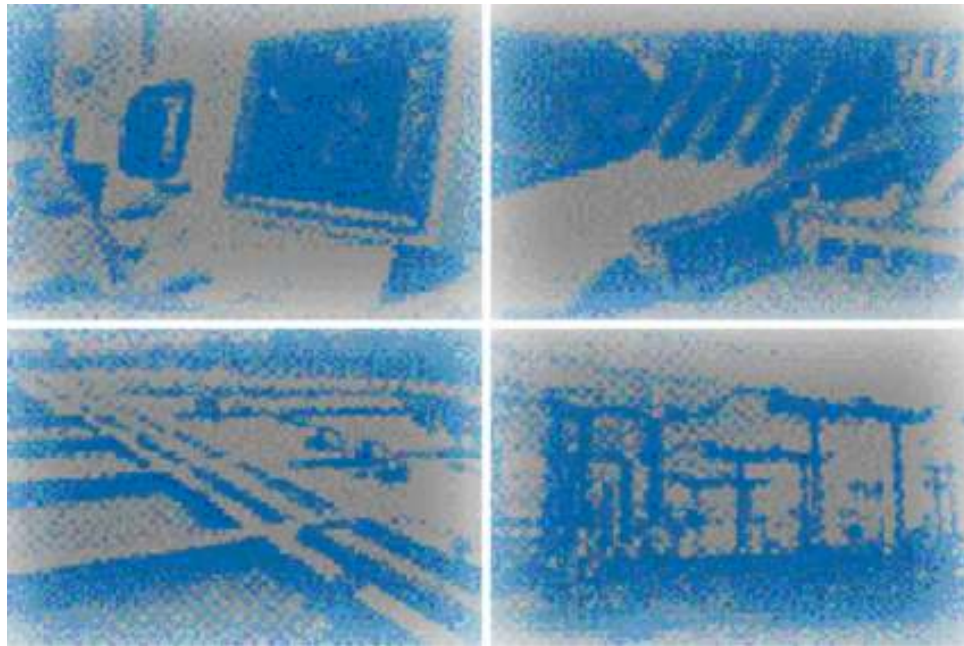


Action View

Software para a Supervisão e Controle de Processos e Equipamentos



Gerando Base de Dados MSDE ou SQL Server

SPIN - Engenharia de Automação Ltda.
Versão 7.1 – Dezembro / 2006

ÍNDICE

GERANDO BASE DE DADOS EM MSDE OU SQL SERVER	4
1 - INTRODUÇÃO	4
2 - CONFIGURAÇÃO	4
2.1 - Arquivo de Inicialização	4
2.2 - Banco de Dados.....	5
3 - GERENCIAMENTO DOS BANCOS DE DADOS NO ACTIONVIEW	5
4 - BANCO DE DADOS PARAMÉTRICO: IMPLEMENTAÇÃO, MIGRAÇÃO DE DADOS E REPLICAÇÃO	5
4.1 - Implementação	6
4.1.1 - Nome do Banco de Dados	7
4.1.2 - Arquivo dos dados	8
4.1.3 - Arquivo de Log de Transações	9
4.1.4 - Esquema da base de dados	10
4.2 Migração de dados.....	11
4.2.1 - Assistente de importação de dados	12
4.2.2 - Base de dados origem dos dados.....	13
4.2.3 - Base de dados destino dos dados	14
4.2.4 - Filtro de dados para importação	15
4.2.5 - Definição das tabelas a serem importadas	16
4.2.6 - Migração e replicação do banco de dados.....	17
4.2.7 - Resumo da configuração de importação e replicação dos dados ..	18
4.2.8 - Migrando os dados.....	19
4.3 - Publicação	19
4.3.1 - Tipo de publicação	20
4.3.2 - Tipos de bancos de dados permitidos para replicação	21
4.3.3 - Especificação de artigos para publicar.....	22
4.3.4 - Nome da Publicação	23
4.3.5 - Edições dos Artigos	24
4.3.6 - Personalizando as propriedades da publicação.....	24
4.3.7 - Filtro de dados a serem incluídos na publicação	25
4.3.8 - Filtro de colunas de tabelas	26
4.3.9 - Assinaturas anônimas.....	27
4.3.10 - Gerenciamento do Snapshot Agent	28
4.3.11 - Resumo do assistente de replicação	29
4.4 - Adicionando uma assinatura.....	31
4.4.1 - Assistente para assinatura.....	31
4.4.2 - Assinantes	33
4.4.3 - Nome da base de dados no assinante.....	34
4.4.4 - Frequência de atualização da assinatura.....	35
4.4.5 - Inicialização da assinatura	36
4.4.6 - Prioridade da assinatura para resolução de conflitos.....	37
4.4.7 - Serviços necessários para a assinatura.....	38
4.4.8 - Concluindo a assinatura.....	39
5 - BANCO DE DADOS HISTÓRICO	40
APÊNDICE.....	41

TRANSACTION LOGS	41
REPLICATION MODEL	42

Gerando Base de Dados em MSDE ou SQL Server

1 - Introdução

O ActionView suporta a configuração de seu banco de dados paramétrico (Action.mdb) e de histórico (Actmov.mdb) dos mestres em SQL Server 2000 ou MSDE 8.00.194 (Service Pack 2) ou posterior. A seguir descreveremos os passos para tal, incluindo a replicação automática entre as bases de dados nos mestres.

Vale à pena lembrar que o MSDE (Microsoft SQL Server Desktop Engine) possui distribuição gratuita, mas possui limitação de 2GB para o banco de dados e máximo de 5 conexões simultâneas. Acima de 5 conexões, a performance do banco é degradada propositalmente.

2 - Configuração

2.1 - Arquivo de Inicialização

Foram feitas alterações a partir da versão 6.1.2 no Action.ini para suporte à implantação do banco de dados em MSDE/SQL Server nos mestres. A partir desta versão, os clientes não terão mais as bases de dados localmente no próprio computador; elas estarão nas máquinas mestras e replicadas entre si. As modificações foram:

[Directories]

BaseAction0=c:\action\DBase\action_mestre.udl	(Base Paramétrica do Mestre Padrão da base de dados)
BaseAction1=c:\action\DBase\ action_escravo.udl	(Base Paramétrica do Escravo Padrão da base de dados)
BaseActmov0=c:\action\DBase\actmov_mestre.udl	(Base Histórica do Mestre Padrão da base de dados)
BaseActMov1=c:\action\DBase\actmov_escravo.udl	(Base Histórica do Escravo Padrão da base de dados)

Os itens BaseAction0 e BaseActmov0 devem apontar para arquivos de extensão UDL que representem as conexões ao banco de dados paramétrico e de histórico da estação que for mestre na base de dados. Para BaseAction1 e BaseActmov1 vale o mesmo, só que para a estação escrava no banco de dados. Se for estação única, Baseaction0 = BaseAction1 e BaseActmov0 = BaseActmov1, apontando para a base paramétrica e de histórico, respectivamente, da estação única.

Não existem mais os itens BaseActionLocal e BaseActmovLocal.

[Rede SPPCOMFG]

Mestre=<Nome do Mestre da Base de Dados>

Escravo=<Nome do Escravo da Base de Dados>

Em caso de estação única, Mestre=Escravo=Nome da estação única.

2.2 - Banco de Dados

Na tabela SysConfig foi criado o campo unTimeLastUpdate (inteiro longo).

3 - Gerenciamento dos bancos de dados no ActionView

Quando o ActionView é iniciado, o programa tem que se conectar a base de dados mais atualizada dos dois mestres. Mas como saber isto? Foi criado um mecanismo onde o programa gravador de histórico de medidas (histrec.exe), na estação que está rodando como mestre corrente, grava periodicamente no campo unTimeLastUpdate o número inteiro longo que representa o horário atual da máquina. Na máquina escrava, este campo não é gravado. Assim, no início, o programa tenta abrir as duas bases de dados paramétricas (BaseAction0 e BaseAction1) e comparar estes dois campos de cada uma para saber qual a mais atual e se conectar a ela. Antes de tentar se conectar à BaseAction0, ele tenta descobrir se a máquina mestre padrão está na rede, dando um ping através da rede no item Mestre da seção Rede SPPCOMFG. O mesmo vale para BaseAction1 e o item Escravo da seção Rede SPPCOMFG. Se uma das estações estiver ausente, automaticamente o programa se conecta à outra. Caso as duas estejam ausentes, o programa se encerra automaticamente.

Este mesmo mecanismo descrito para a base de dados paramétrica também é usado para a base de dados histórica.

Depois desta conexão inicial, cada vez que é solicitada uma conexão a um banco de dados paramétrico ou de movimento, o programa checa quem é o mestre atual e se ele está conectado na rede e estabelece a conexão.

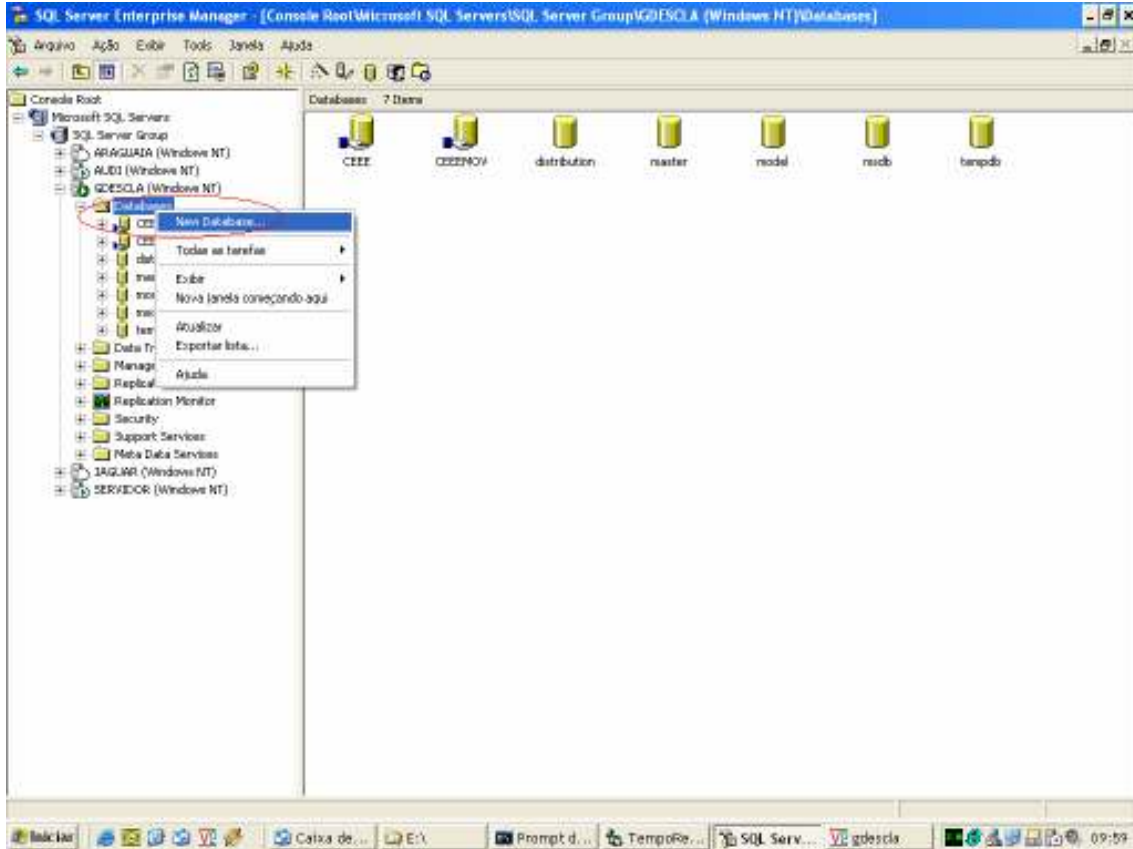
4 - Banco de dados paramétrico: Implementação, migração de dados e replicação

Deve-se ter o MSDE/ SQL Server instalado nas duas máquinas mestras e um banco de dados publicado em uma delas e um banco de dados assinatura na outra.

O seguinte exemplo usa as ferramentas Enterprise Manager e Query Analyzer do SQL Server 2000, mas podem ser usadas outras ferramentas disponíveis para criação do banco de dados, importação de dados e montagem do esquema de replicação.

