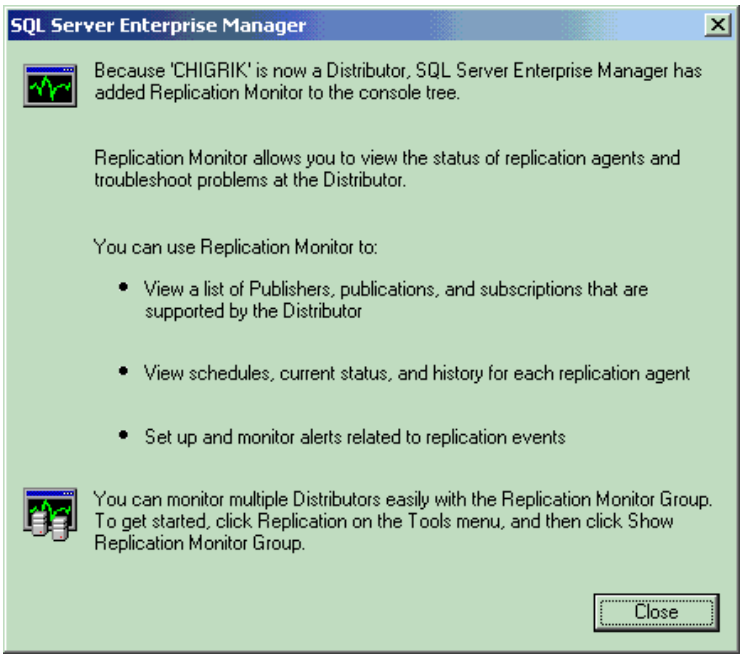


Figure 11.

Now we are ready to start creating publications and articles. Select Tools => Replication => Create and Manage Publications as shown in Figure 12.

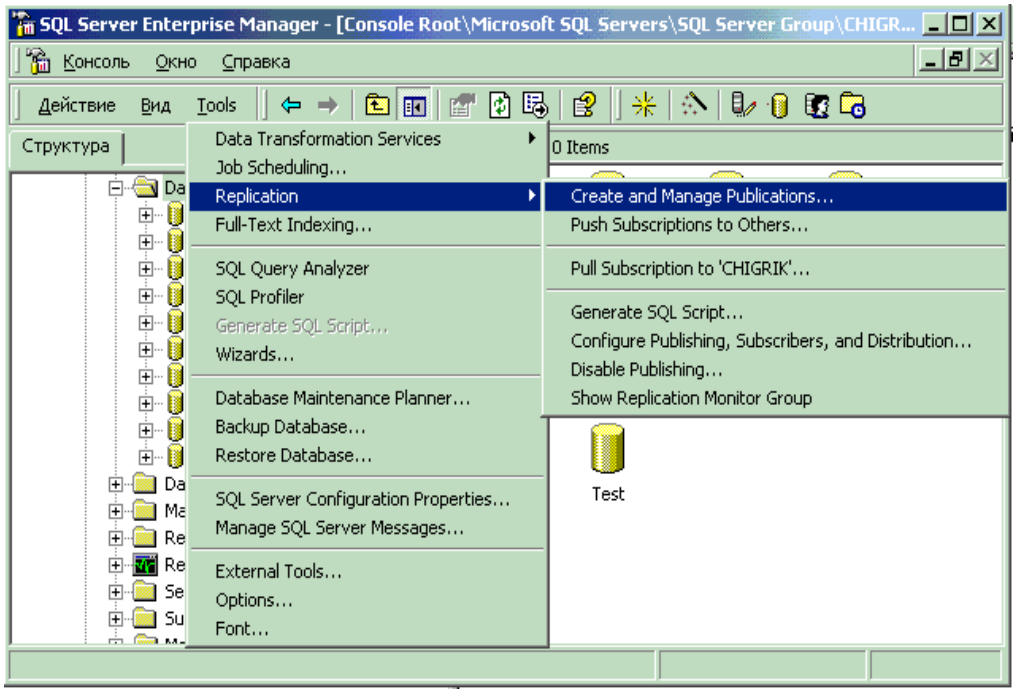


Figure 12.

You will see Create and Manage Publications dialog box, as shown in Figure 13. Choose pubs database and click the Create Publication button.

