

REPLICAÇÃO DE DADOS

- SQL Server 2000 -

INTRODUÇÃO

Replicação é o processo que mantém duas ou mais réplicas dos dados em diferentes servidores SQL Server 2000. Logo, pode-se replicar desde um conjunto de linhas ou colunas de uma tabela até o próprio Banco de dados inteiro. No caso do SQL Server, temos várias funções e assistentes que permitem configurar este processo de replicação de dados. Quando é configurada a replicação, o SQL Server se encarrega de manter sincronizadas todas as réplicas.

A confiabilidade da replicação é alta, levando ainda em consideração, que ele é capaz de reiniciar do ponto onde foi interrompido, caso exista alguma falha de comunicação na rede ou quaisquer outras falhas. Portanto, o SQL Server é capaz de garantir a integridade dos dados replicados.

MODELOS DE REPLICAÇÃO NO SQL SERVER

O SQL Server utiliza um modelo de replicação, muitas vezes chamado de metáfora de replicações.

- ⇒ **Publisher**
- ⇒ **Distributor**
- ⇒ **Subscribers**
- ⇒ **Article**

Em uma visão geral, podemos dizer que o Publisher, ou publicador, é quem publica informações a serem replicadas para um ou mais Subscribers, que podemos chamar de assinantes. Entre o publicador e os assinantes, ainda pode existir o Distributor, ou distribuidor, que faz o envio dos dados para os assinantes.

Publisher (Publicador) – Um servidor atuando como Publisher é um servidor que está disponibilizando dados para serem replicados para outros servidores. O Publisher mantém a cópia original dos dados e envia o conjunto de dados a ser replicado para um ou mais Distribuidores. O publicador disponibiliza conjuntos definidos de dados através de publicações. Podemos ter uma ou mais publicações, cada uma representando um conjunto de dados.

Distributor (Distribuidor) – O distribuidor recebe do Publisher uma cópia dos dados que foram alterados e devem ser replicados e disponibiliza estes dados para um ou mais assinantes. A função do Distribuidor varia com o tipo de replicação que estamos implementando.

Subscriber (Assinante) – Um Assinante é um servidor SQL Server que armazena uma réplica dos dados e recebe atualizações periódicas dos mesmos. Os dados recebidos pelo assinante podem ser alterados em determinadas configurações.

Article (Artigo) – Um artigo pode ser uma tabela de dados ou um objeto de Banco de Dados, como uma Stored Procedure que é configurado para replicação. Podemos ter um artigo que engloba uma tabela inteira, ou um artigo, que engloba apenas algumas colunas, chamando de filtragem vertical.

Com isso, compreendemos que é através da criação de um artigo, que definimos quais dados devem ser replicados. Isto evita que tenhamos que replicar uma tabela inteira quando apenas um subconjunto dos dados é necessário.

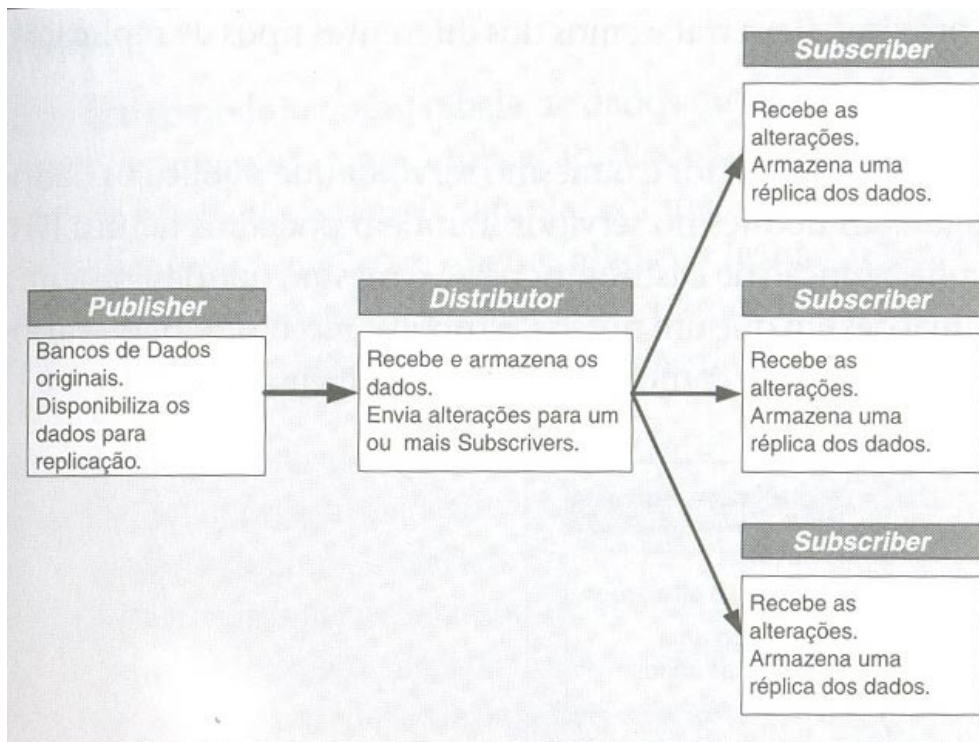


Figura 1 – Visão geral do modelo de replicação

Enfim, com todos esses conceitos bem definidos o SQL Server os trata através de uma publicação que nada mais é do que um conjunto e um ou mais artigos de um Banco de Dados. A publicação é a unidade de replicação. Trocando em miúdos, o Assinante inscreve-se em uma Publicação e não em um artigo. Mesmo que somente tenhamos que replicar dados de um único artigo, devemos criar uma Publicação que inclui esse artigo.

Podemos criar uma ou mais Publicações em um Banco de Dados, porém uma Publicação não pode conter artigos de diferentes Bancos de Dados.

SUBSCRIÇÕES E SEUS TIPOS

Chamamos de subscrição o processo pelo qual um servidor se inscreve para receber cópia dos dados de uma ou mais Publicações. E, ao se inscrever para isso, o servidor torna-se um assinante, ou assinante.

Sendo que existem dois tipos de subscrições: Pull subscription e Push Subscription. O tipo de subscrição vai definir como a mesma será criada e administrada e também de que maneira os dados serão replicados.

Push Subscription – Este tipo de subscrição é feita durante a criação ou administração de uma Publicação. Neste tipo de subscrição, o Distribuidor é responsável por enviar as alterações para ou mais Assinantes. Utilizamos este método quando queremos ter um controle centralizado sobre o agendamento das replicações. É muito fácil configurar várias subscrições em um único servidor, do que ter que configurar a subscrição em diversos servidores.

Pull Subscription – Neste, a subscrição é feita no Assinante, que tomará a iniciativa de solicitar o início da replicação e envio dos dados do Distribuidor para o Assinante. Neste caso, o controle e agendamento da replicação, fica a cargo de cada Assinante. A administração fica mais distribuída, e ao mesmo tempo, damos maior liberdade para cada Assinante.