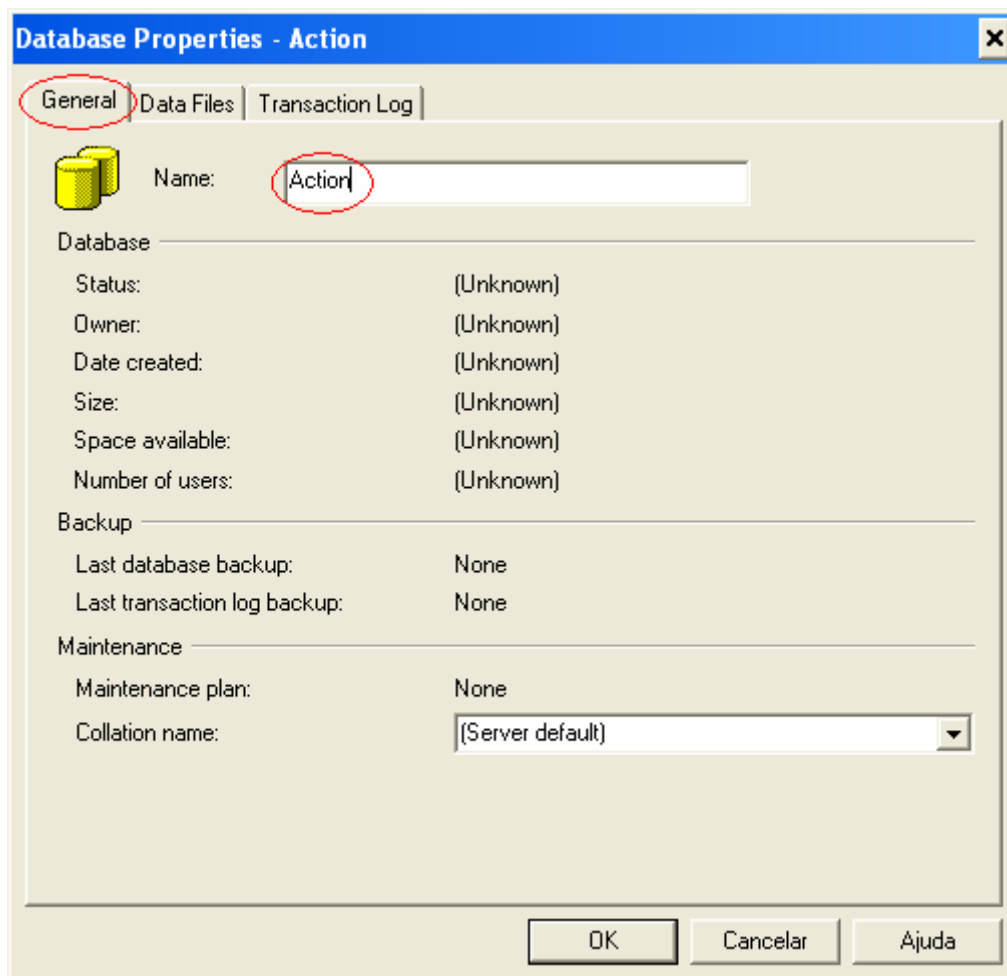
4.1 - Implementação

Dispare o programa Enterprise Manager e clique em “New Database” com o botão direito encima da guia “Databases”.



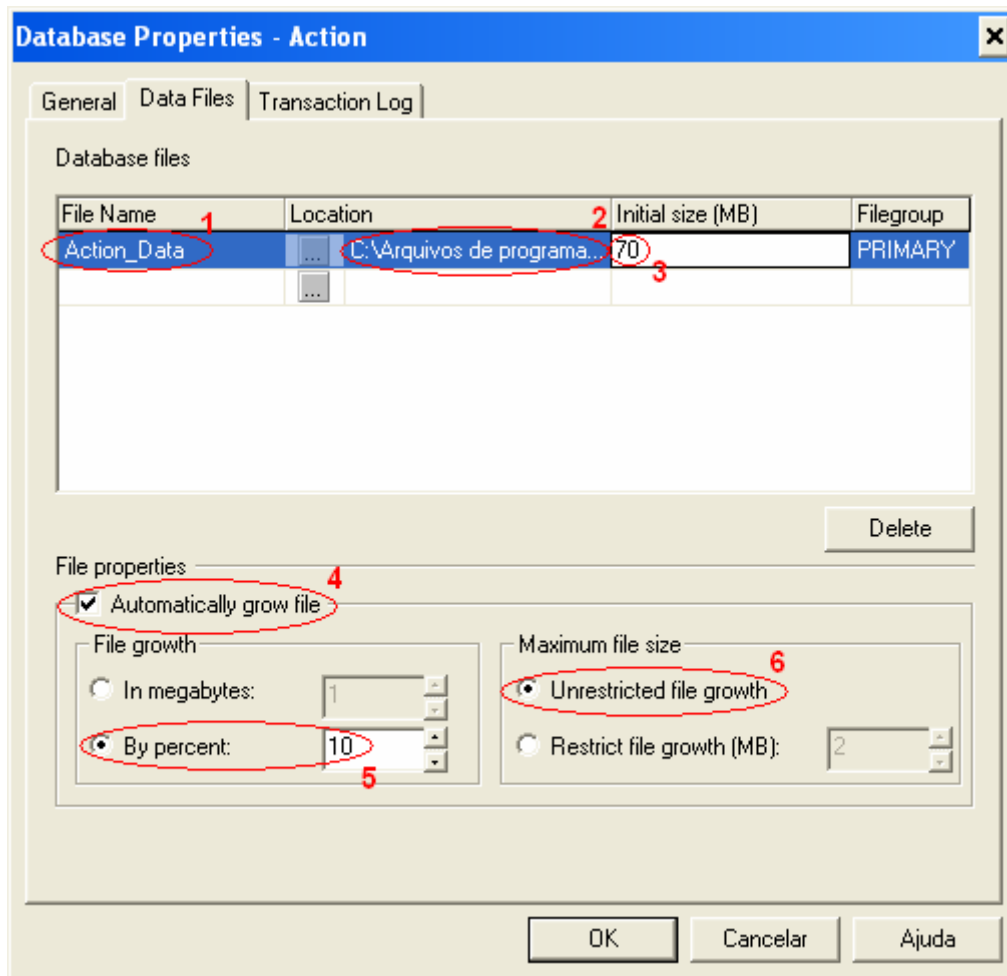
4.1.1 - Nome do Banco de Dados

Na tela que irá se abrir, digite o nome do Banco de Dados. No caso do nosso exemplo, o nome será Action.



Clique na pasta Data Files.

4.1.2 - Arquivo dos dados



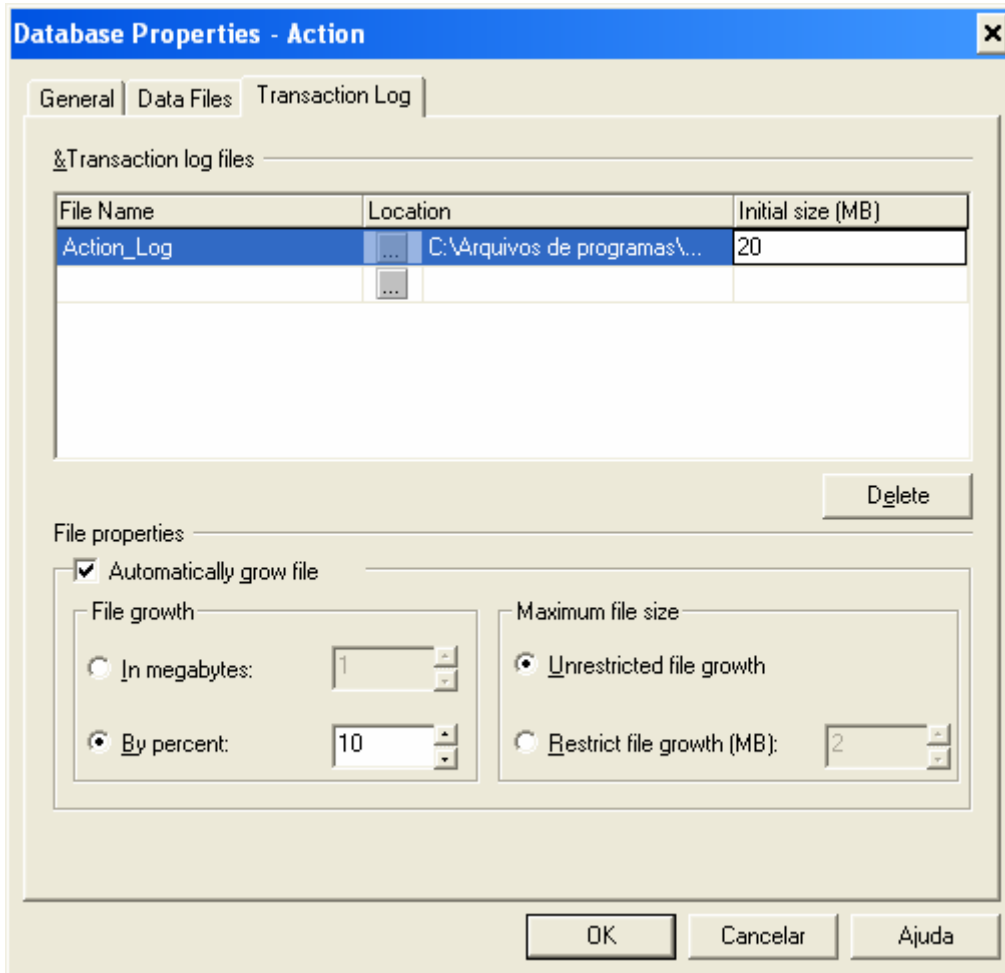
A definição do tamanho do banco de dados e de seu arquivo de log de transações dependerá especificamente do tamanho da sua instalação.

1. Nome do arquivo que será gravado contendo os dados da base paramétrica.
2. Pasta onde será gravado o arquivo de dados definido acima.
3. Tamanho inicial da base de dados.
4. Esta opção define que o arquivo dos dados crescerá sempre que necessário.
5. A sub-opção File Growth define como o arquivo crescerá quando for necessário. Pode ser de forma absoluta, em X Megabytes ou de forma percentual em X %.
6. A sub-opção Maximum file size define o tamanho máximo do arquivo de dados. Poderá ser enquanto houver espaço em disco, ou limitado a um tamanho específico.

Todo banco de dados SQL Server possui pelo menos um arquivo de dados e um arquivo de log. O log de transações permite ao SQL Server recuperar transações. Este arquivo é um registro serial de todas as

modificações efetuadas no banco de dados. Ele cresce continuamente conforme a base de dados vai sendo modificada. Para maiores detalhes consulte a documentação no Books Online do SQL Server ou na Internet em msdn.microsoft.com ou no apêndice no final deste manual na seção [Transaction Logs](#). Clique na pasta Transaction Log.

4.1.3 - Arquivo de Log de Transações



A criação do arquivo de log é similar ao que foi descrito para a criação do arquivo de dados.

Pressione o botão OK para proceder à criação do Banco de Dados. Esta etapa geralmente costuma ser um pouco demorada.

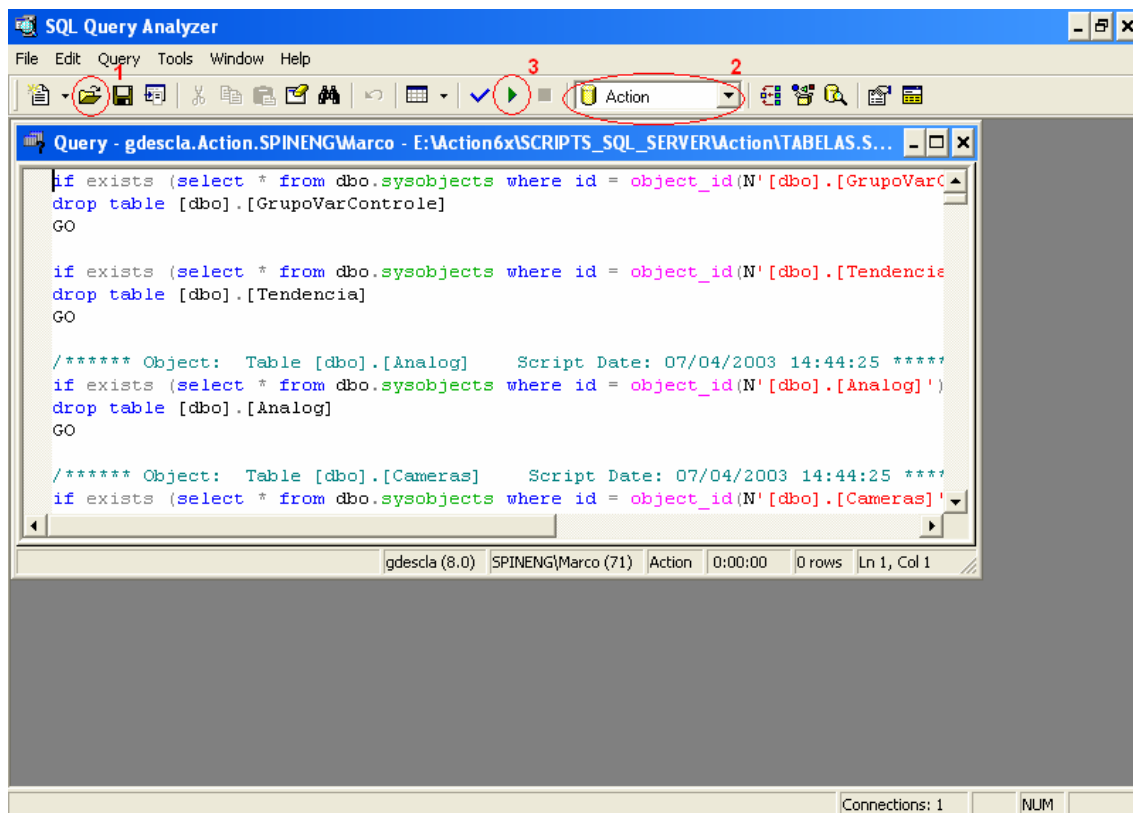
4.1.4 - Esquema da base de dados

Criado o arquivo de dados e o arquivo de log, faz-se necessário a “formatação lógica” deste arquivo. Deve-se definir toda a estrutura do banco de dados com suas tabelas, índices, constraints, relacionamentos, etc.

A base de dados paramétrica possui quatro scripts que fazem tudo isto. São eles:

Tabelas.SQL
Índices.SQL
Constraints.SQL
Relacionamentos.SQL

Para execução destes scripts no banco de dados, será usado o aplicativo Query Analyzer.