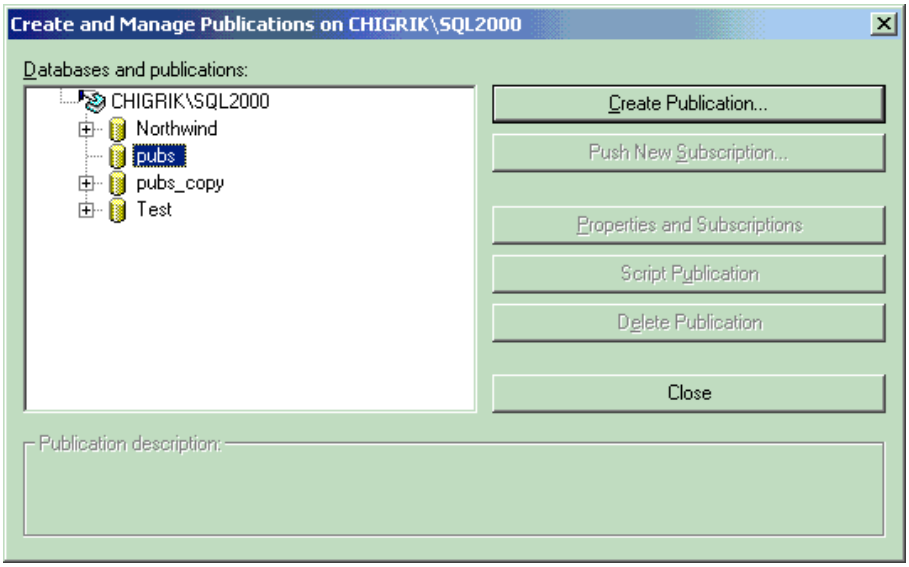


Figure 13.

The Create Publication wizard will be launch.
Click the Next button, as shown in Figure 14.

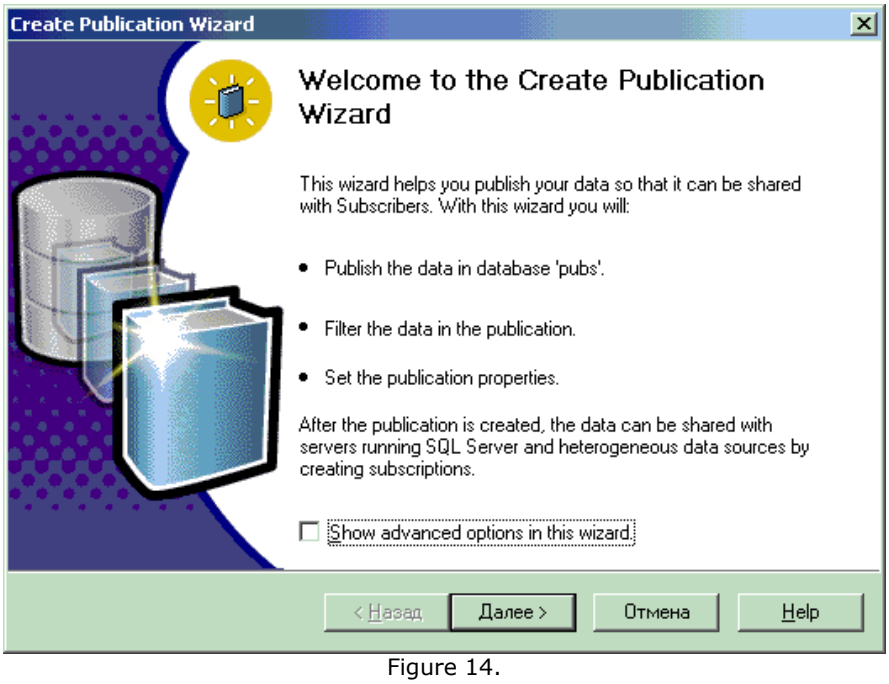


Figure 14.

Choose the pubs database and click the Next button, as shown in Figure 15.

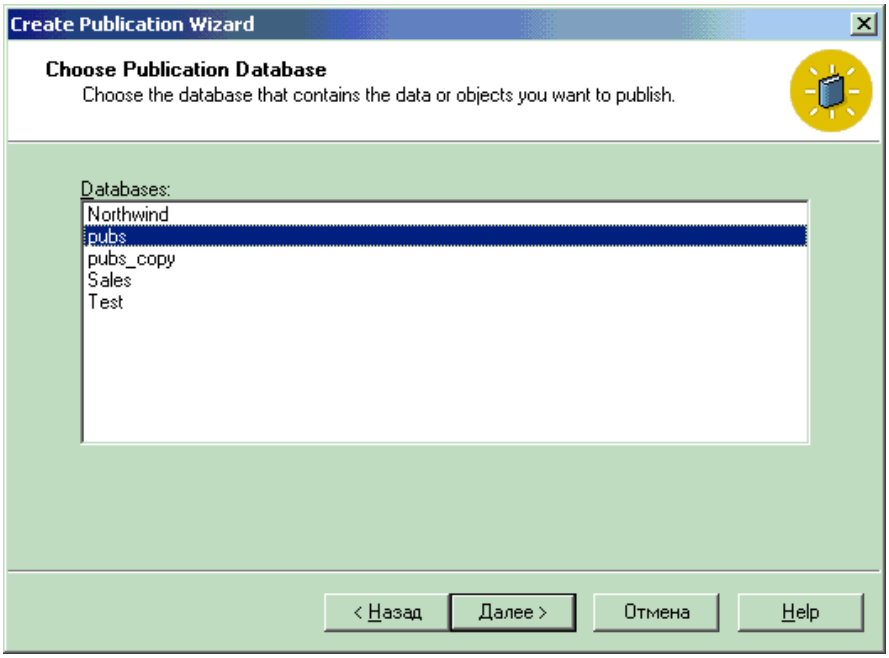


Figure 15.