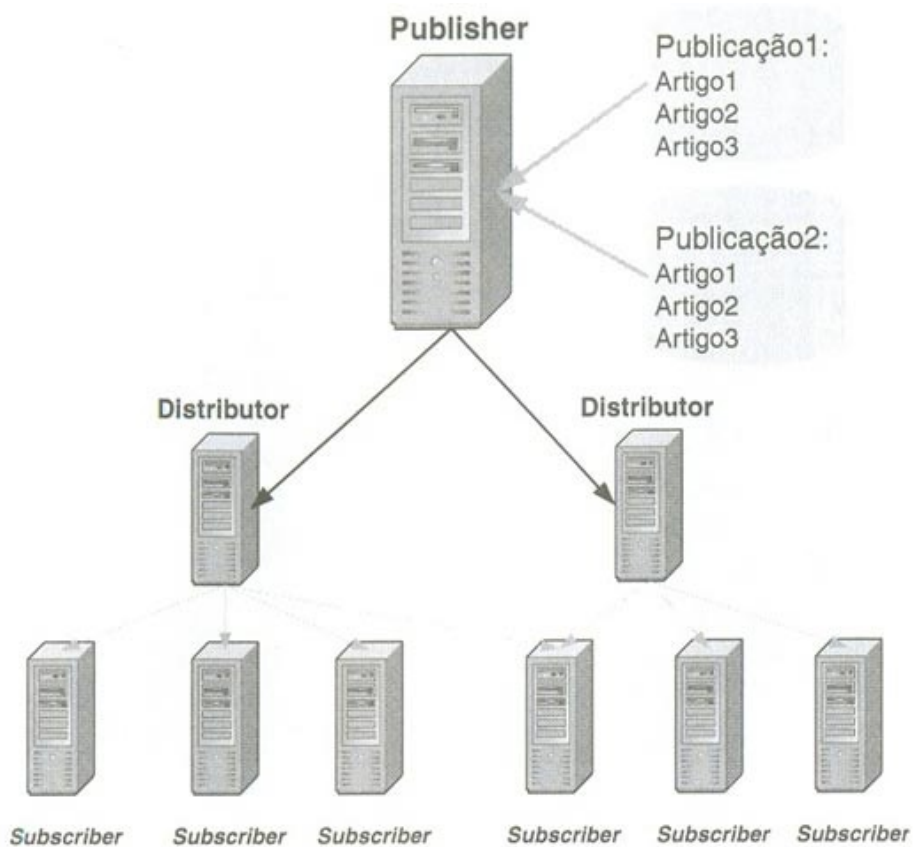


Figura 2 – Modelo de Replicação com todos os autores

TIPOS DE REPLICAÇÃO

Snapshot replication – Todo o conjunto de dados é replicado a cada intervalo programado. Um ou mais assinantes são atualizados com uma cópia exata dos dados publicados pelo Publisher. São enviados, para os assinantes, todos os dados, quer tenham sido modificados ou não.

É o tipo mais simples de replicação, pois os servidores envolvidos na replicação não precisam monitorar os dados para verificar quais alterações foram feitas, uma vez que todos os dados são replicados. A principal desvantagem deste tipo de replicação é que a mesma transmite uma grande quantidade de informações a cada ciclo de replicação. Abaixo, algumas características da Snapshot replication:

- ⇒ Não é indicada para grandes volumes de dados;
- ⇒ Exige pouco do processador, uma vez que não existe a necessidade de um monitoramento constante para verificação das alterações efetuadas no conjunto de dados a ser replicado;
- ⇒ Possui uma latência elevada, sendo a latência igual ao intervalo de replicação. Por exemplo, se uma Snapshot replication está programada para ocorrer uma vez ao dia, a latência será de 24 horas.

Transactional Replication – Nesta replicação as alterações efetuadas no Publisher é que são replicadas para um ou mais Assinantes. Qualquer alteração feita, em um ou mais artigos de uma publicação, é imediatamente capturada a partir do Log de Transações do Banco de Dados e replicada para o Distribuidor. Dependendo das configurações e da velocidade de conexão entre os servidores, podemos manter as diversas réplicas sincronizadas quase que instantaneamente.

A principal vantagem é que podemos manter as diversas réplicas praticamente idênticas, diminuindo bastante a latência (questão de segundos) de atualização entre as diversas réplicas. A grande desvantagem é que este monitoramento contínuo em busca de alterações, implica em uma maior exigência de memória e processamento no servidor SQL Server.

Merge Replication – A principal característica deste tipo de replicação é que alterações podem ser feitas em qualquer uma das replicas dos dados. As alterações feitas em uma das replicas serão repassadas para as demais. O SQL Server utiliza um mecanismo próprio para resolução de conflitos de replicação.

Ao iniciarmos o processo de replicação, temos uma Snapshot replication na primeira vez que os dados são enviados para um ou mais Assinantes. Na seqüência, o mecanismo de replicação fica monitorando as alterações efetuadas nas diversas réplicas e fica repassando as alterações para as demais réplicas. Não temos a garantia de que todas as réplicas estarão perfeitamente sincronizadas instantaneamente. Porém, temos a certeza de que as diversas réplicas irão convergir com o passar do tempo.

REPLICANDO OS DADOS

Para começar a implementação, devemos ter registrado no SQL Server os servidores aos quais queremos usar para a replicação de dados. Conforme, imagem a seguir:

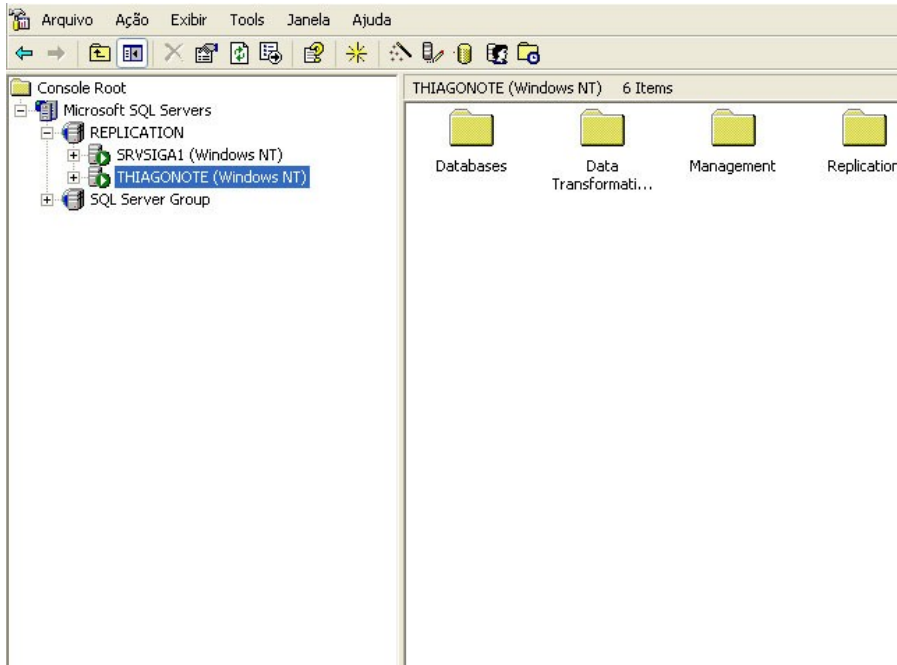


Figura 3 - Lista de servidores SQL Server 2000 registrados

Em seguida, clicamos no servidor onde os dados serão obtidos para a replicação dos dados e criamos uma nova Publicação.