Os scripts devem ser executados na seguinte ordem:

1. Tabelas.SQL
2. Índices.SQL
3. Constraints.SQL
4. Neste ponto deve ser feita a importação de dados do Access
5. Relacionamentos.SQL

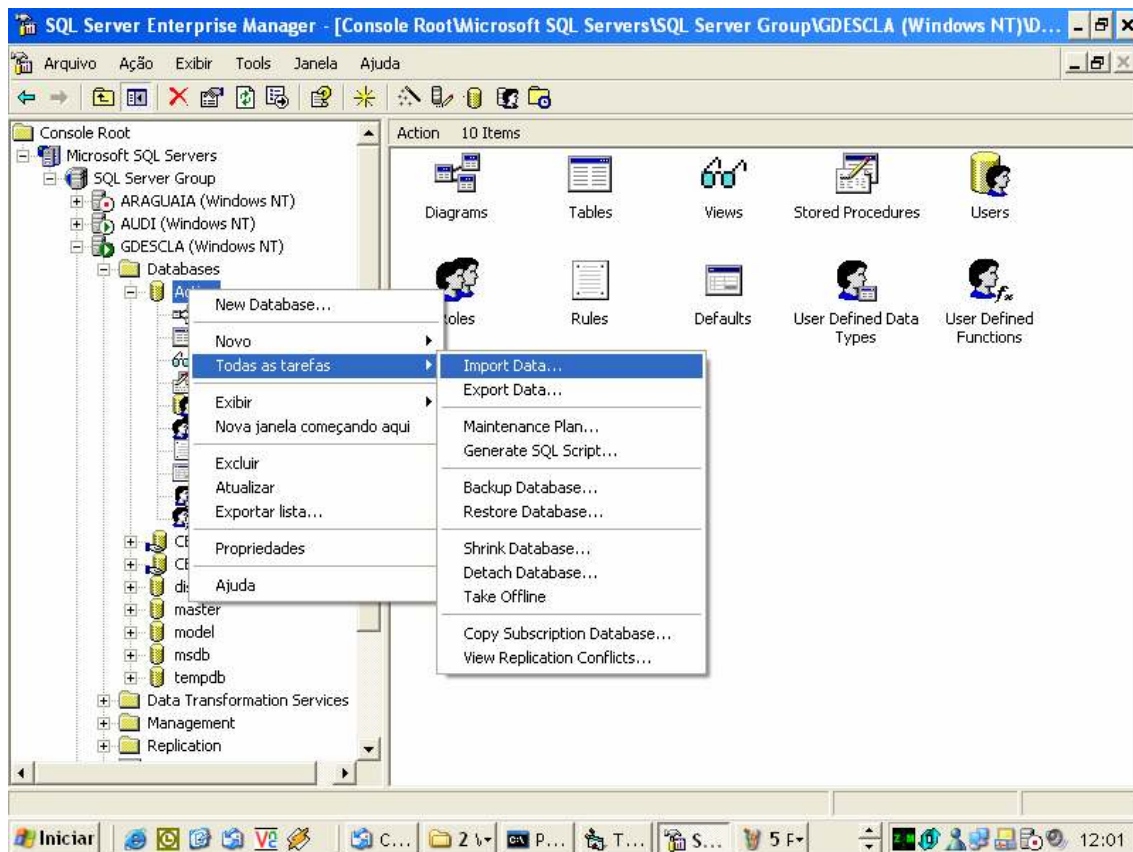
Para executar o script, siga os procedimentos abaixo:

1. Abra o script
2. Verifique se a base de dados selecionada é a base paramétrica
3. Execute o script

Agora deverá ser feita a importação de dados. Somente depois da importação será executado o script de relacionamentos.

4.2 Migração de dados

Para fazer a importação dos dados do banco de dados paramétrico Action.mdb para o banco de dados Action do MSDE/SQL Server será usado o programa Enterprise Manager. Clique com o botão direito sobre a base de dados paramétrica, selecione a opção Todas as Tarefas e depois a opção Import Data.

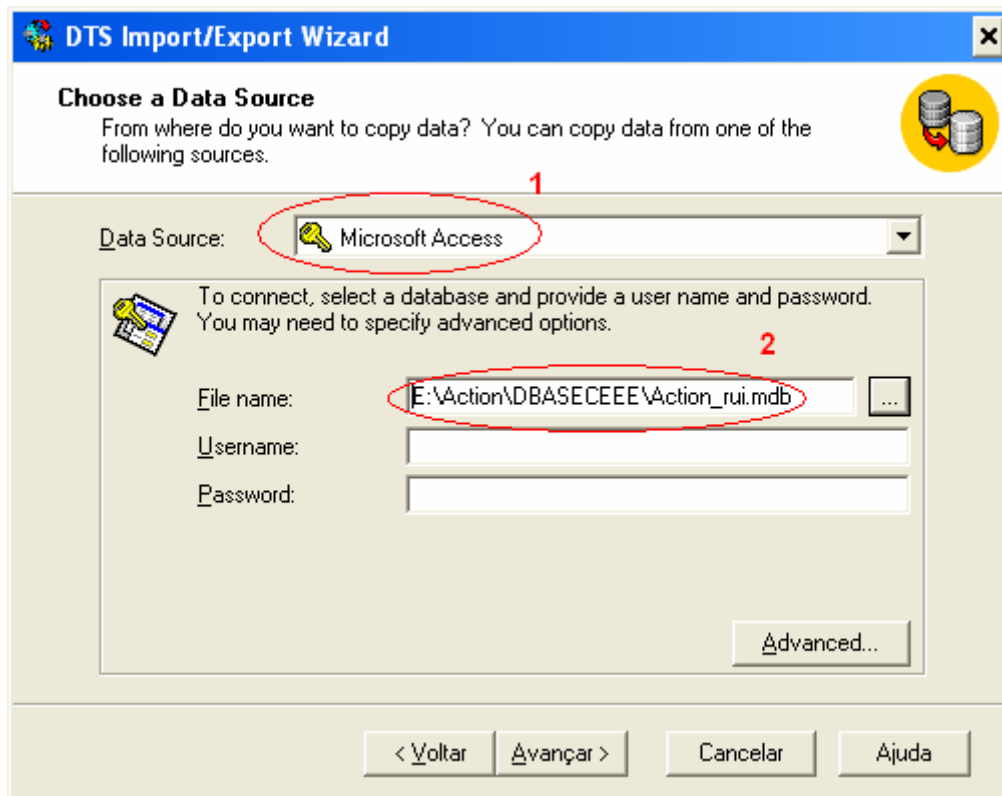


4.2.1 - Assistente de importação de dados



Na tela inicial do assistente de importação/exportação de dados, é informado que este assistente é capaz de importar e exportar dados heterogêneos com transformação entre os mais populares formatos de dados. Clique em avançar.

4.2.2 - Base de dados origem dos dados

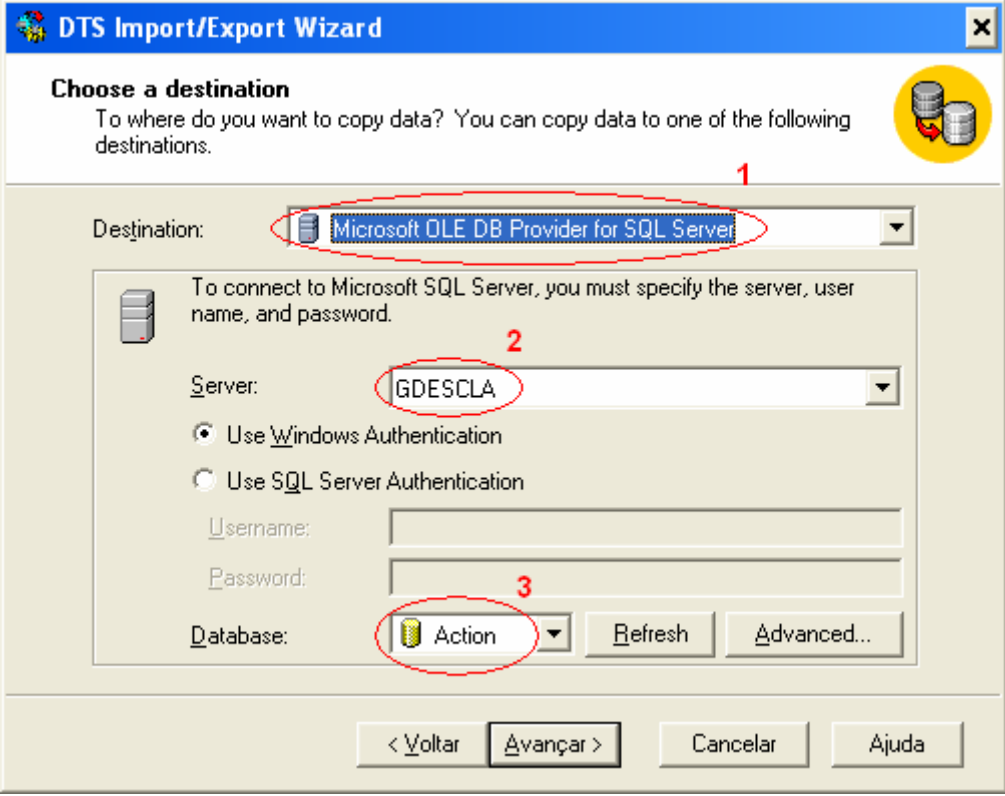


Esta tela é usada para definir a origem dos dados.

1. No campo Data Source, defina a fonte dos dados como sendo do tipo Access.
2. No campo FileName, selecione o caminho completo para a base de dados Action.mdb.

Clique em avançar.

4.2.3 - Base de dados destino dos dados

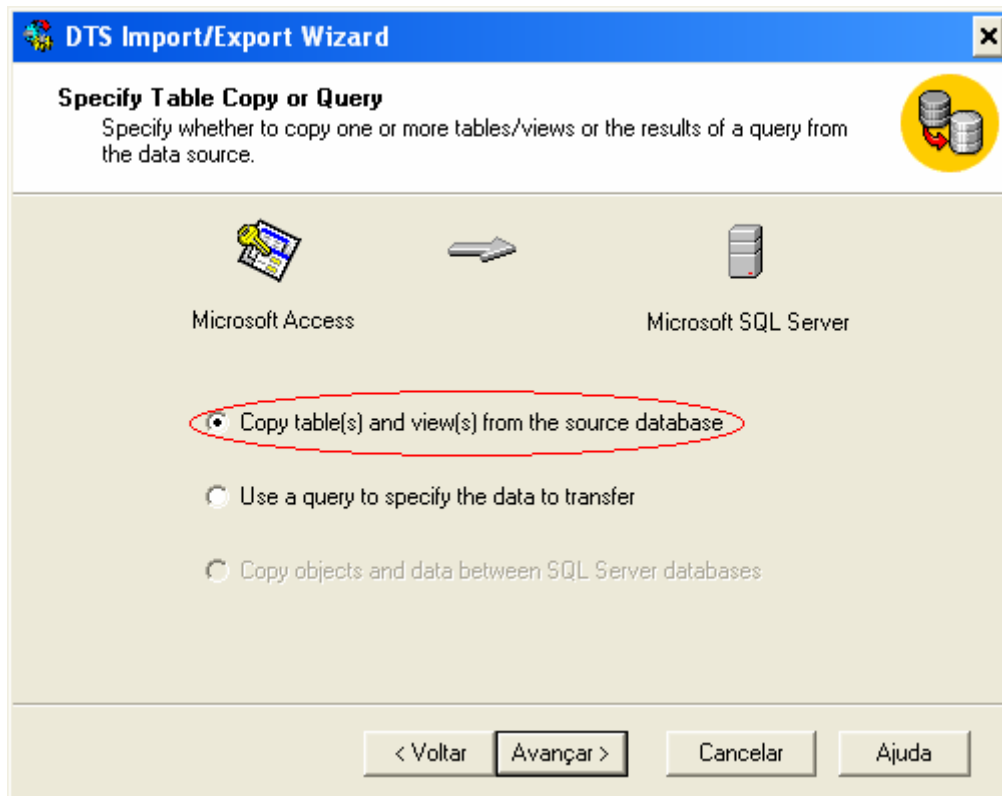


Esta tela é utilizada para definir o destino dos dados.

1. No campo destination, defina Microsoft OLE DB Provider for SQL Server.
2. No campo Server defina o computador onde está o SQL Server/MSDE contendo a base de dados Action.
3. No campo Database defina o nome da base paramétrica (Action).

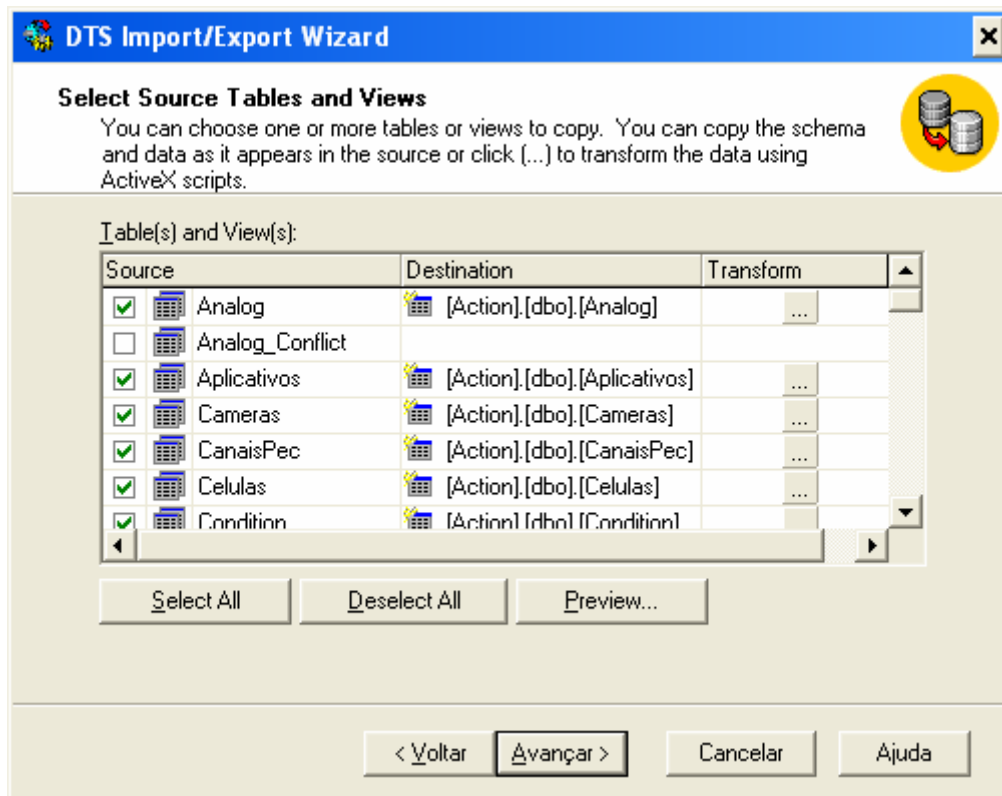
Clique em avançar.