Select Snapshot publication and click the Next button, as shown in Figure 16.

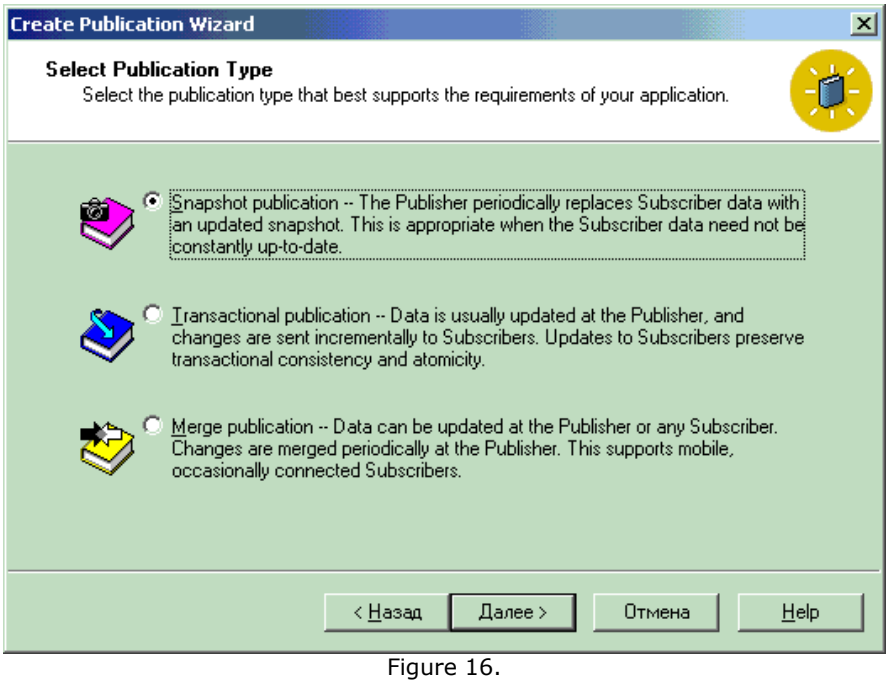


Figure 16.

Select all of the types of Subscribers that you expect to subscribe to this publication and click the Next button, as shown in Figure 17.

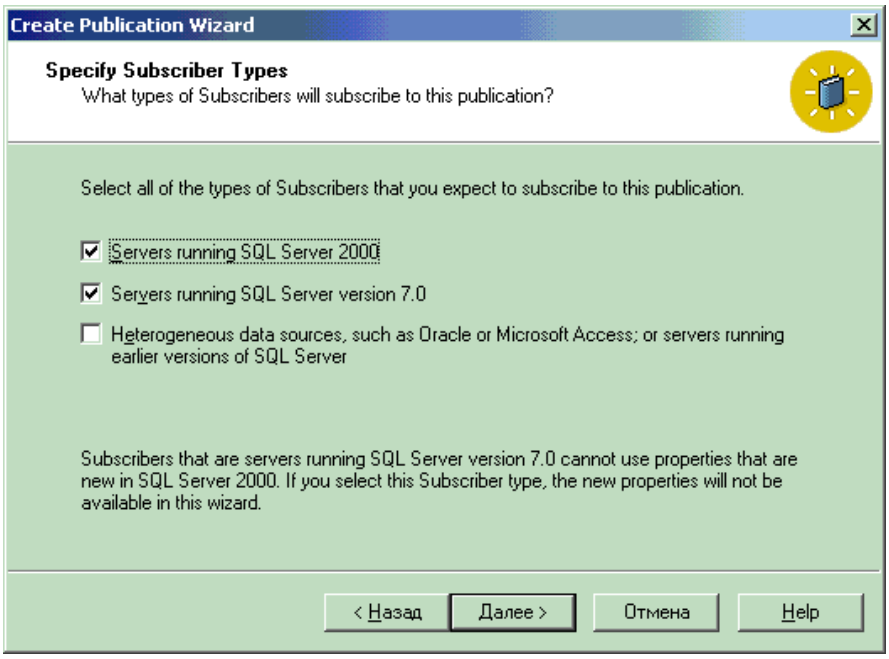


Figure 17.

Choose authors table to publish as article and click the Next button, as shown in Figure 18.