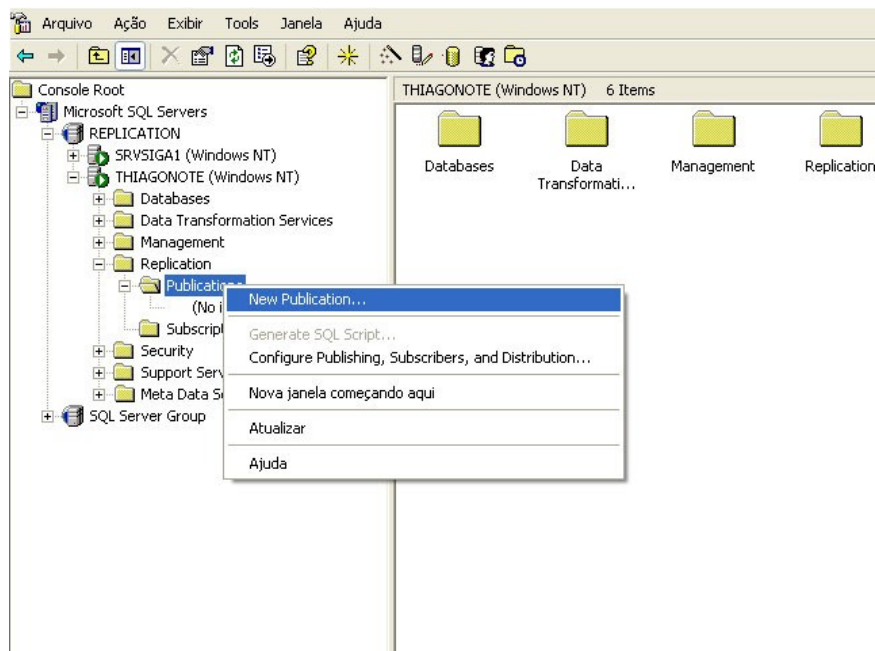


Figura 4 – Criando uma nova Publicação

Com isso, faremos a configuração utilizando um assistente do SQL Server, para criar uma nova Publicação.

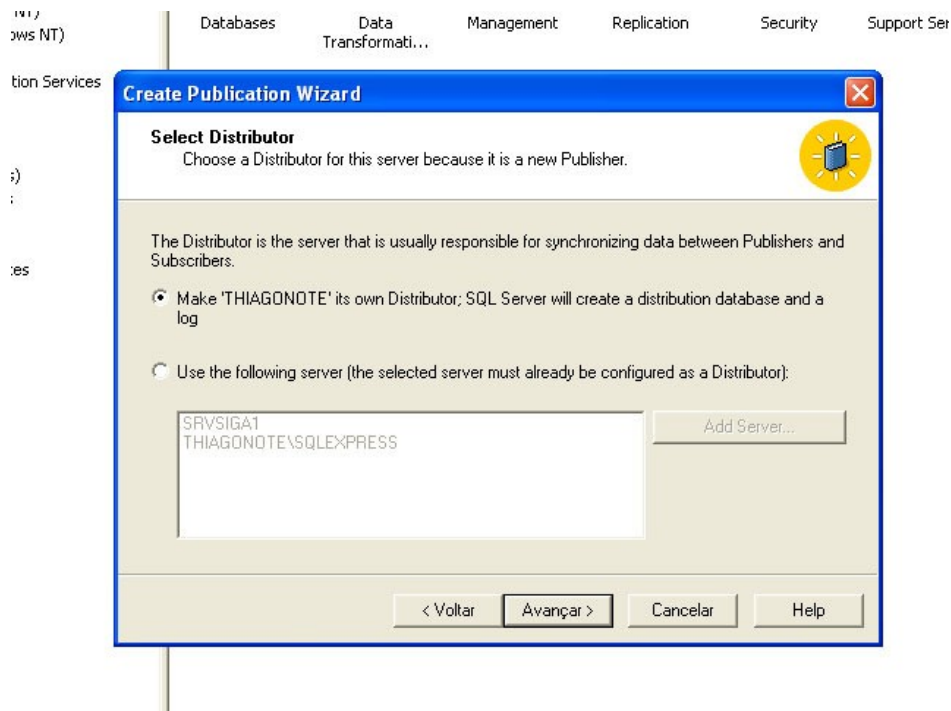


Figura 5 – Opções do Publicador/Distribuidor

Nesta janela, selecionamos a opção para que o servidor, neste caso, o THIAGONOTE seja não só Publicador, como também o Distribuidor dos dados, pois como vimos anteriormente, a existência de um servidor com a função de ser o distribuidor não é obrigatória.

Ao continuarmos, temos que selecionar o Banco de Dados que queremos publicar para o(s) outro(s) servidore(s).

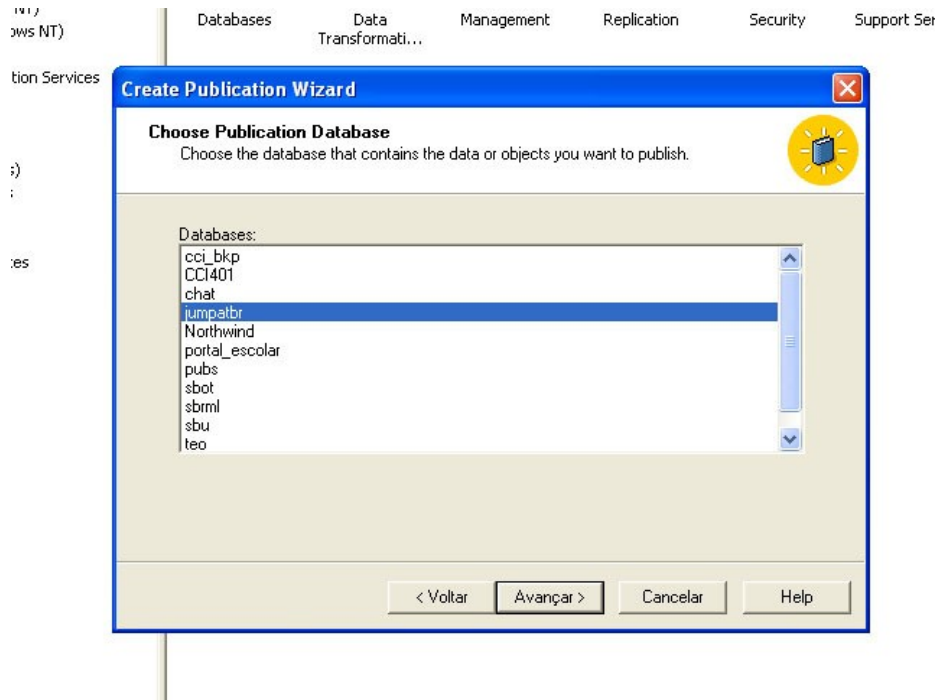


Figura 6 – Selecionando o Database

Após este passo, temos que escolher o tipo de replicação que desejamos realizar nesta publicação.