4.2.4 - Filtro de dados para importação



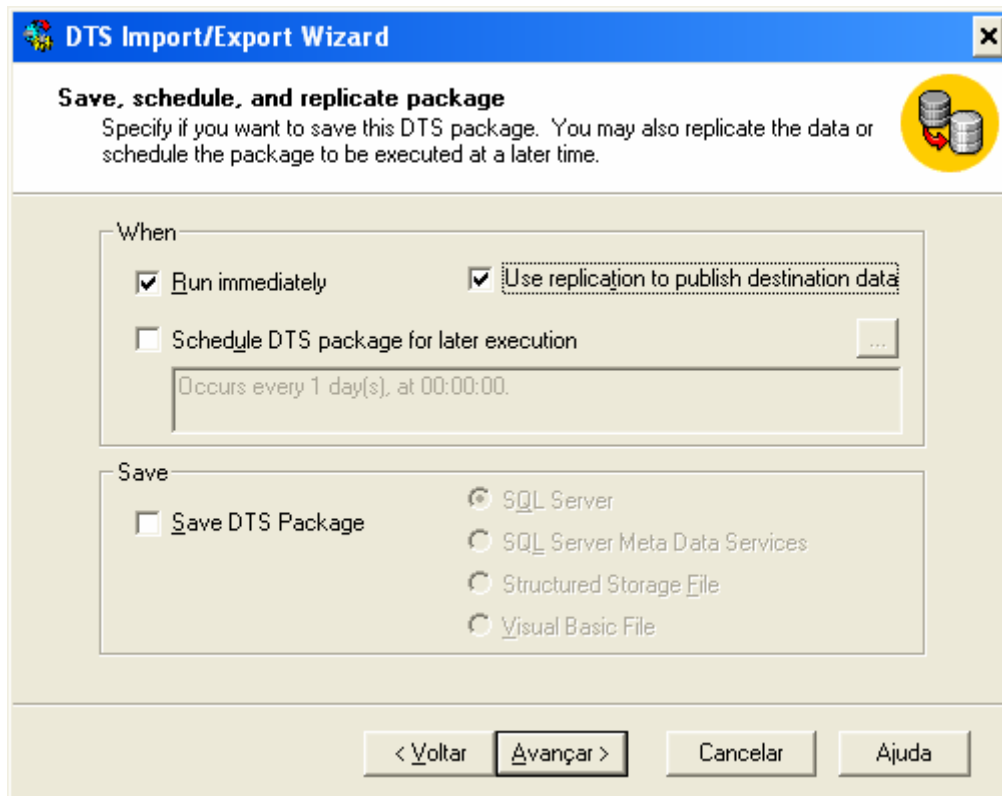
Esta tela permite copiar todas as tabelas e vistas para o banco de dados de destino ou permite filtrar os dados que serão importados através de consultas. No nosso caso específico, iremos importar todas as tabelas. Clique na opção “Copy table(s) and view(s) from the source database”. Clique em avançar.

4.2.5 - Definição das tabelas a serem importadas



Esta tela permite definir, para cada tabela da base de dados de origem a ser importada, transformações específicas em cada uma de suas colunas para a tabela da base de dados de destino através do botão “Transform”. Clique no botão “Select All” e em seguida no botão Avançar.

4.2.6 - Migração e replicação do banco de dados



Nesta tela, selecione a opção “Run Imediately” e “Use replication to publish destination data”. A segunda opção fará com que o banco de dados seja publicado para replicação. Clique em avançar.

4.2.7 - Resumo da configuração de importação e replicação dos dados



Todos os parâmetros para importação foram informados. Esta tela apresenta um resumo do que será feito.

Source: Microsoft Access
Using Microsoft Jet 4.0 OLE DB Provider
Location: e:\action\databaseeee\action_rui.mdb

[Definição da fonte dos dados Access e sua localização.](#)

Destination: Microsoft SQL Server
Using Microsoft OLE DB Provider for SQL Server
Location: GDESCLA
Database: Action

[Definição do destino dos dados no SQL Server.](#)

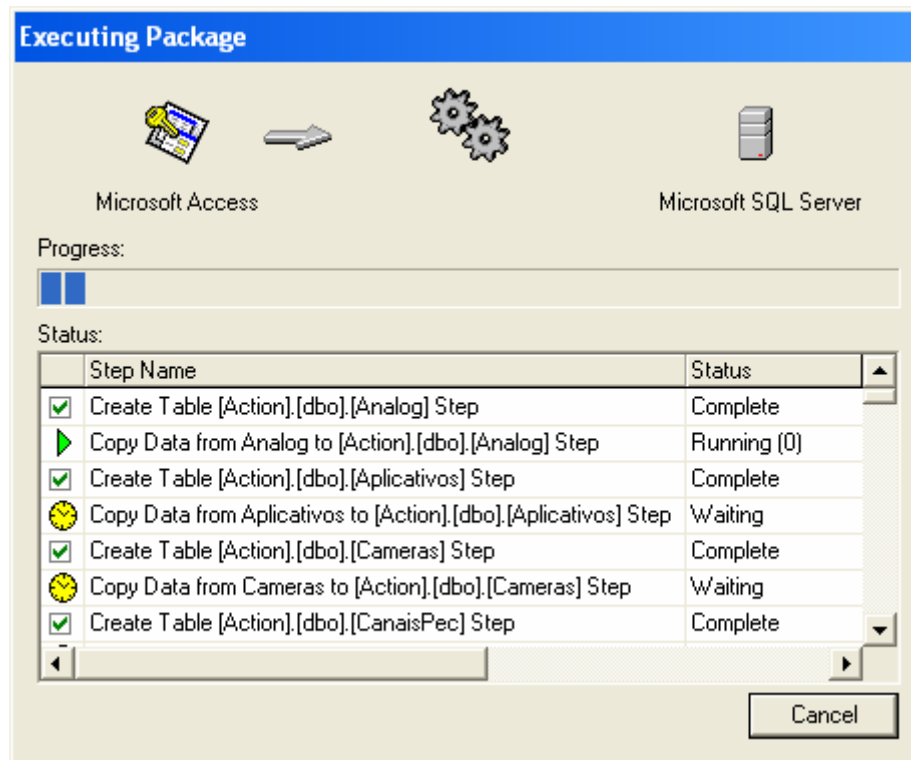
Tables
Analog -> [Action].[dbo].[Analog]
Aplicativos -> [Action].[dbo].[Aplicativos]

...

[Nomes das tabelas na origem e no destino.](#)

Clique em concluir para iniciar a migração dos dados.

4.2.8 - Migrando os dados



Migrados os dados, agora se pode executar o script de relacionamentos no aplicativo Query Analyzer, como descrito anteriormente.

Nesta etapa poderão ocorrer diversos erros, se na base de dados de origem, existirem incoerências nos dados. Caso isto venha a ocorrer, e você tenha um contrato de manutenção e suporte com a Spin, entre em contato com nosso analista da área de EPRO - Engenharia do produto.

Passada a etapa de migração dos dados, vem à etapa de publicação do banco de dados para replicação.

4.3 - Publicação

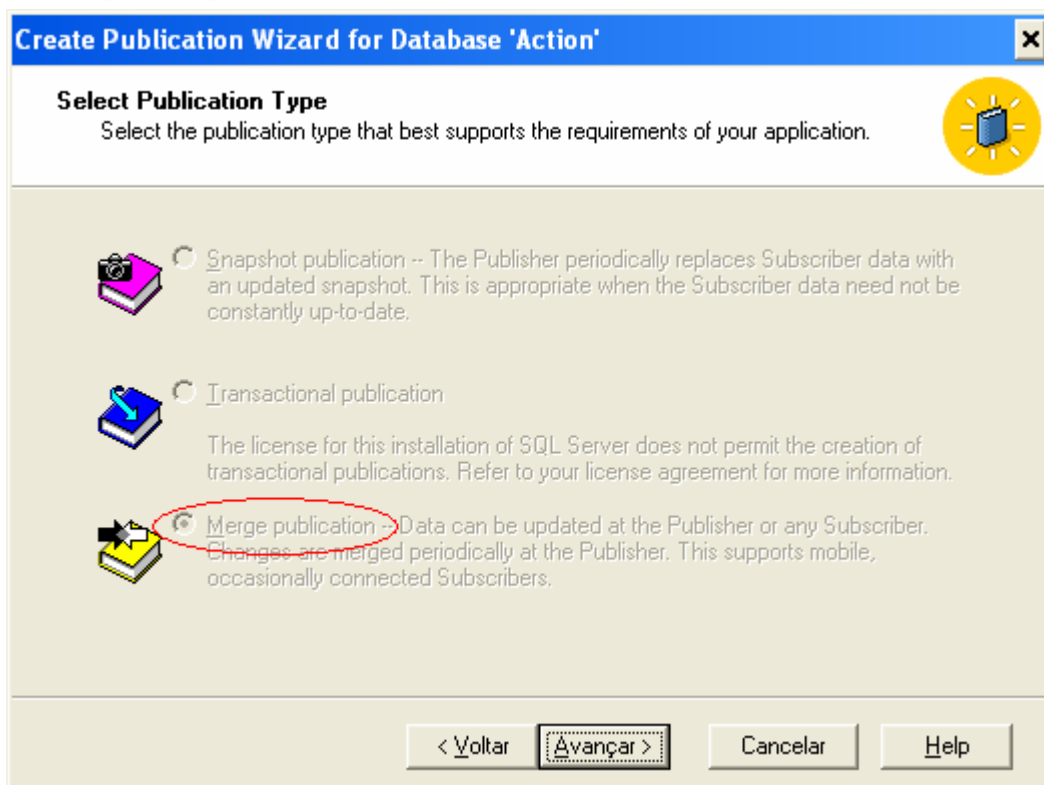
A replicação no MSDE/SQL Server consiste, de uma maneira simplificada, em publicar uma base de dados e criar assinantes desta publicação. Este assistente permite publicar os dados, filtrar os dados que serão publicados horizontalmente (linhas) e/ou verticalmente (colunas) entre outras coisas.

Para maiores detalhes sobre a replicação no SQL Server, leia no apêndice o item [“Replication Model”](#).



Clique em avançar.

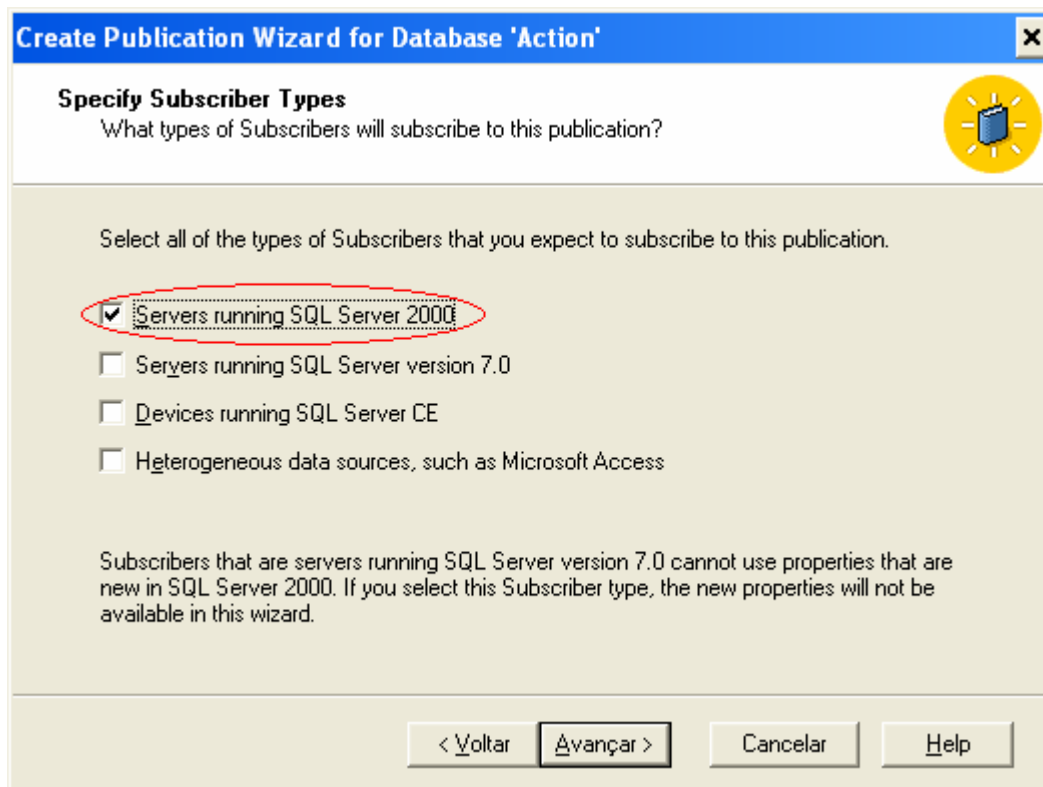
4.3.1 - Tipo de publicação



Existem três tipos possíveis de publicação para o SQL Server. Será abordada a publicação mista, onde os dados podem ser modificados tanto no publicador como nos assinantes. É uma replicação bi-direcional. Neste tipo de publicação, as mudanças serão periodicamente sincronizadas no publicador.