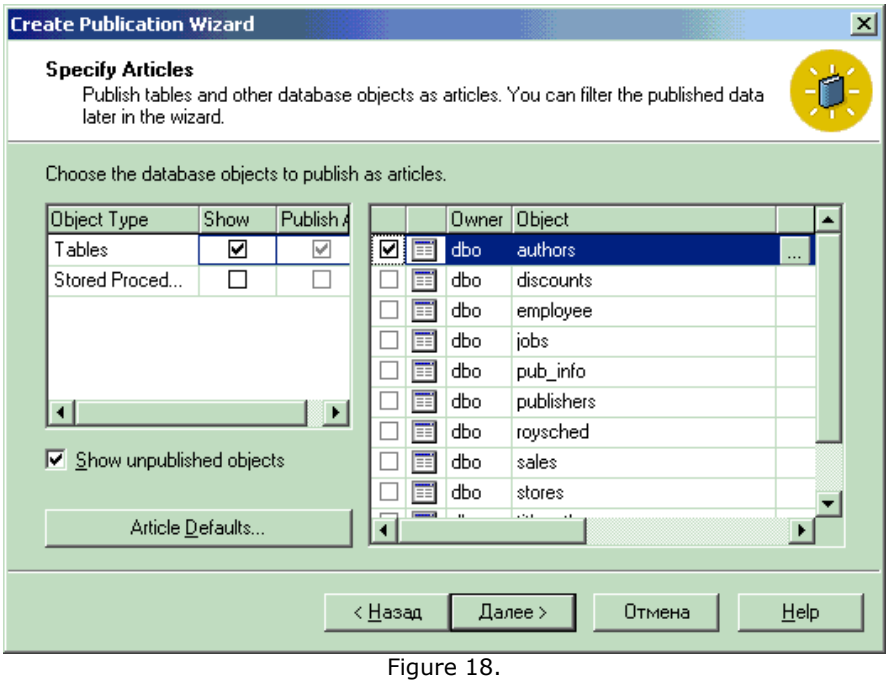


Figure 18.

Specify pubs_article as the publication name and click the Next button, as shown in Figure 19.

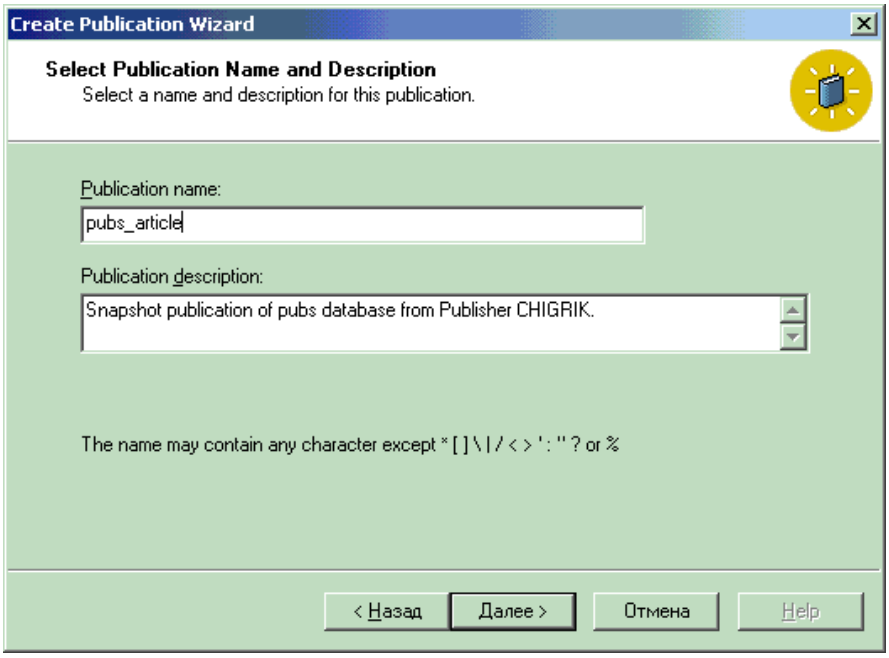


Figure 19.

You can specify data filters on this step, but in this example, we don't use any data filters. Check *No, create the publication as specified* and click the Next button, as shown in Figure 20.

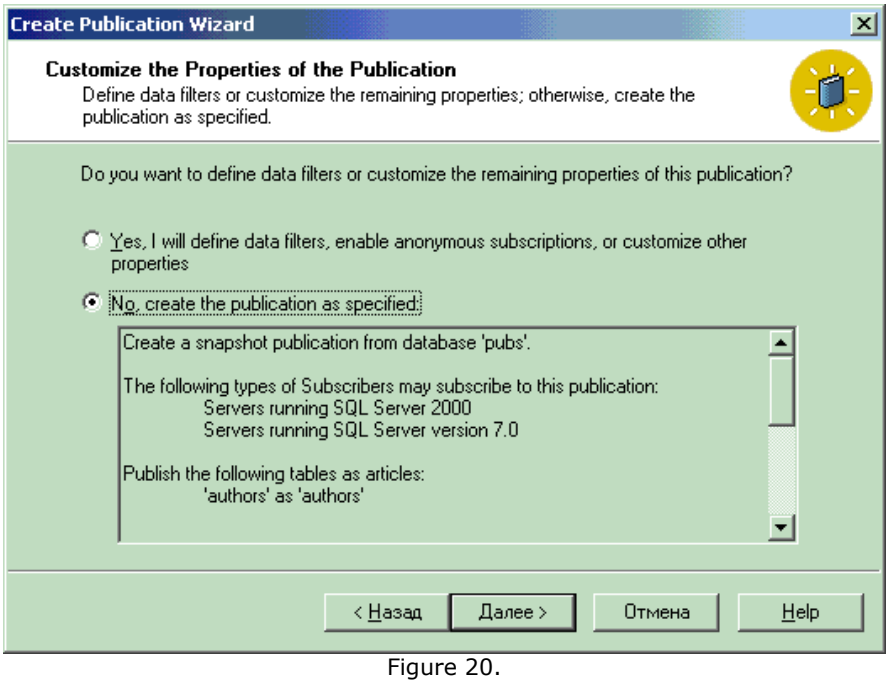


Figure 20.