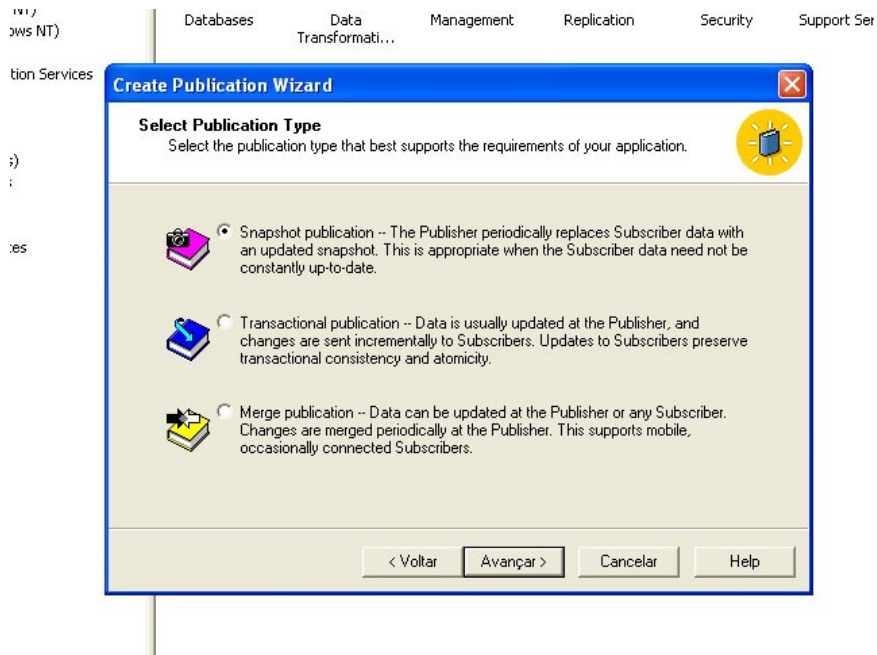


Figura 7 – Tipos de Publicação

Na tela seguinte o assistente nos mostra onde será armazenada a cópia do banco realizada para a replicação dos dados no Snapshot.

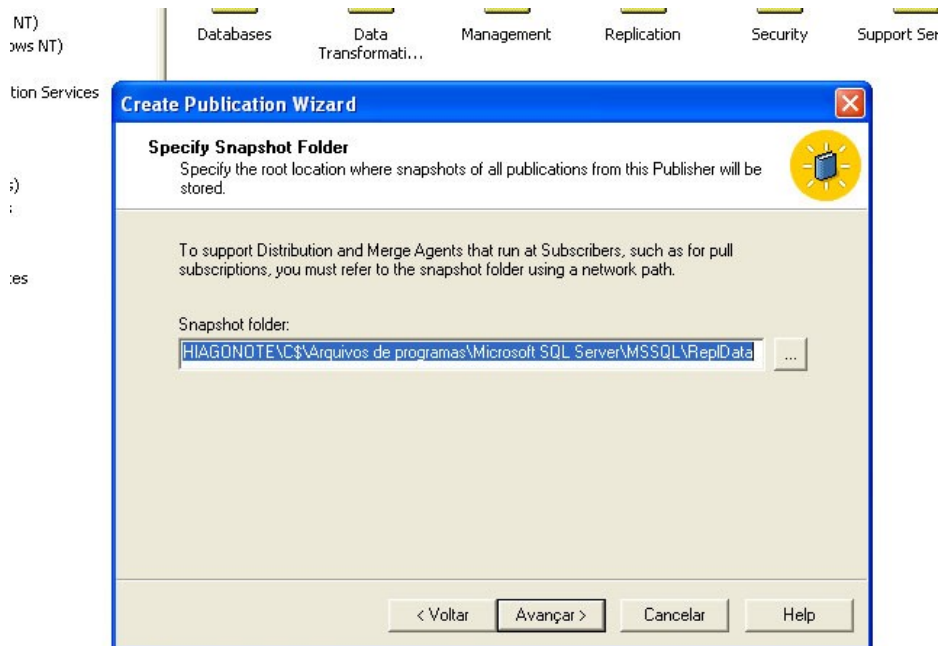


Figura 8 – Local da rede onde será salvo o Snapshot para a publicação

Agora devemos escolher quais objetos vão ser gravados no Snapshot para serem replicados.

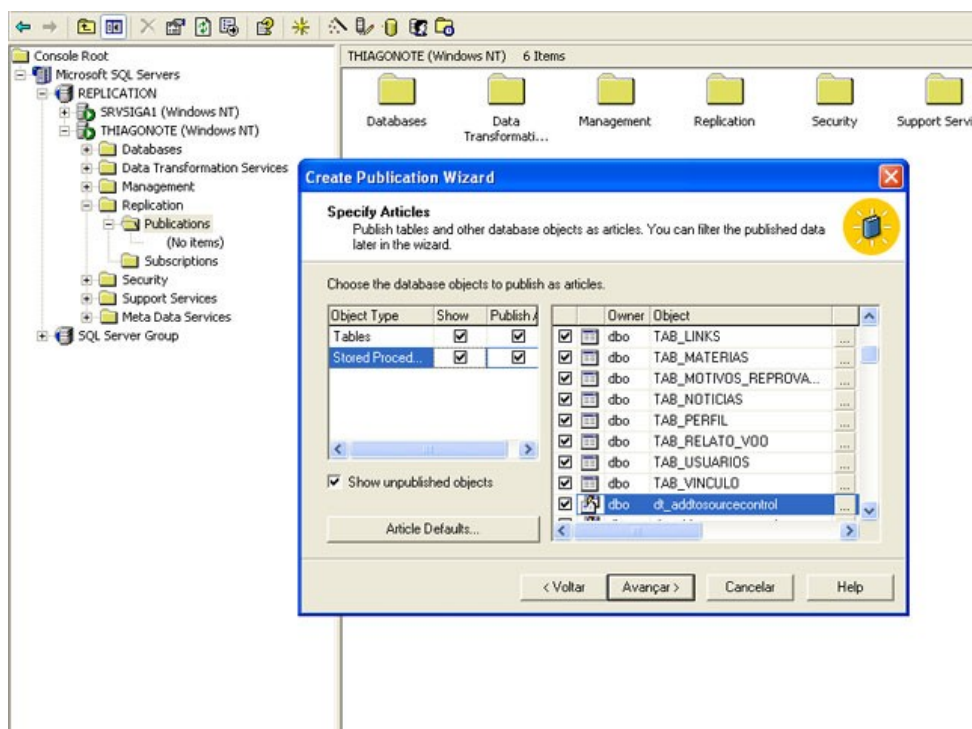


Figura 9 – Definindo os artigos (objetos do banco de dados) que serão publicados

O passo seguinte é nomear a publicação que criamos.

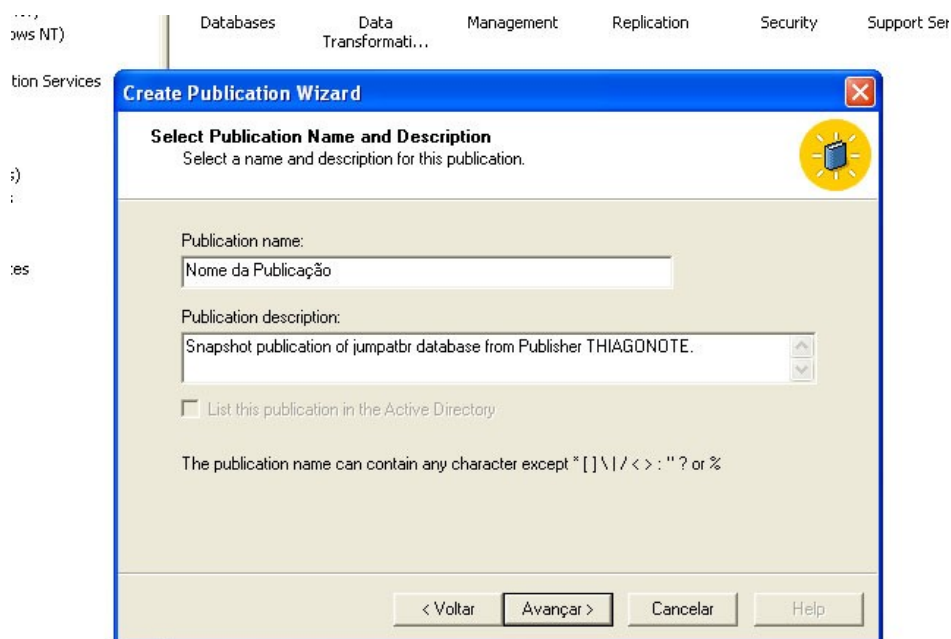


Figura 10 – Definindo um nome para a Publicação

Neste último passo, o assistente pergunta se desejamos definir algum filtro de dados ou personalizar algumas propriedades da replicação, neste caso, não entraremos nestes detalhes, pois se tratam de opções e formas de customização da replicação mais avançadas.