Selecione a opção Merge publication.
Clique em avançar.

4.3.2 - Tipos de bancos de dados permitidos para replicação

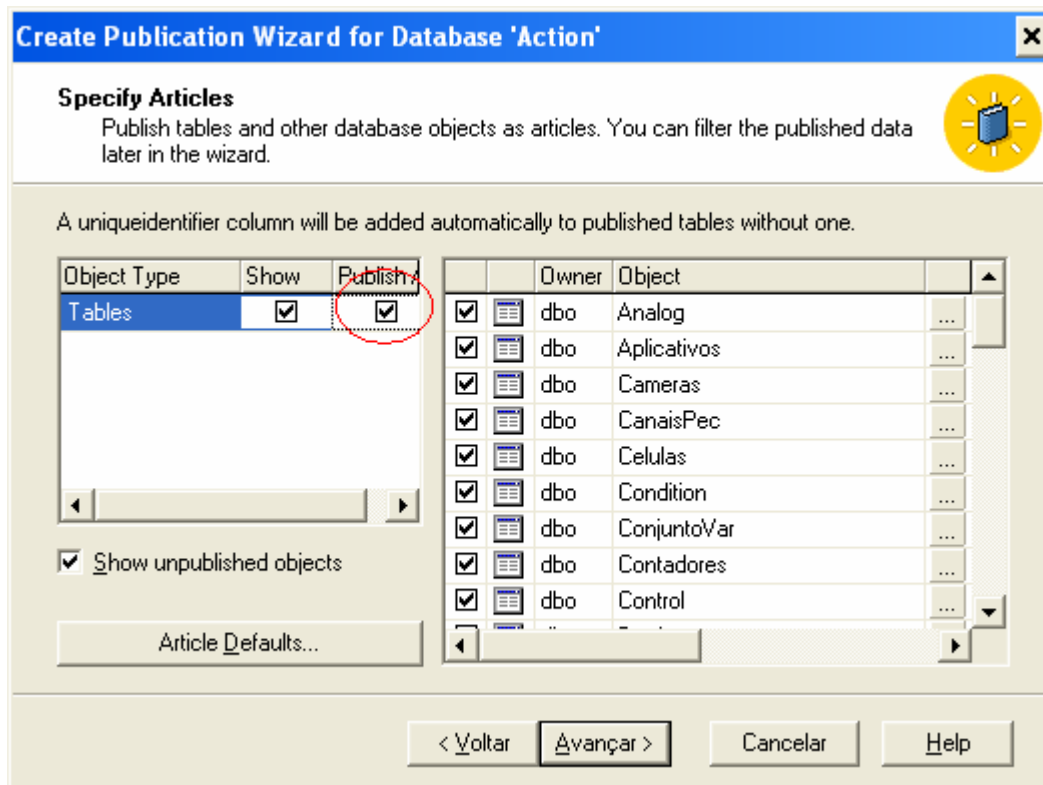


Pode-se definir que tipos de banco de dados poderão se replicar com esta publicação.

Selecione a opção “Servers running SQL Server 2000” especificando que somente poderão ser assinantes desta publicação, bases de dados SQL Server/MSDE.

Clique em avançar.

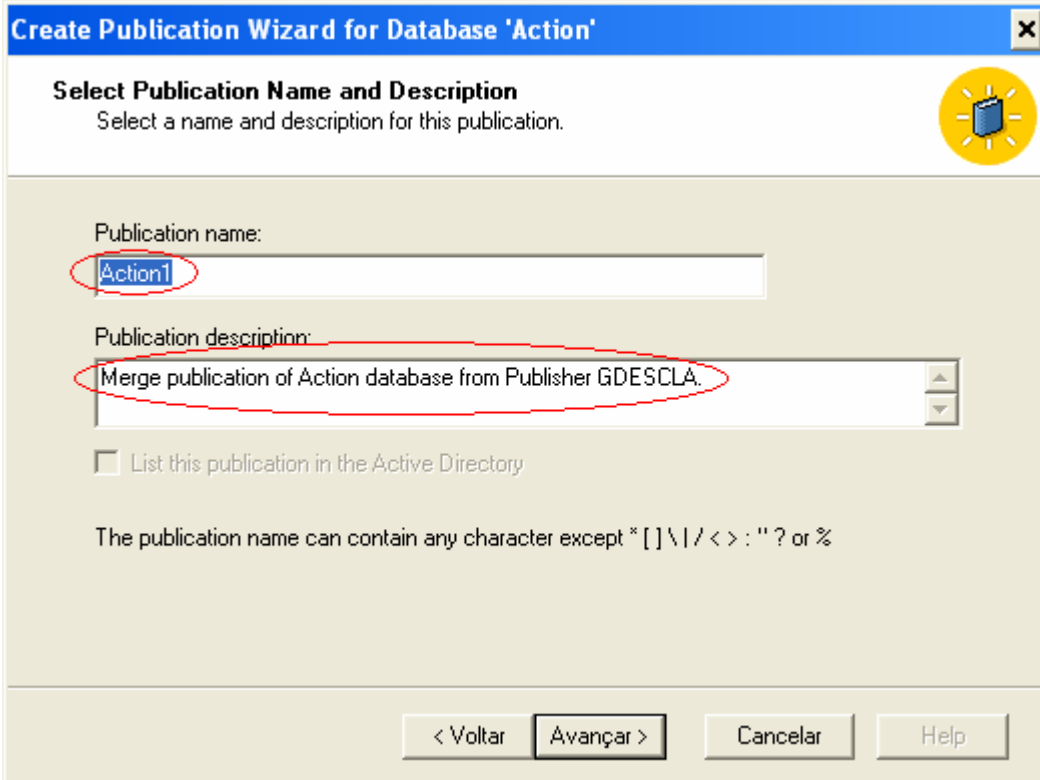
4.3.3 - Especificação de artigos para publicar



Define quais tabelas devem ser publicadas como “artigos” para sincronização com as assinaturas.

Selecione a opção “Publish All” para publicar todas as tabelas.
Clique em avançar.

4.3.4 - Nome da Publicação



Create Publication Wizard for Database 'Action'

Select Publication Name and Description
Select a name and description for this publication.

Publication name:
Action1

Publication description:
Merge publication of Action database from Publisher GDESCLA.

List this publication in the Active Directory

The publication name can contain any character except *[]\|/ < > : " ? or %

< Voltar Avançar > Cancelar Help

Especifique um nome e uma descrição para a publicação deste banco de dados.

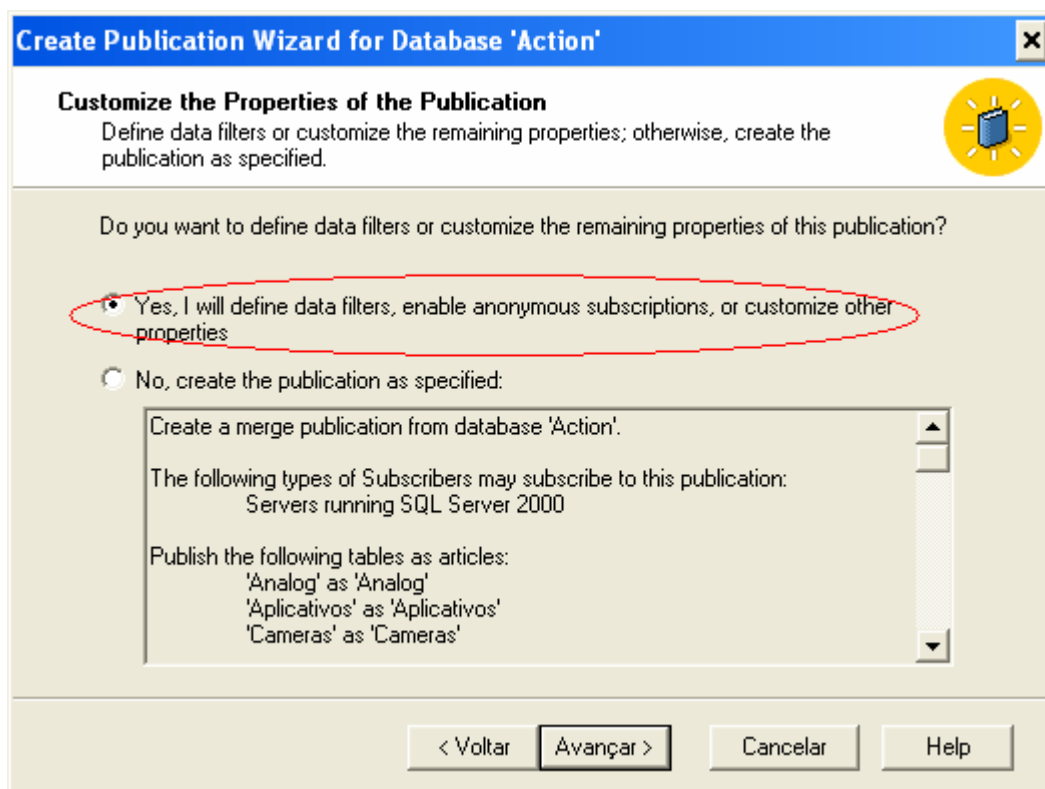
Clique em avançar.

4.3.5 - Edições dos Artigos

Poderá aparecer uma tela chamada “Article Issues”. Nesta tela o SQL Server relata a necessidade de criação de alguns campos em algumas tabelas para que a replicação funcione corretamente.

Clique em avançar.

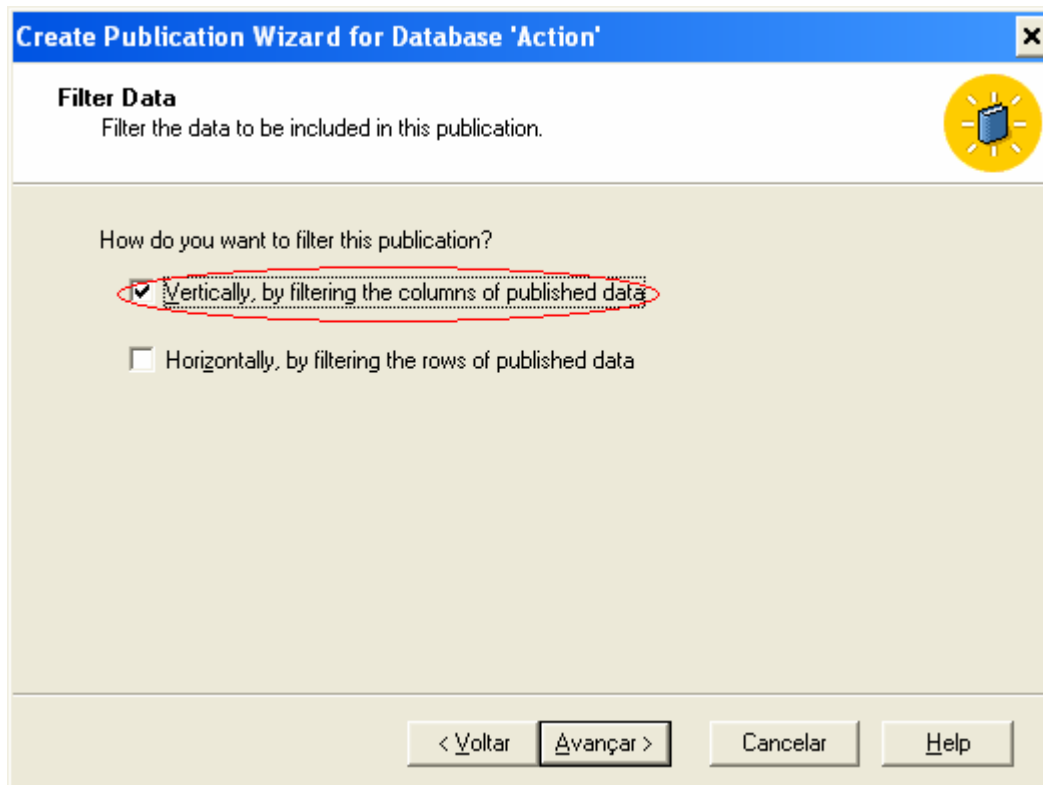
4.3.6 - Personalizando as propriedades da publicação



Selecione a opção “Yes, I will define data filters, enable anonymous subscriptions, or customize other properties”. Será necessário fazer algumas customizações para o funcionamento do gerenciamento de banco de dados do Action View.

Clique em avançar.