Click the Finish button to create the publication, as shown in Figure 21.

Figure 21.

Now the 'pubs_article' publication was created, so click the Close button, as shown in Figure 22.

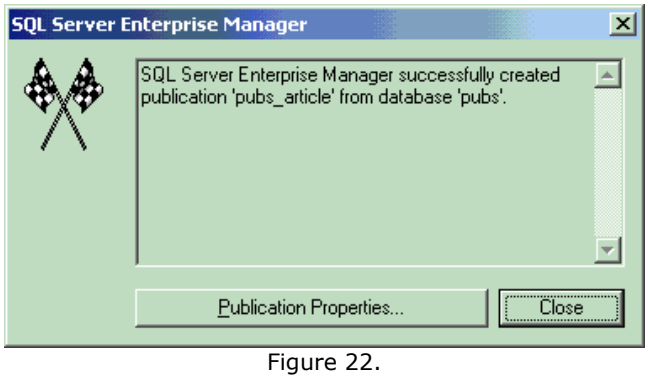


Figure 22.

Now you can create new subscription. Click the Push New Subscription button, as shown in Figure 23.

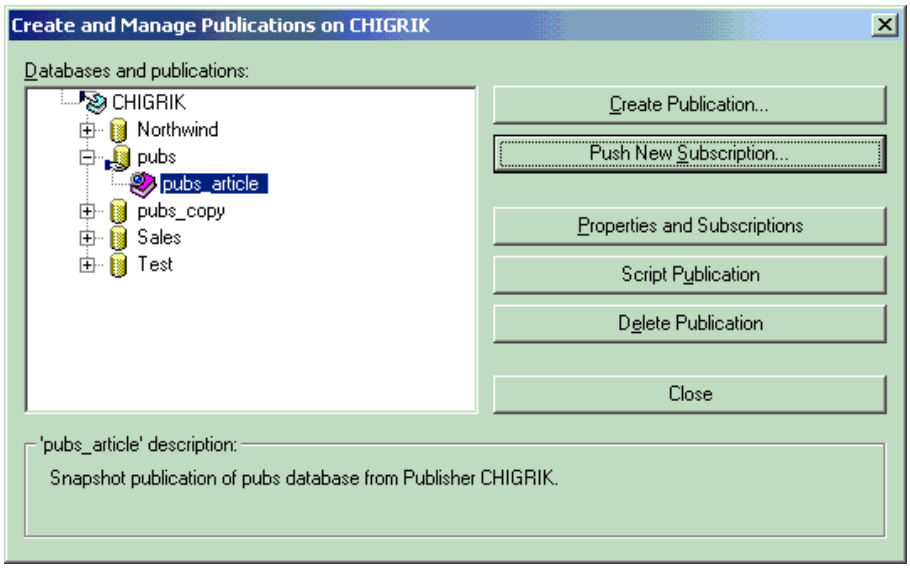


Figure 23.

This will launch the Push Subscription wizard shown in Figure 24. Click the Next button.

Figure 24.

Select CHIGRIK to select all subscribers in that group and click the Next button, as shown in Figure 25.

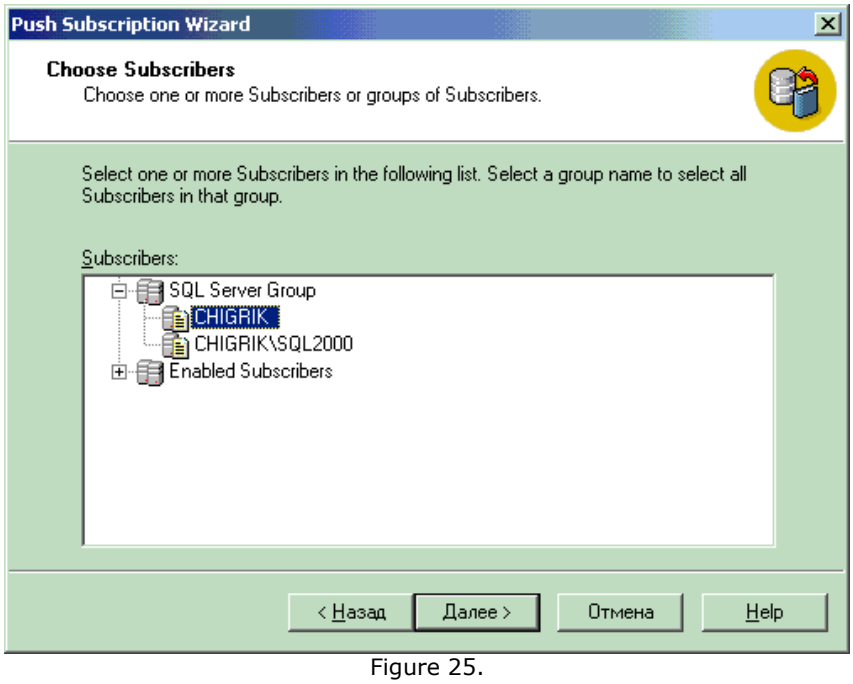


Figure 25.