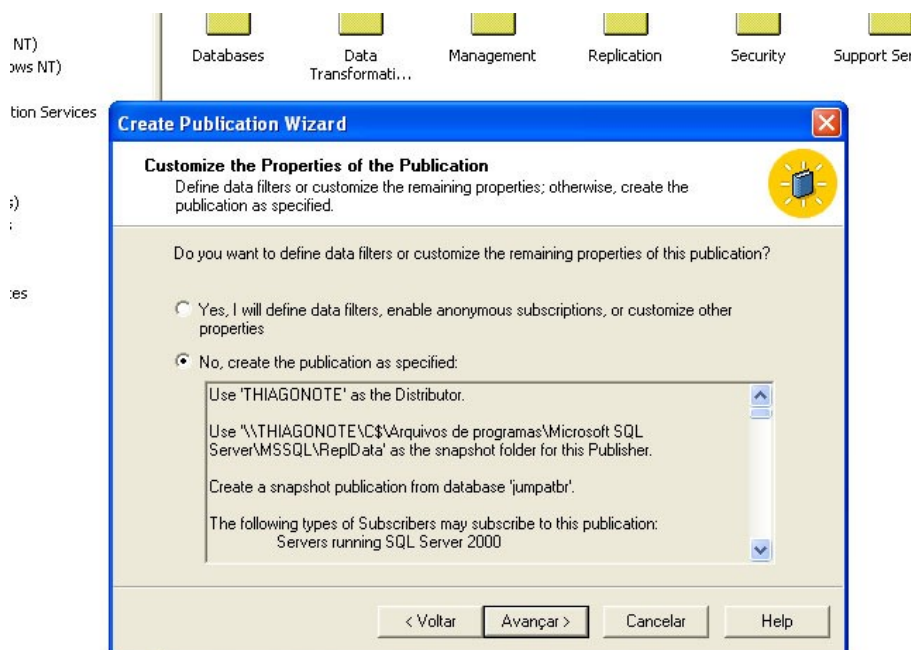


Figura 11 – Definindo filtros de dados e/ou customizando opções de publicação

E, por fim, finalizando a criação da Publicação através do Assistente do SQL Server, é feita a configuração do Distribuidor pelo próprio SQL Server.

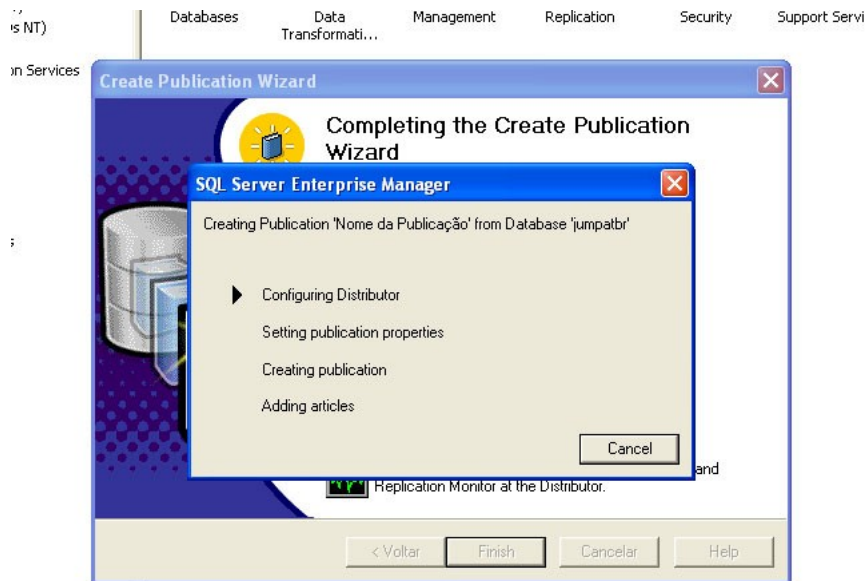


Figura 12 – Finalizando o assistente da Publicação

Com a publicação criada, faremos agora a criação da Assinatura, ou seja, da Inscrição dos servidores que receberão os dados do Publicador. Para isso, clicamos, sobre a Publicação que acabamos de criar, para configurar um Push Subscription seguindo o passo a passo do assistente, conforme as imagens a seguir:

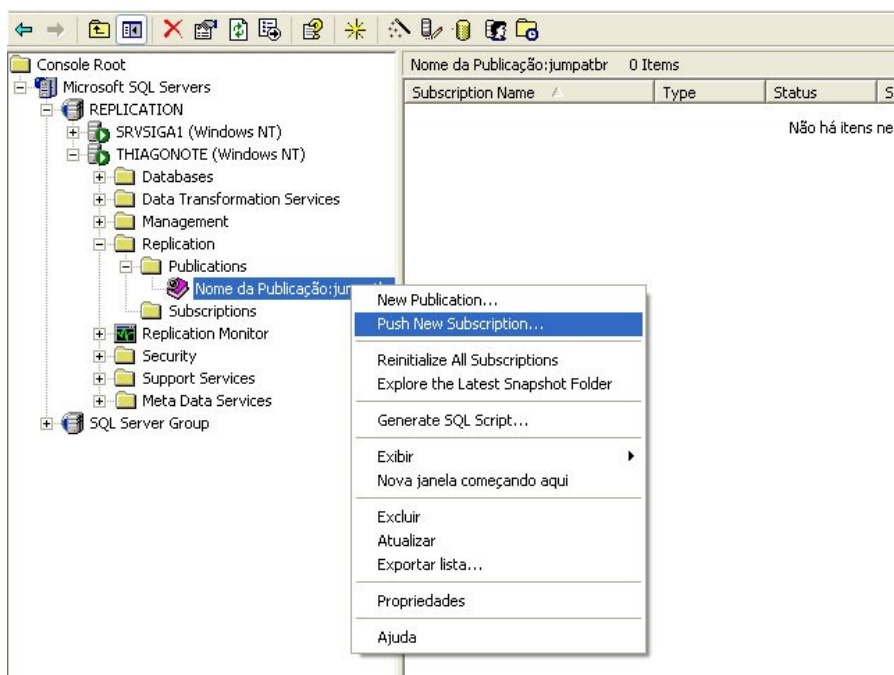


Figura 13 – Criando uma Subscription

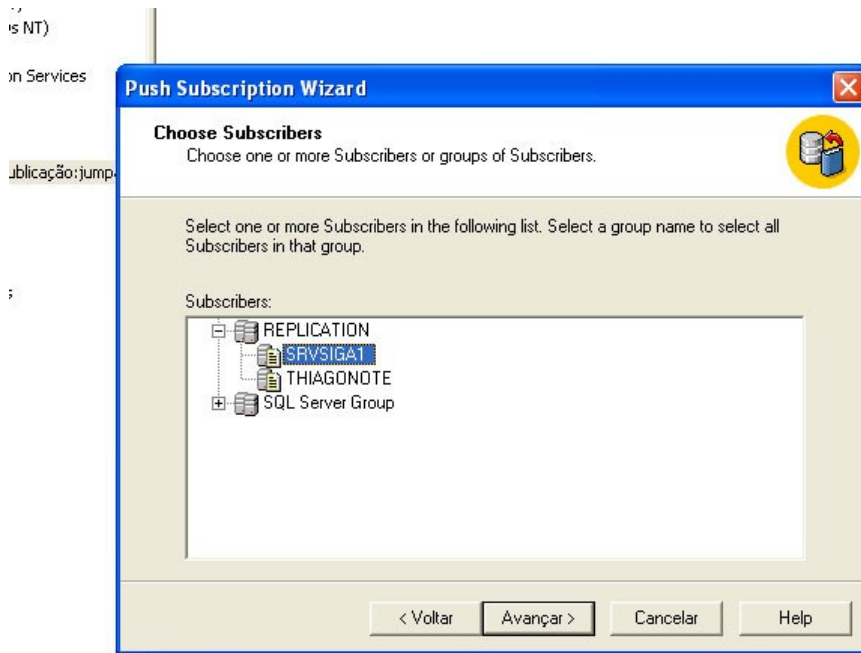


Figura 14 – Escolhendo o servidor que será o Subscriber

Com a escolha do servidor que receberá as publicações, partimos para a configuração do Banco de Dados de destino. Nesta tela do assistente, escolhemos o nome do Banco de Dados existente, ou conforme a Figura 14, criamos o Banco de Dados para a publicação.

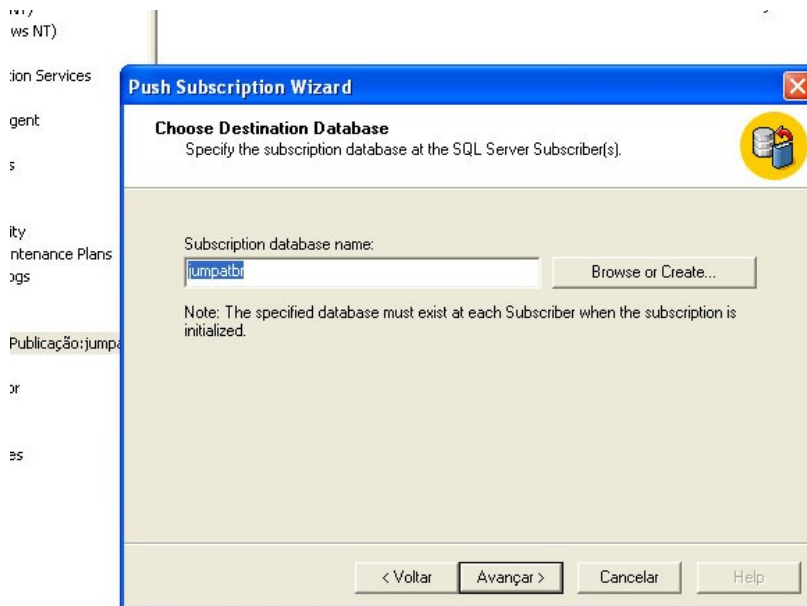


Figura 15 – Escolhendo o DataBase

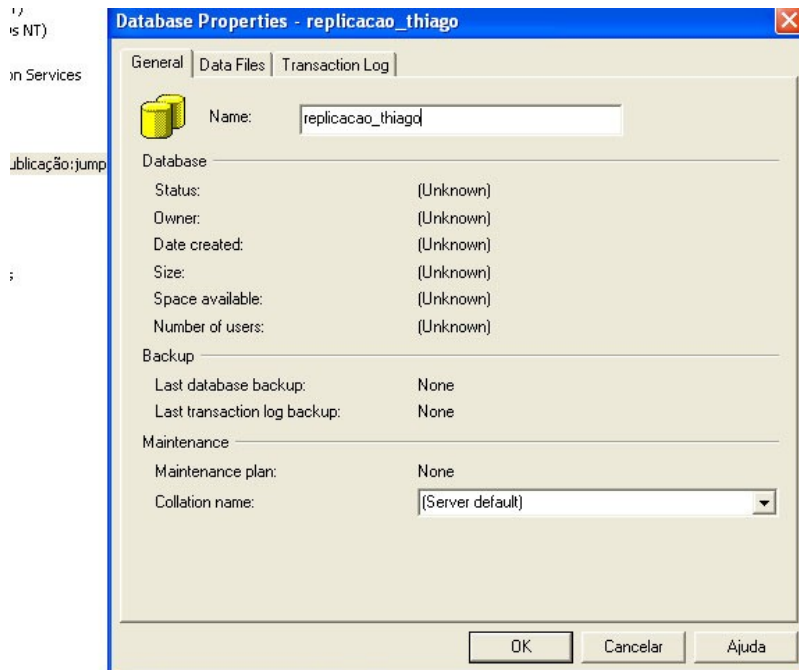


Figura 16 – Criando o Database no servidor que receberá os dados

Neste momento, temos que configurar quando rodará o snapshot para replicar os dados no(s) servidor(es) que receberão os dados.

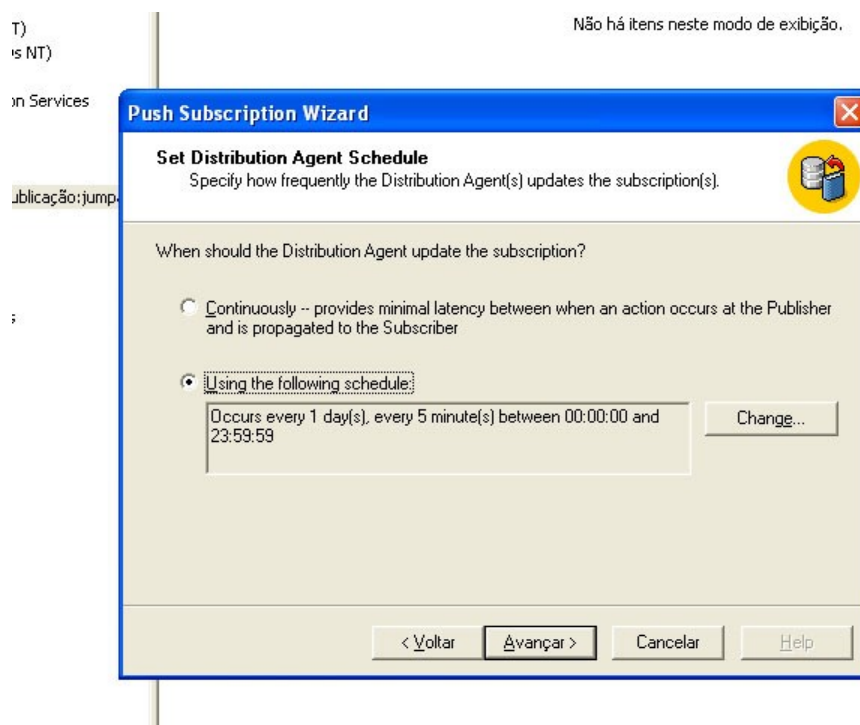


Figura 17 – Selecionando a frequência com que a replicação será realizada

Logo em seguida, verificamos o status do Agente do SQL Server que é o responsável pela criação do Snapshot e conseqüentemente a realização da replicação.