4.3.7 - Filtro de dados a serem incluídos na publicação

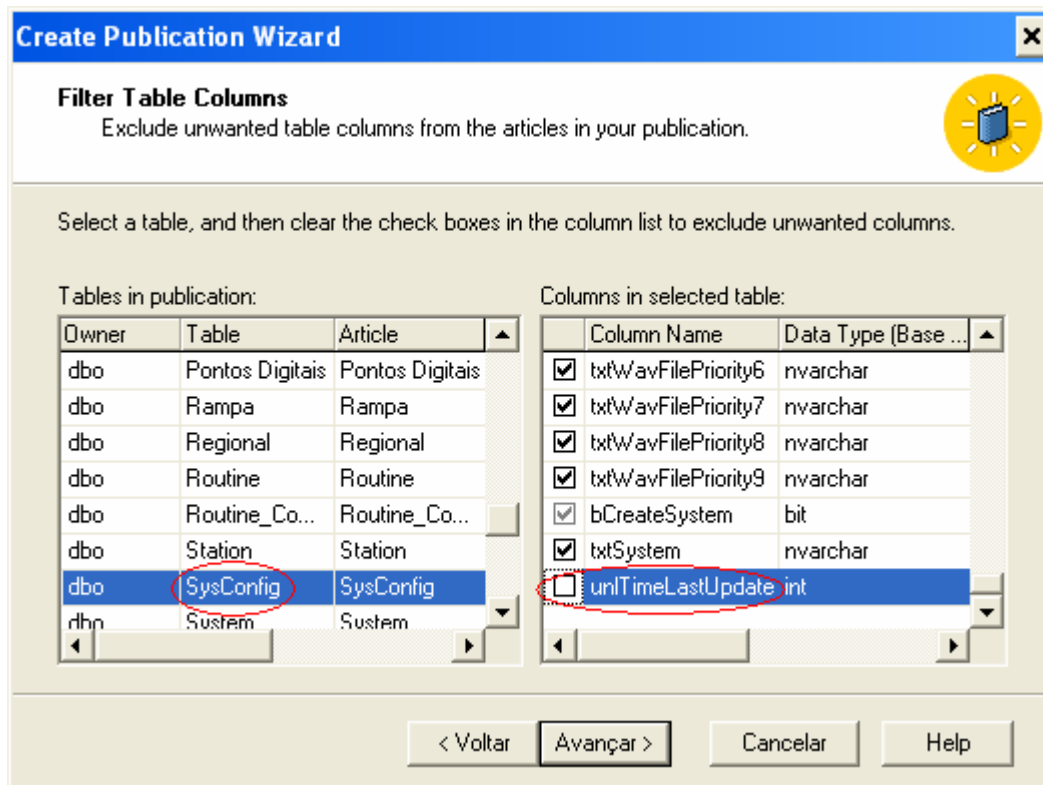


Pode-se filtrar os dados que serão publicados verticalmente, filtrando as colunas dos dados publicados, ou horizontalmente, filtrando as linhas dos dados publicados.

Selecione a opção de filtro de dados para replicação “Vertically, by filtering the columns of published data”. Com esta opção poderemos filtrar todos os campos de dados (colunas das tabelas) que serão publicados.

Clique em avançar.

4.3.8 - Filtro de colunas de tabelas

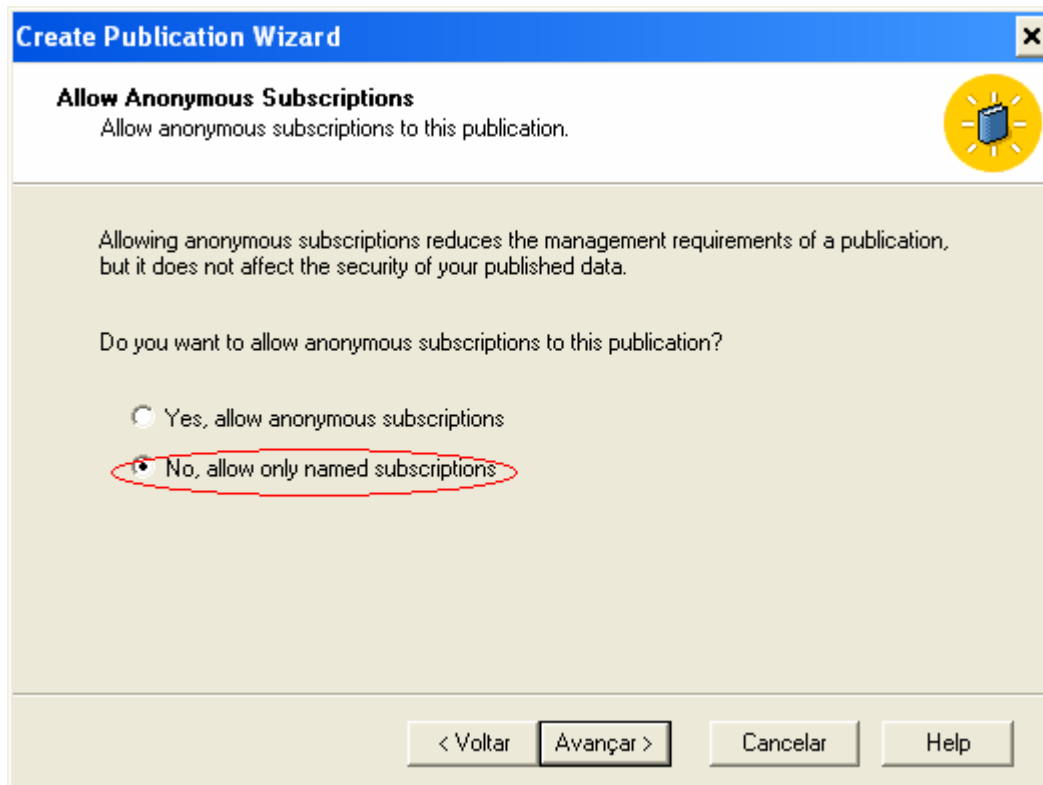


Pode-se excluir alguma coluna de algum artigo (tabela) que não se queira publicar.

Existe um campo especial da tabela SysConfig que não deve ser replicado. É o campo unTimeLastUpdate, que como vimos anteriormente é o campo onde o mestre grava o horário da máquina, permitindo aos clientes e ao escravo ActionView, na entrada do programa, determinarem qual a base de dados mais recente onde o mestre está gravando. Este campo deve ser desselecionado da publicação da base de dados.

Clique em avançar.

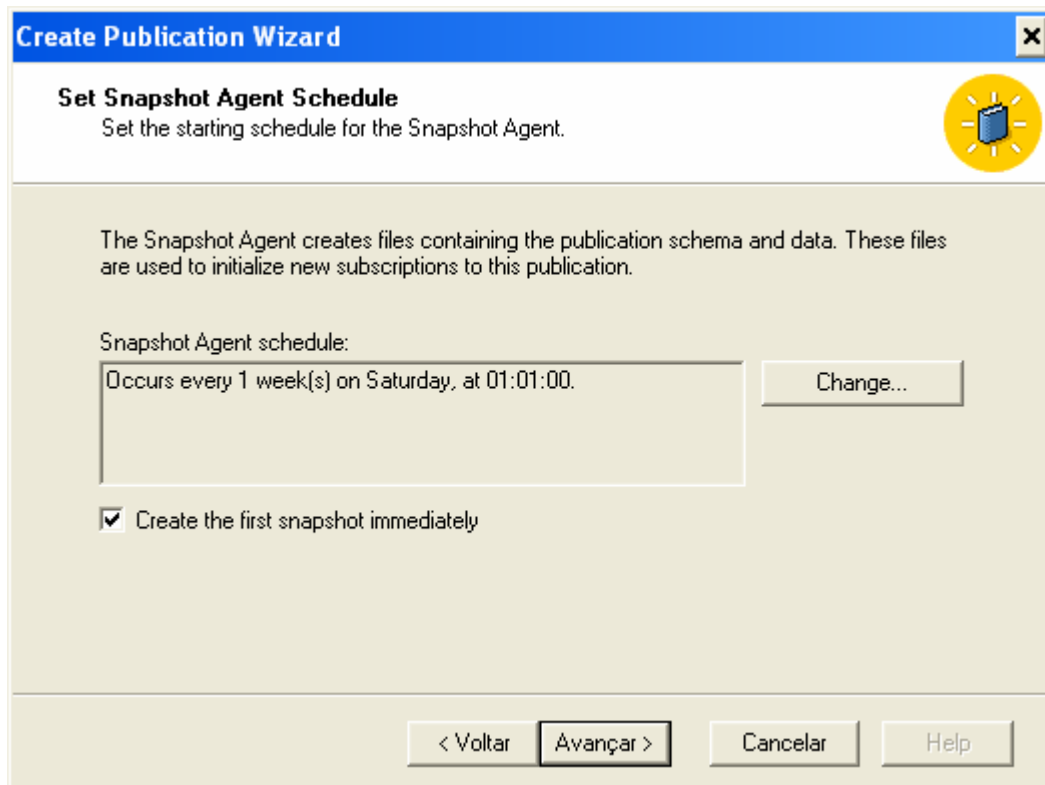
4.3.9 - Assinaturas anônimas



Em nossa publicação defina que somente serão permitidas assinaturas nomeadas. Todas as réplicas desta publicação deverão ser formalmente nomeadas.

Clique em avançar.

4.3.10 - Gerenciamento do Snapshot Agent

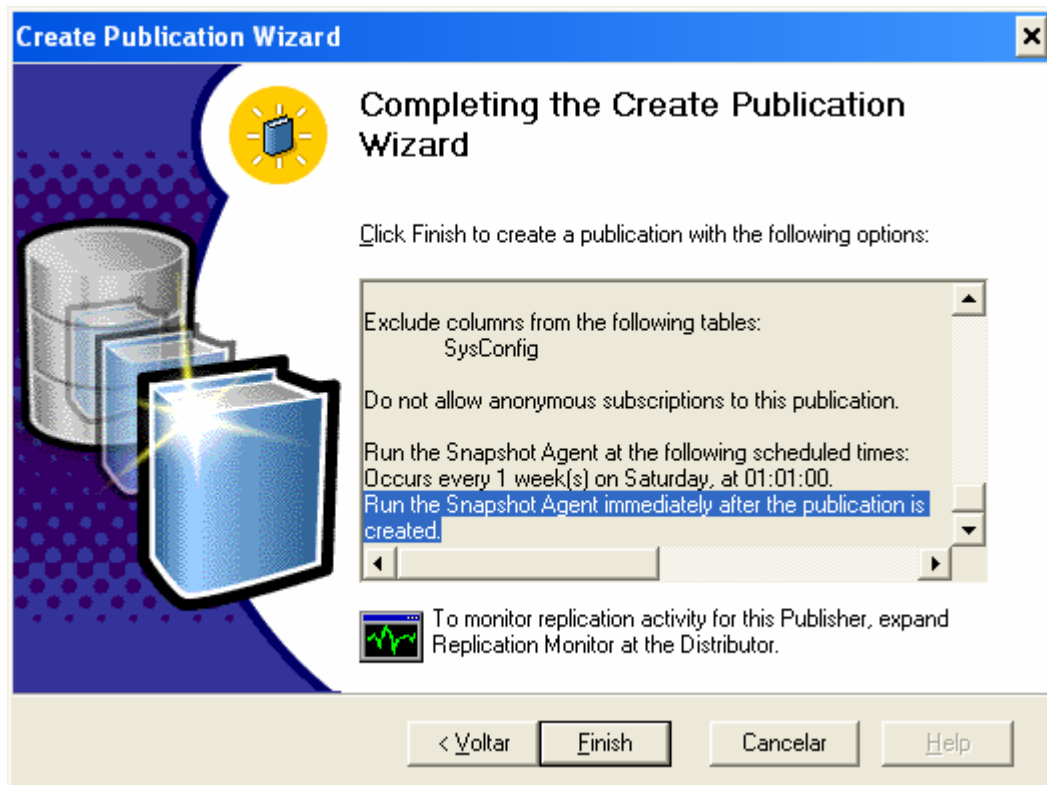


O Snapshot Agent cria os arquivos que contêm o esquema publicado (estrutura das tabelas do banco de dados) e os dados. Estes arquivos serão usados para inicializar novas assinaturas desta publicação.

Defina a frequência de criação destes arquivos. O padrão é de uma vez por semana aos Sábados à 01:01:00 hs.

Clique em avançar.

4.3.11 - Resumo do assistente de replicação



Nesta tela é apresentado um resumo de todas as propriedades que foram definidos para a publicação.

Create a merge publication from database 'Action'.

The following types of Subscribers may subscribe to this publication:

Servers running SQL Server 2000

[Somente será permitida a assinatura de bases de dados SQL Server 2000](#)

Publish the following tables as articles:

'Analog' as 'Analog'
 'Aplicativos' as 'Aplicativos'
 'Cameras' as 'Cameras'
 'SysConfig' as 'SysConfig'

[As tabelas relacionadas acima serão publicadas e se replicarão.](#)

The name of this publication is 'Action'. The description is 'Merge publication of Action database from Publisher GDESCLA.'.

Filter data in this publication using the following criteria.

Exclude columns from the following tables:
SysConfig

Aqui, o assistente da publicação nos informa que a tabela SysConfig terá campo(s) não publicado(s).

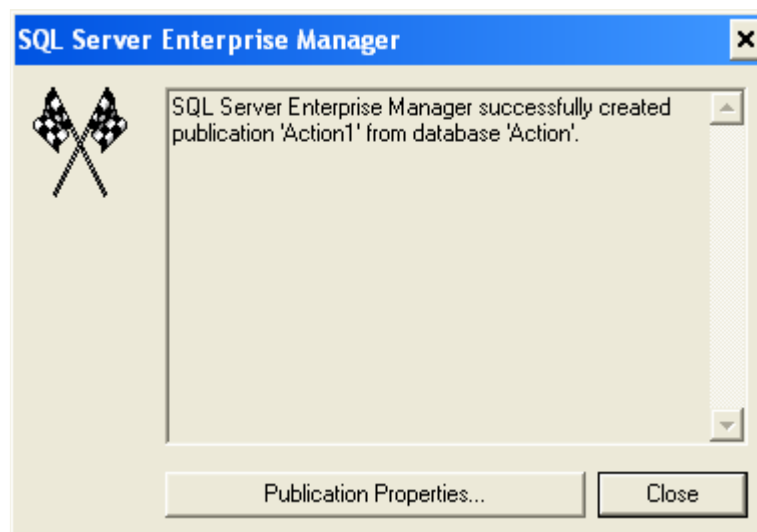
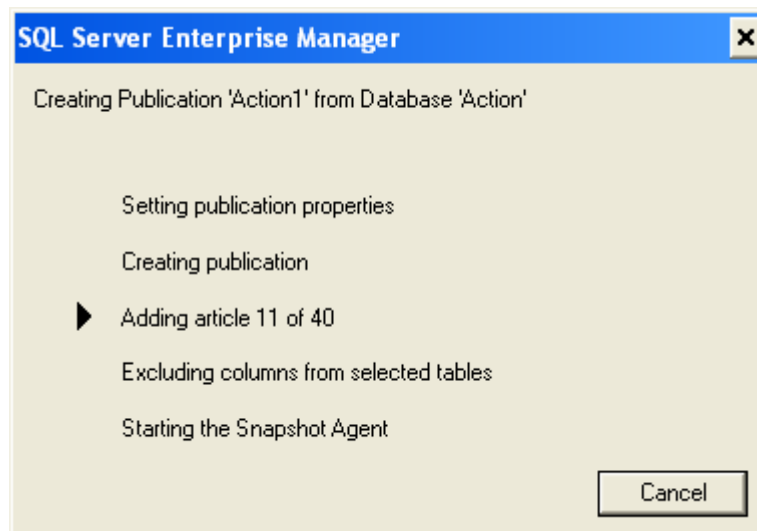
Do not allow anonymous subscriptions to this publication.