Select pubs_copy database as the subscription database and click the Next button, as shown in Figure 26.

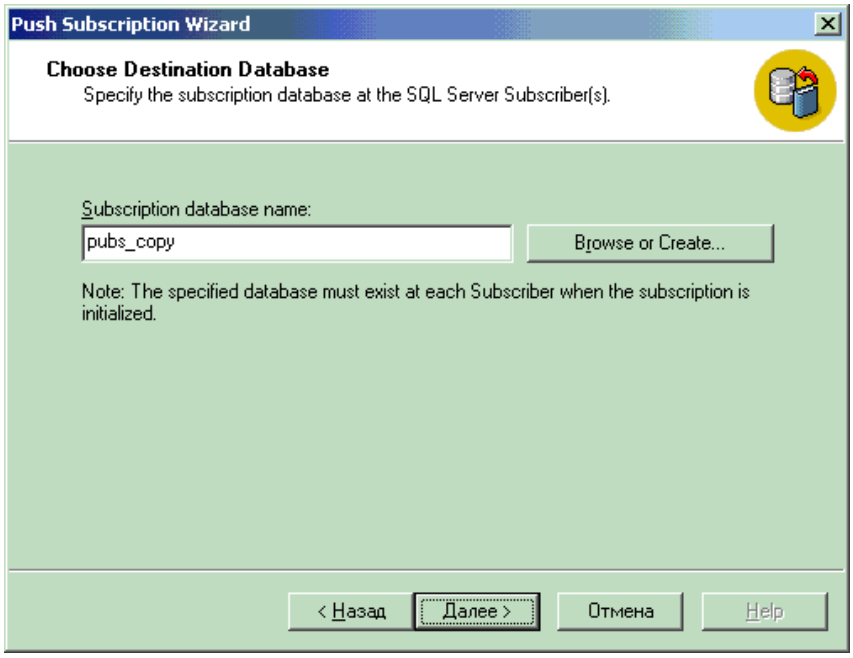


Figure 26.

Specify how frequently Distribution Agent updates the subscription (in this example, every 1 day(s), every 20 minute(s) between 9:00:00 and 18:00:00) and click the Next button, as shown in Figure 27.

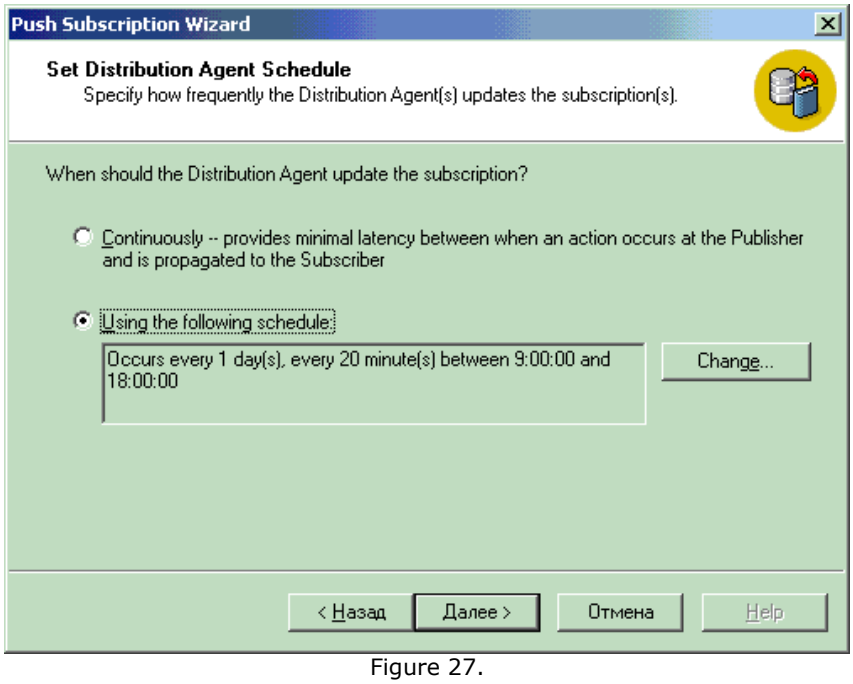


Figure 27.

Check *Start the Snapshot Agent to begin the initialization process immediately* and click the Next button, as shown in Figure 28.