Figura 18 – Finalizando a criação da Subscription

Assim, finalizamos a instalação e o SQL Server configura da forma correta para a realização da publicação e obtemos a imagem de sucesso na criação.

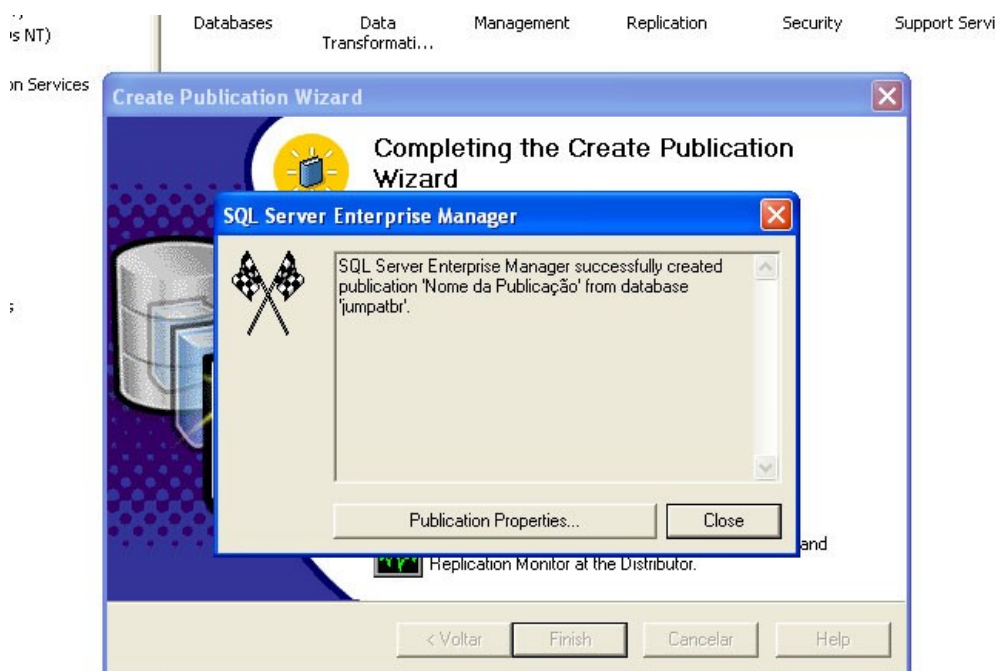


Figura 19 – Criação da Subscription efetuada

Agora, para simular a replicação dos dados iremos forçar a criação do Snapshot, inicializando os Jobs que são criados pelos assistentes. Pois, é através destes Jobs que o SQL Server executa a replicação dos dados na data e frequência definida.

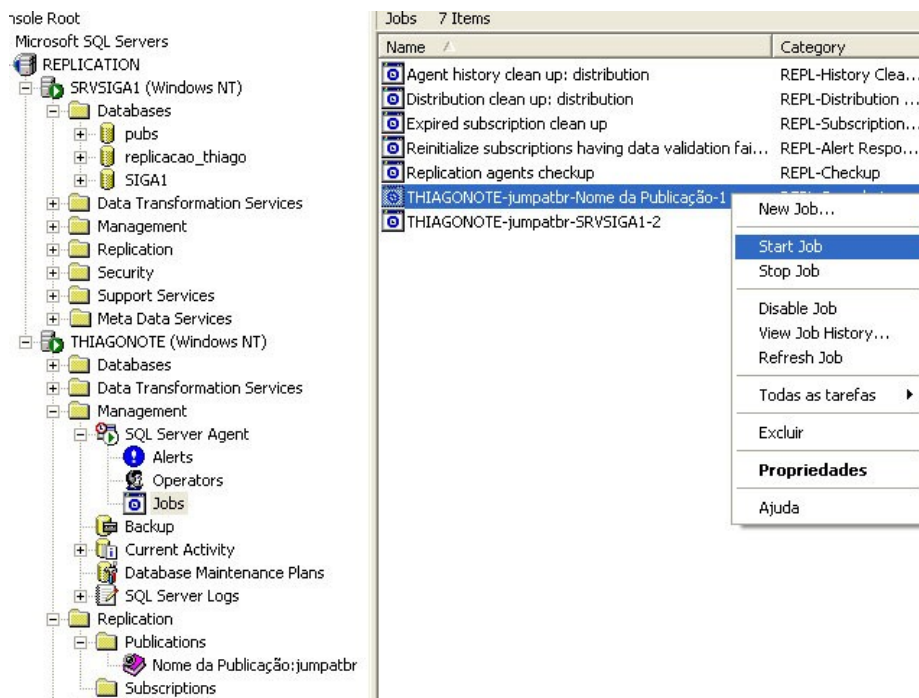


Figura 20 – Inicializando o Job que cria o Snapshot do Database

Com isso, podemos verificar no servidor SIGA1, que o Banco de Dados de destino, que chamamos de “replicacao_thiago”, está criado, porém não existe nenhuma tabela criada. Até porque, até agora só executamos o Job que cria a Snapshot do Banco de Dados.

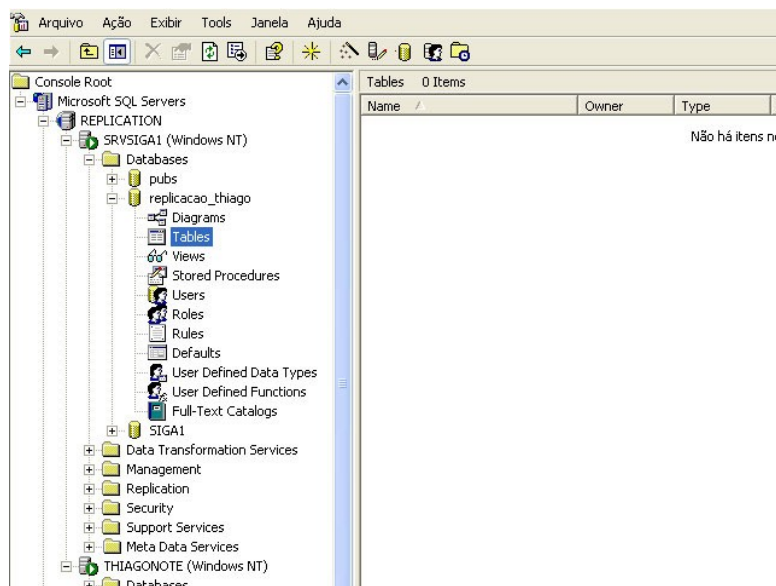


Figura 21 – Verificação das tabelas no servidor SIGA1 antes da replicação

Depois que o Job de replicação é realizado já podemos verificar que as tabelas foram criadas no servidor **SIGA1**, na tabela “replicacao_thiago”, conforme a figura 21.