Não serão permitidas assinaturas anônimas.

Run the Snapshot Agent at the following scheduled times:
Occurs every 1 week(s) on Saturday, at 01:01:00.
Run the Snapshot Agent immediately after the publication is created.

O Agente de atualização dos arquivos necessários à criação de novas assinaturas será executado uma vez por semana, aos sábados, à 01:01:00hs.

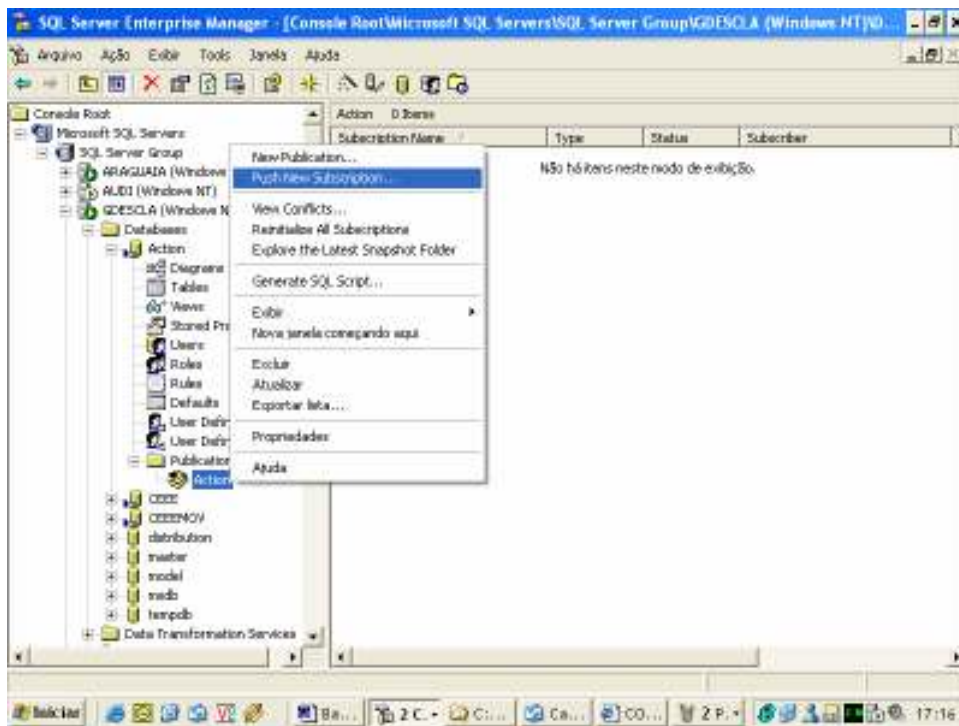
Clique em concluir para configurar a publicação.



4.4 - Adicionando uma assinatura

Feita a publicação do banco de dados, agora se deve criar uma assinatura deste banco de dados para o outro mestre. O Enterprise Manager é novamente a ferramenta a ser utilizada. Na máquina onde se criou a publicação e na outra máquina mestra o serviço do MSDE/SQL Server deverá ser iniciado automaticamente assim como o SQL Server Agent.

Na máquina onde foi publicado o banco, abra o item Publications do seu banco de dados, dê um clique com o botão direito no nome de sua publicação e clique em “Push New Subscription”.

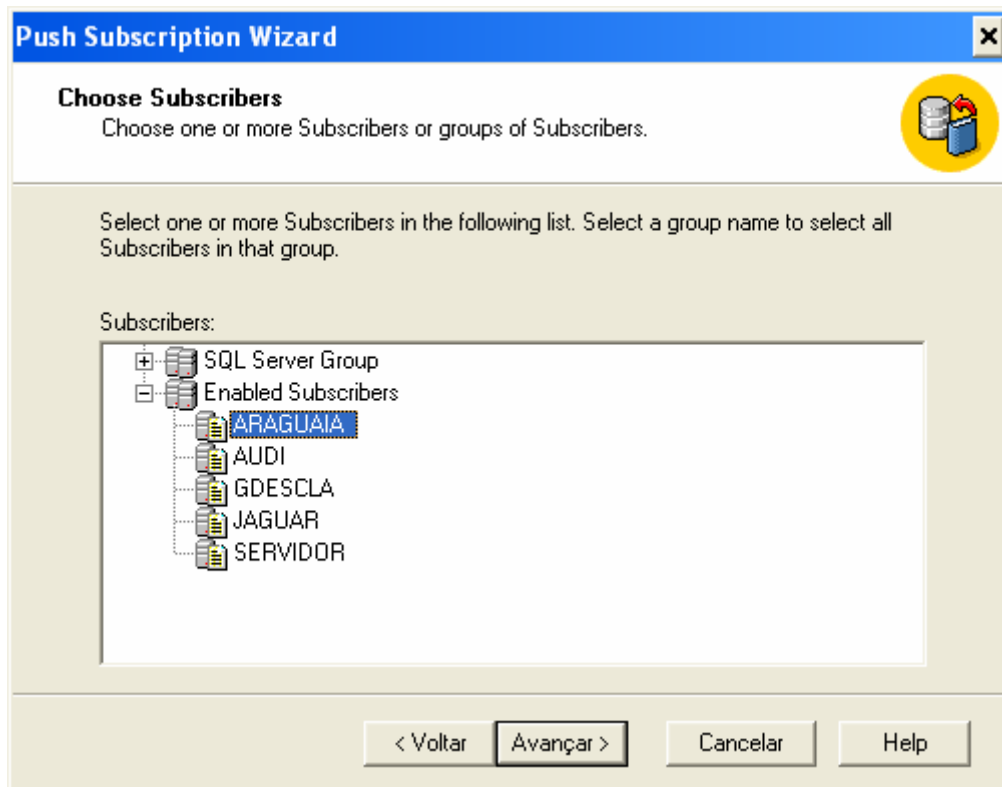


4.4.1 - Assistente para assinatura



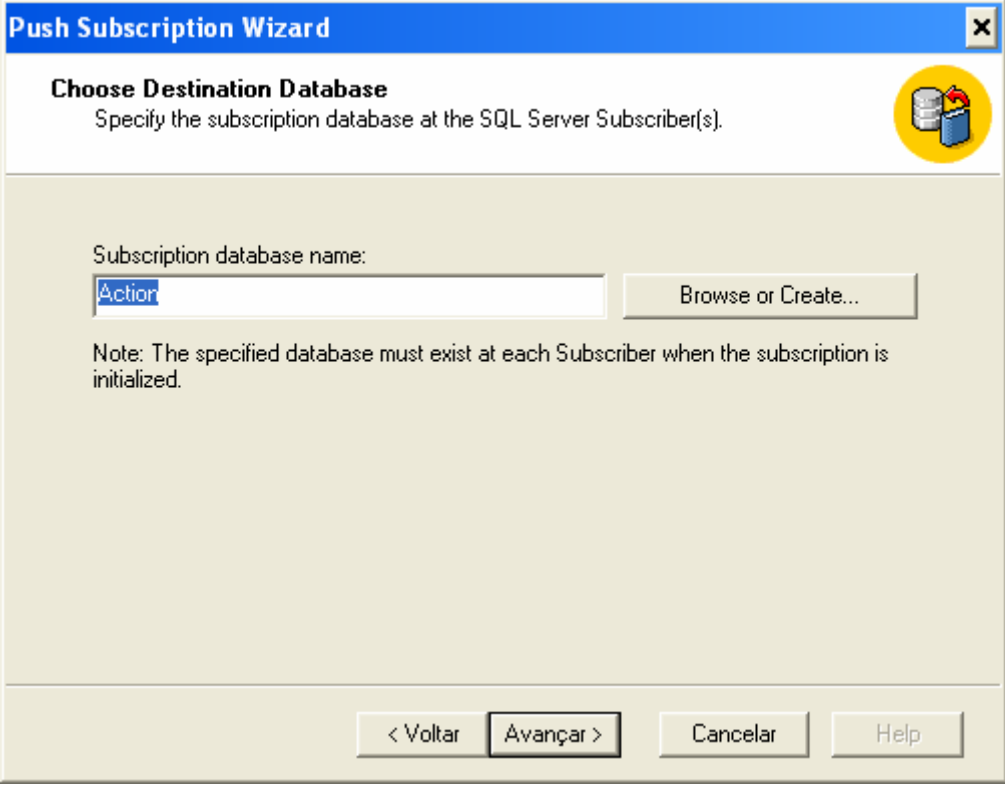
O assistente de assinatura guiará através de todo processo, definindo o(s) assinante(s), selecionando a base de dados onde a assinatura será criada e especificando a inicialização e frequência de sincronização com a publicação. Clique em avançar.

4.4.2 - Assinantes



Selecione o(s) assinante(s) desta publicação.
Clique em avançar.

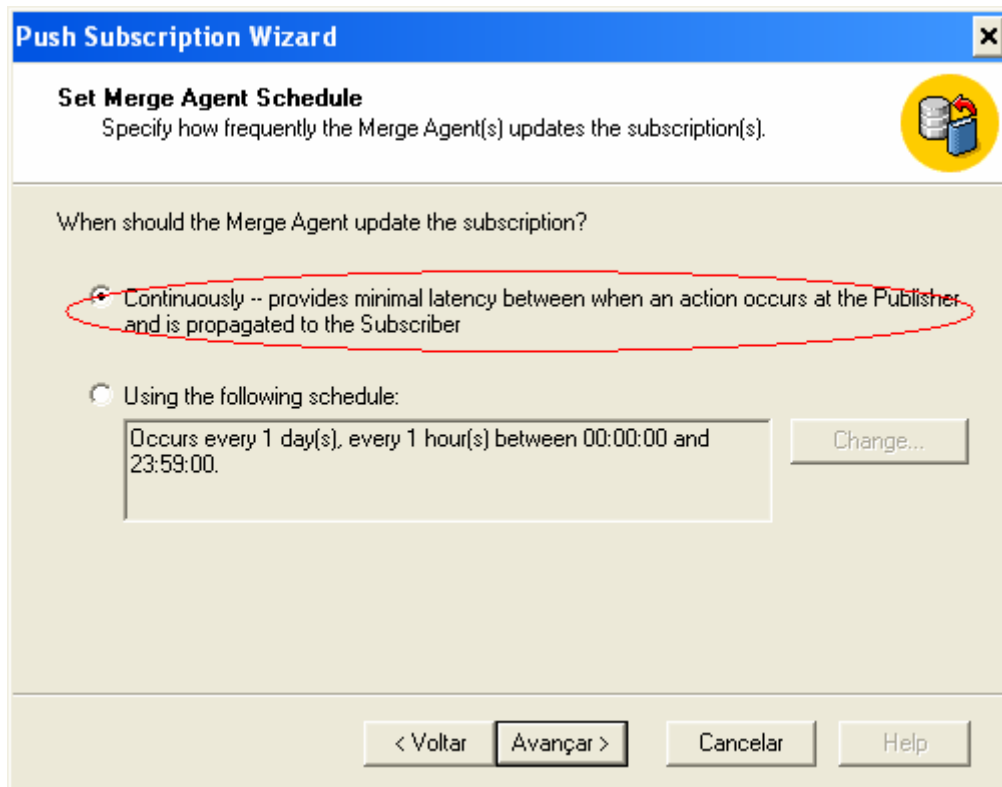
4.4.3 - Nome da base de dados no assinante



The screenshot shows a Windows-style dialog box titled "Push Subscription Wizard". The main heading is "Choose Destination Database" with the instruction "Specify the subscription database at the SQL Server Subscriber(s)". A text input field contains the name "Action". To the right of the input field is a button labeled "Browse or Create...". Below the input field is a note: "Note: The specified database must exist at each Subscriber when the subscription is initialized." At the bottom of the dialog are four buttons: "< Voltar", "Avançar >", "Cancelar", and "Help".

Selecione o nome da base de dados no assinante. Ela deverá existir ou ser criada através do botão "Browse or Create".
Clique em avançar.