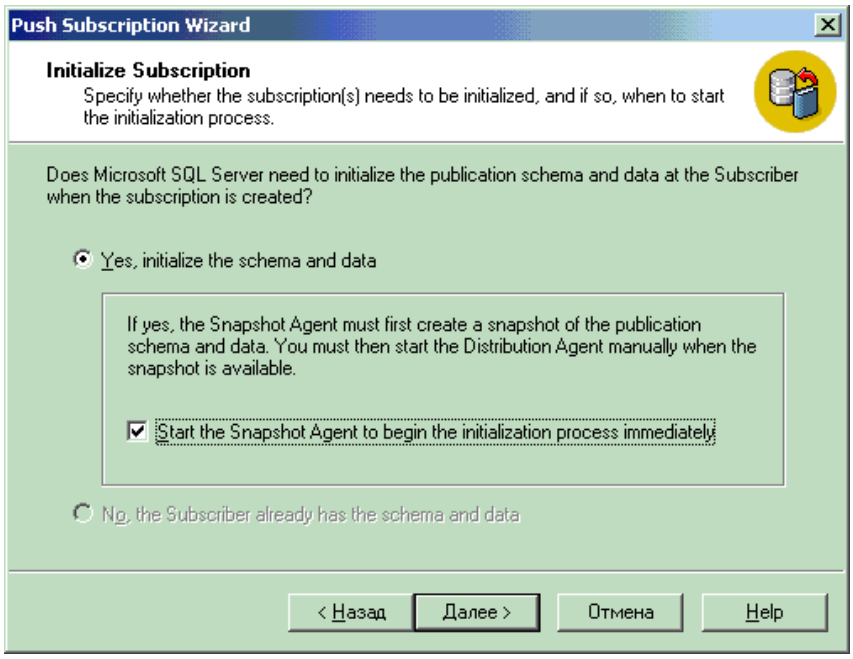


Figure 28.

Click the Next button, as shown in Figure 29.

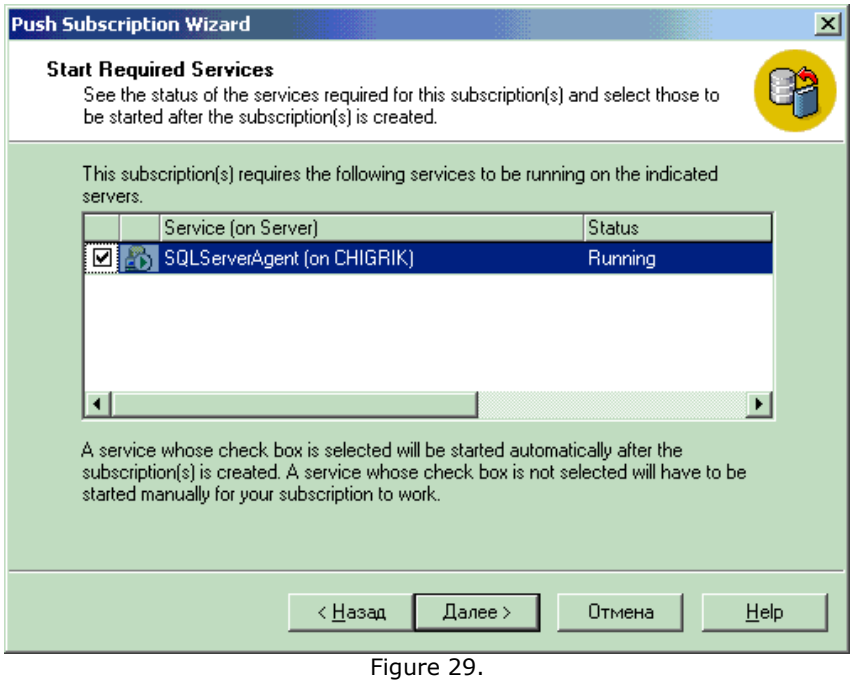


Figure 29.

Click the Finish button to subscribe with the options you specified on the previous steps, as shown in Figure 30.

Figure 30.

Click the Close button, as shown in Figure 31.

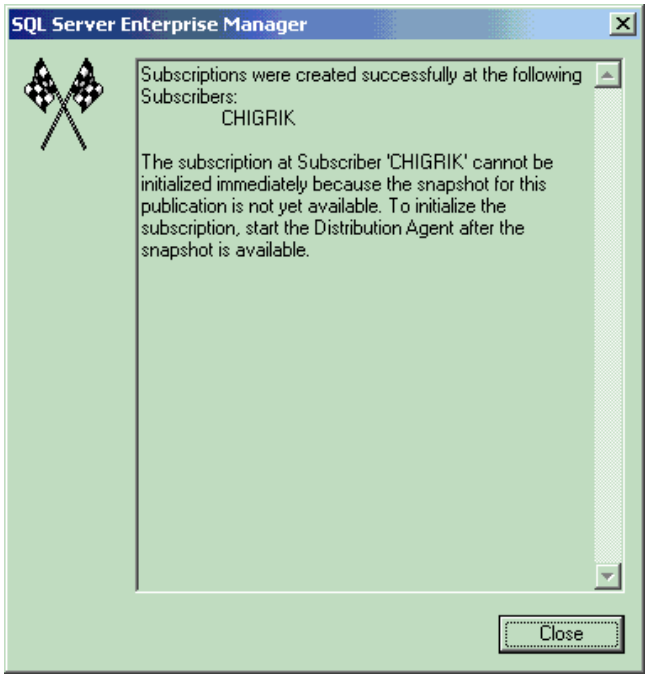


Figure 31.