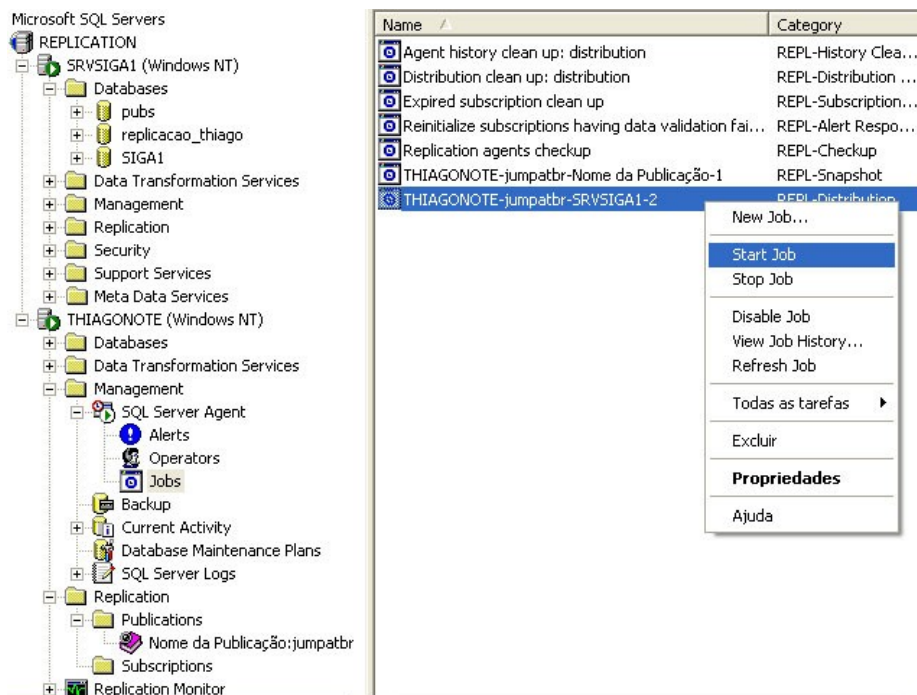


Figura 22 – Inicializando o JOB que replica os dados para o servidor de Subscription

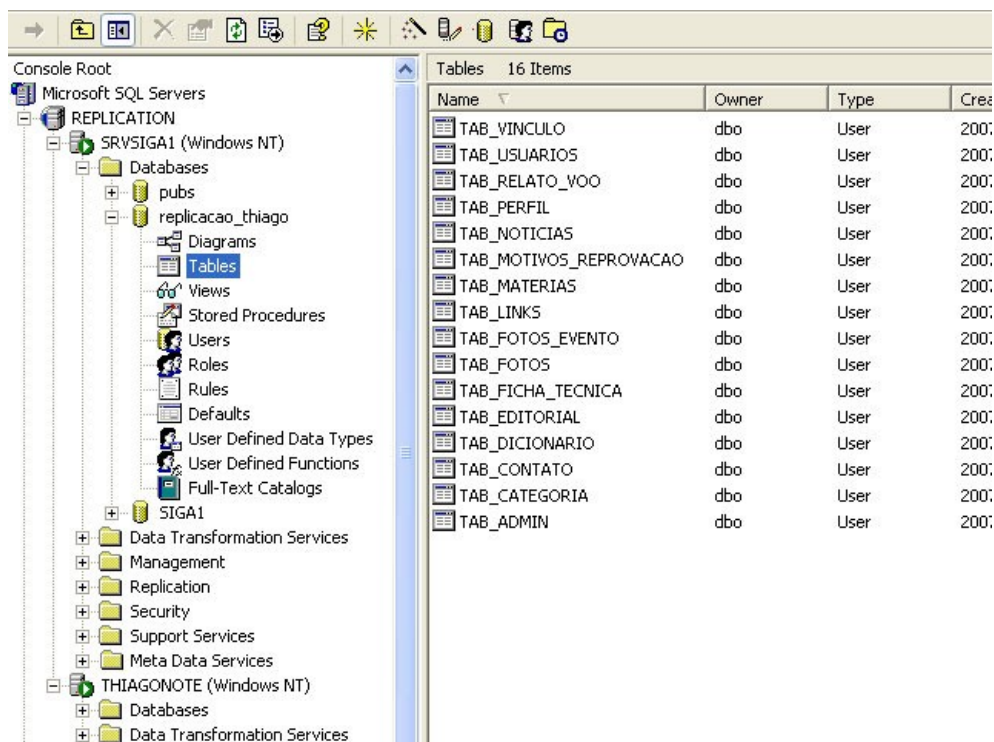


Figura 23 – Verificando as tabelas do servidor SIGA1, depois da execução do Job de replicação

Portanto, notamos que as tabelas contidas no servidor THIAGONOTE foram exatamente copiadas para o servidor SIGA1. Além das tabelas conforme visualizado na figura 21, todas as Stored Procedures foram criadas também.

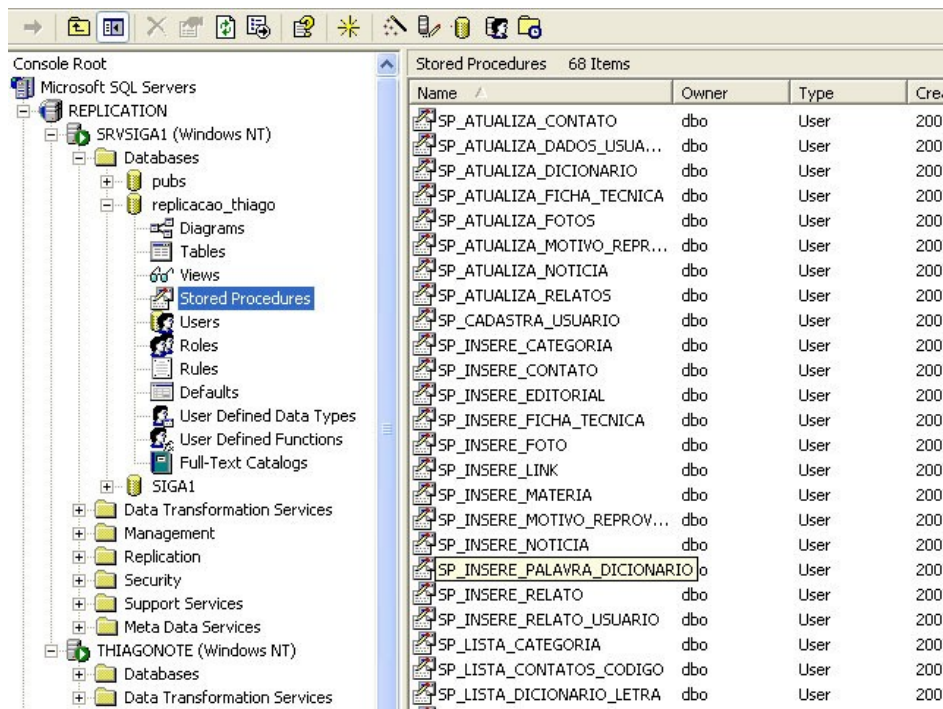


Figura 24 – Todas Stored Procedures criadas no servidor SIGA1 após a replicação

CONCLUSÃO

Concluimos então, que a replicação é o método preferido de distribuição de dados, pois ela demanda em um menor custo do que a utilização de transações distribuídas. O SQL Server 2000 ainda permite que você escolha um, dos três tipos de replicação existentes, dependendo do modelo físico utilizado.

Teoricamente, qualquer modelo de replicação pode ser utilizado por qualquer um dos tipos de replicação, muito embora, cada modelo tenha um tipo específico para cada caso de replicação.

Uma publicação é a menor unidade de replicação. Um banco de dados pode ter muitas publicações, com diferentes tipos de replicação, entretanto, cada publicação corresponde a apenas um banco de dados.

BIBLIOGRAFIA

[1] Battisti, Júlio. SQL Server 2000 – Administração & Desenvolvimento – Curso Completo. Axcel Books.

[2] Dusan Petkovic-SQL Server 2000- *Macron Books*

<http://support.microsoft.com/ph/2852>

www.microsoft.com/sql/default.mspx