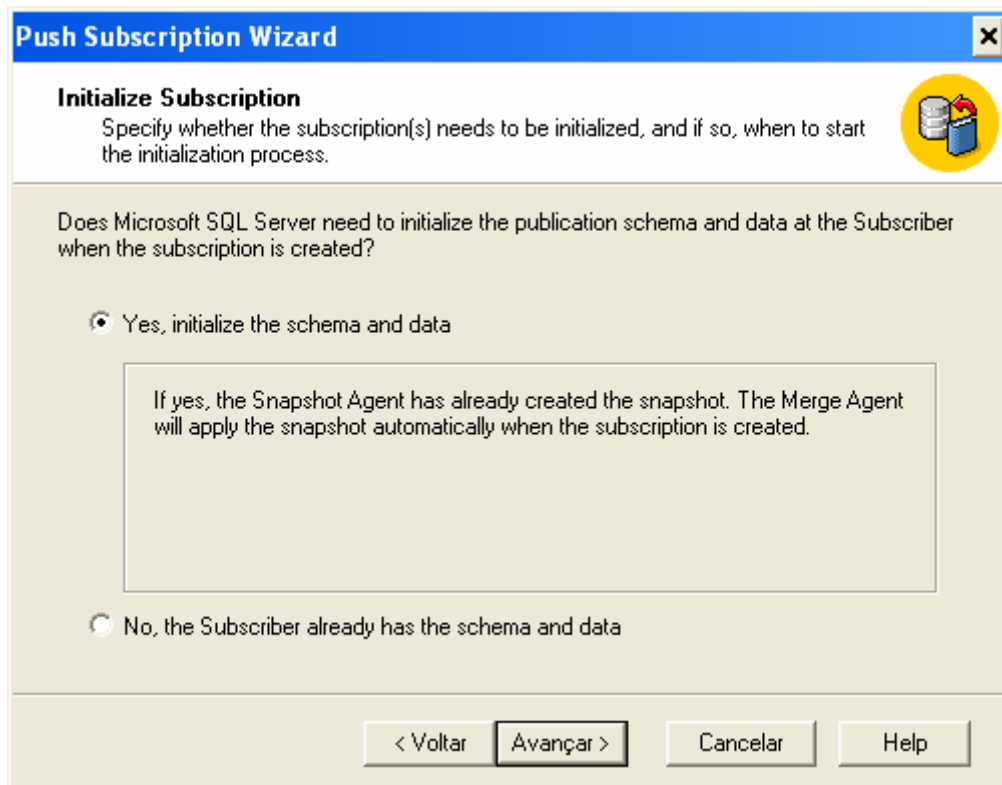
4.4.4 - Frequência de atualização da assinatura



Selecione a opção “Continuously -- ...” Assim haverá uma latência mínima entre uma alteração de dados em alguma base de dados para se propagar para a outra.

Clique em avançar.

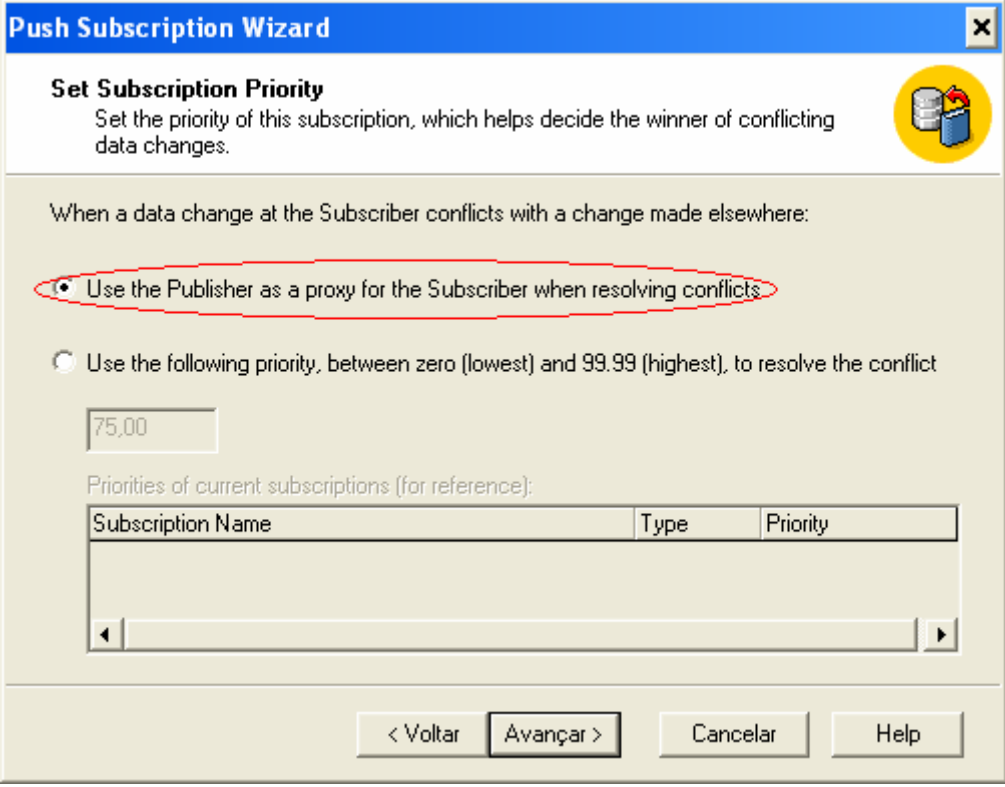
4.4.5 - Inicialização da assinatura



Marque a opção para inicializar esquema e dados da assinatura. Assim será criado um retrato fiel dos dados e da estrutura do banco de dados publicado.

Clique em avançar.

4.4.6 - Prioridade da assinatura para resolução de conflitos



Push Subscription Wizard

Set Subscription Priority
Set the priority of this subscription, which helps decide the winner of conflicting data changes.

When a data change at the Subscriber conflicts with a change made elsewhere:

- Use the Publisher as a proxy for the Subscriber when resolving conflicts
- Use the following priority, between zero (lowest) and 99.99 (highest), to resolve the conflict

75,00

Priorities of current subscriptions (for reference):

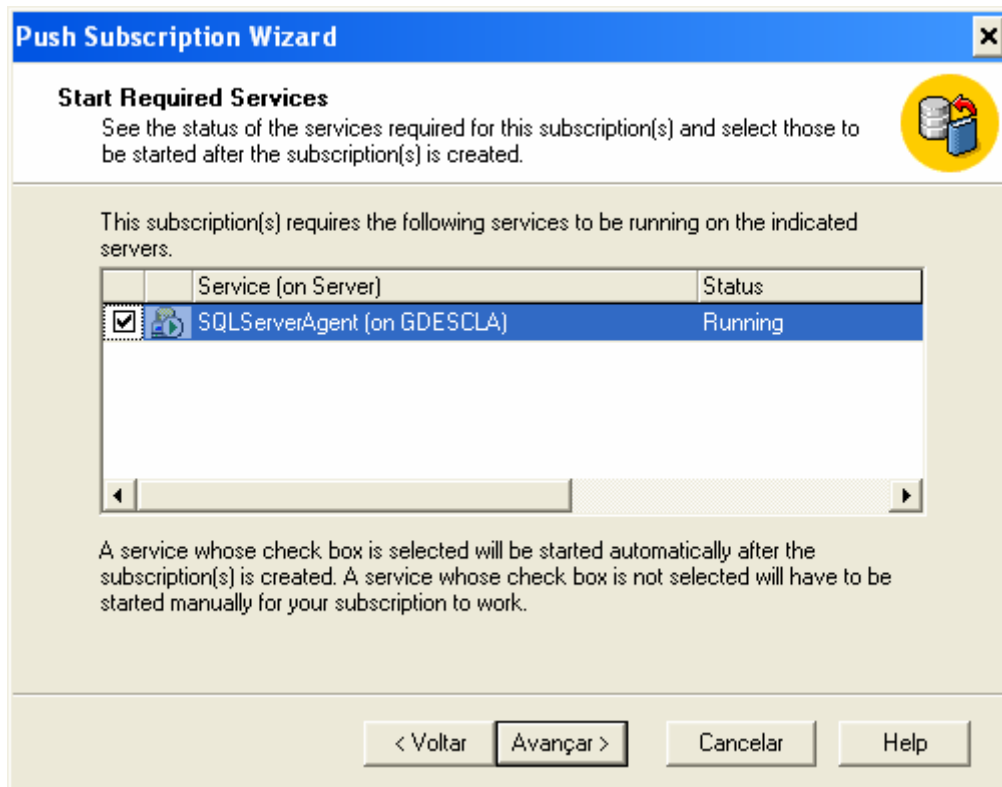
Subscription Name	Type	Priority

< Voltar Avançar > Cancelar Help

Para resolução de conflitos ocorridos durante a sincronização entre as bases de dados, pode-se associar prioridades para as assinaturas que determinarão o “vencedor” do conflito. Selecione a opção “Use the Publisher as a proxy for the Subscriber when resolving conflicts” para que o Publicador resolva os conflitos segundo seu próprio algoritmo de resolução de conflitos.

Clique em avançar.

4.4.7 - Serviços necessários para a assinatura



Para a assinatura se sincronizar corretamente, é necessária a execução de determinados serviços. Esta tela apresenta estes serviços que devem estar marcados para ser iniciados automaticamente com o computador.

Clique em avançar.

4.4.8 - Concluindo a assinatura



Aqui é apresentado um resumo do que o assistente fará para criar esta assinatura.

Create a subscription to publication 'Action' from Publisher 'GDESCLA'.

Create a subscription at the following Subscriber(s):
ARAGUAIA

[Criar assinatura na máquina <MAQUINA>](#).

Create each subscription in a database named 'Action'.

Run the agent(s) that synchronizes the subscription(s) at the Distributor.

[Executar o agente sincronizador da assinatura no distribuidor \(Publicador\)](#).

Run the Merge Agent(s) continuously.

[Executar o Merge Agent continuamente](#).

Initialize the subscription(s) immediately.

When resolving conflicts, the subscription(s) uses the priority of the Publisher, GDESCLA.

[Os conflitos serão resolvidos usando a prioridade da publicação](#).

Verify that the following services are started:
SQLServerAgent (on GDESCLA)

Clique em Finish.

5 - Banco de dados histórico

Para a base de dados histórica, o procedimento adotado deve ser exatamente o mesmo para a base paramétrica, a não ser pelo script a ser executado. No item 4.1.4 são relacionados os scripts necessários para definir o esquema das tabelas na base de dados. O script para a base de dados de histórico é:

Actmov.SQL

O restante do processo é exatamente igual.

Apêndice

Transaction Logs

A database in Microsoft® SQL Server™ 2000 has at least one data file and one transaction log file. Data and transaction log information is never mixed on the same file, and individual files are used by only one database.

SQL Server uses the transaction log of each database to recover transactions. The transaction log is a serial record of all modifications that have occurred in the database as well as the transaction that performed each modification. The transaction log records the start of each transaction. It records the changes to the data and enough information to undo the modifications (if necessary later) made during each transaction. For some large operations, such as CREATE INDEX, the transaction log instead records the fact that the operation took place. The log grows continuously as logged operations occur in the database.

The transaction log records the allocation and deallocation of pages and the commit or rollback of each transaction. This allows SQL Server either to apply (roll forward) or back out (roll back) each transaction in the following ways:

- A transaction is rolled forward when you apply a transaction log. SQL Server copies the after image of every modification to the database or reruns statements such as CREATE INDEX. These actions are applied in the same sequence in which they originally occurred. At the end of this process, the database is in the same state it was in at the time the transaction log was backed up.
- A transaction is rolled back when you back out an incomplete transaction. SQL Server copies the before images of all modifications to the database since the BEGIN TRANSACTION. If it encounters transaction log records indicating that a CREATE INDEX was performed, it performs operations that logically reverse the statement. These before images and CREATE INDEX reversals are applied in the reverse of their original sequence.

At a checkpoint, SQL Server ensures that all transaction log records and database pages modified are written to disk. During the recovery process of each database that occurs when SQL Server is restarted, a transaction must be rolled forward only when it is not known whether all the data modifications in the transaction were actually written from the SQL Server buffer cache to disk. Because a checkpoint forces all modified pages to disk, it represents the point at which the startup recovery must start rolling forward transactions. Because all pages modified before the checkpoint are guaranteed to be on disk, there is no need to roll forward anything done before the checkpoint.

Transaction log backups enable you to recover the database to a specific point in time (for example, prior to entering unwanted data), or to the point of failure. Transaction log backups should be a consideration in your media recovery strategy. For more information, see [Selecting a Recovery Model](#).

Replication Model

Microsoft® SQL Server™ 2000 replication uses a publishing industry metaphor to represent the components and processes in a [replication topology](#). The model is composed of the following: Publisher, Distributor, Subscribers, Publications, articles, and subscriptions.

There are also several replication processes that are responsible for copying and moving data between the Publisher and Subscriber. These are the Snapshot Agent, Distribution Agent, Log Reader Agent, Queue Reader Agent, and Merge Agent. For more information about the agent processes, see [Agents and Monitors](#).

Publisher

The Publisher is a server that makes data available for replication to other servers. The Publisher can have one or more publications, each representing a logically related set of data. In addition to being the server where you specify which data is to be replicated, the Publisher also detects which data has changed during transactional replication and maintains information about all publications at that site.

Distributor

The Distributor is a server that hosts the distribution database and stores history data, and/or transactions and meta data. The role of the Distributor varies depending on which type of replication you implement. For more information, see [Types of Replication](#).

A remote Distributor is a server that is separate from the Publisher and is configured as a Distributor of replication. A local Distributor is a server that is configured to be both a Publisher and a Distributor of replication.

Subscribers

Subscribers are servers that receive replicated data. Subscribers subscribe to publications, not to individual articles within a publication, and they subscribe only to the publications that they need, not all of the publications available on a Publisher. Depending on the type of replication and replication options you choose, the Subscriber could also propagate data changes back to the Publisher or republish the data to other Subscribers.

Publication

A publication is a collection of one or more articles from one database. This grouping of multiple articles makes it easier to specify a logically related set of data and database objects that you want to replicate together.

Article

An article is a table of data, a partition of data, or a database object that is specified for replication. An article can be an entire table, certain columns (using a vertical filter), certain rows (using a horizontal filter), a stored procedure or view definition, the execution of a stored procedure, a view, an indexed view, or a user-defined function.

Subscription

A subscription is a request for a copy of data or database objects to be replicated. A subscription defines what publication will be received, where, and when. Synchronization or data distribution of a subscription can be requested either by the Publisher (a [push subscription](#)) or by the Subscriber (a [pull subscription](#)). A publication can support a mixture of push and pull subscriptions.

See Also

[Implementing Replication](#)

[Publishers, Distributors, and Subscribers](#)

[Subscribing to Publications](#)