Click the Close button to close the Create and Manage Publication dialog, as shown in Figure 23.

Note. On the last step we got the following message:

```
The subscription at Subscriber 'CHIGRIK' cannot be initialized
immediately because the snapshot for this publication is not yet
available. To initialize the subscription, start the Distribution
Agent after the snapshot is available.
```

To work around it, you should start the Distribution Agent after the snapshot is available. See Figure 32.

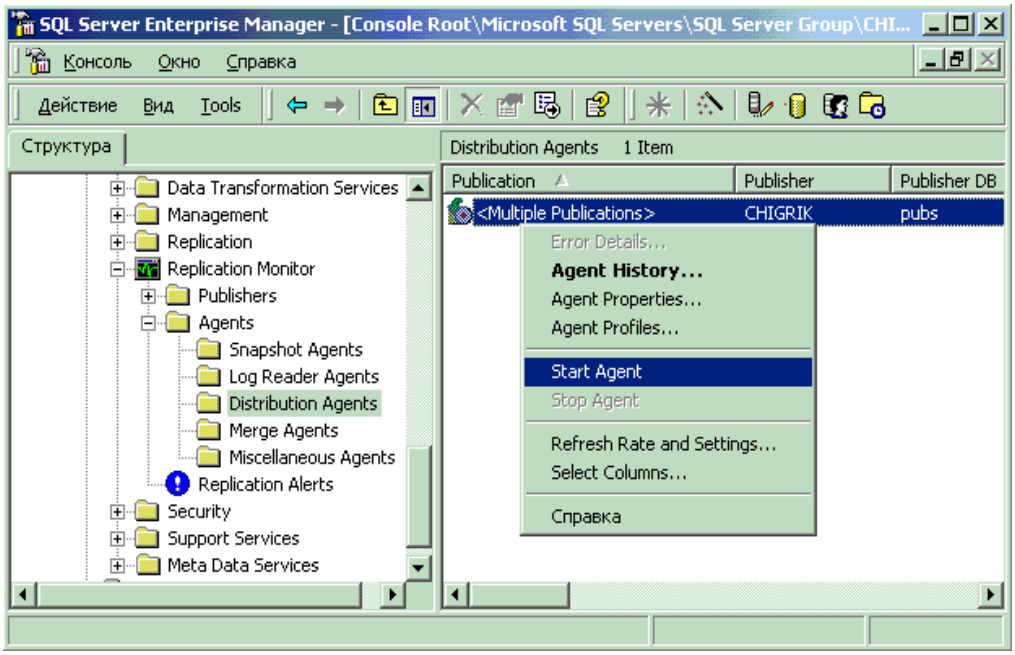


Figure 32.

Backup and restore strategies

The backup and restore strategies are differ for each replication types. Here, I want to describe the backup and restore strategies for the Snapshot replication.

There are four main strategies for backing up and restoring Snapshot replication:

- Backup Publisher, master and model databases.
- Backup Publisher, Distributor, master and model databases.
- Backup Publisher, Subscriber(s), master and model databases.
- Backup Publisher, Distributor, Subscriber(s), master and model databases.

Backing up Publisher, master and model databases is a simplest strategy. This strategy has its own advantages and disadvantages. The advantages are that it requires the least amount of storage resources and does not require coordinating the backup with the backup of any other servers. The main disadvantage of this strategy is that you may need to setup replication from the beginning in the event of a Publisher or Distributor failure. With this strategy, you should backup publication database after changing existing publications or after the new publications were added.

Backing up Publisher, Distributor, master and model databases is a more frequently used strategy, than the first one, because in this case you don't need to reestablish replication in the event of a Publisher or Distributor failure. The main disadvantage of this strategy is that you should backup Publisher and Distributor's databases simultaneously (or as closely as possible). It also requires more computing and storage resources than the first way.

Backing up Publisher, Subscriber(s), master and model databases let you to reduce the amount of time required to recover a Subscriber(s) by avoiding the need to reinitialize the Subscriber(s) with a new snapshot, but in the event of a Distributor failure, you should setup replication from the beginning.

Backup Publisher, Distributor, Subscriber(s), master and model databases is a most complex backup strategy. The main advantage of this strategy is that in the event of a Publisher, Distributor or Subscriber(s) failure, you can quickly restore fail database without setting up replication from the beginning. The disadvantage of this strategy is that you should backup Publisher and Distributor's databases simultaneously (or as closely as possible), and this strategy requires most computing and storage resources.

For each strategy, you should backup msdb and master databases on the Publisher, Distributor and Subscriber(s). msdb database is used by SQL Server Agent for scheduling alerts and jobs (it contains also snapshot jobs), and master database is a main system database contains entries for each Subscriber(s), each login account, about system configuration settings and so on.

Literature

1. SQL Server Books Online.

2. How Snapshot Replication Works

3. Planning for Snapshot Replication

4. Enhancing Snapshot Replication Performance

5. Snapshot Replication

6. Strategies for Backing Up and Restoring Snapshot Replication

7. Replication Types