Replication Sample Script

Last updated by | Vitor Tomaz | Feb 24, 2023 at 3:31 AM PST

Contents

Replication Sample Script

This is a "How To" TSG to provide you with a simple example for setting up a Managed Instance for Transactional Replication. It configures the MI for distribution, then adds Publisher and Subscriber databases, creates sample tables with some data, and configures publication and subscription.

The script assumes that you already have a file share storage account. See How to configure the snapshot directory for steps to achieve this.

Configure Distribution

```sql
-- Create the SQL Login for the replication agents
SET NOCOUNT ON
GO
USE [master]
GO
CREATE LOGIN TRANREPLADMIN WITH PASSWORD = '$trongPa11word';
GO

-- Add the distributor and the distribution database.
USE [master]
GO
-- check if distribution has already been configured:
exec sp_get_distributor
GO
-- check server name to verify to be connected to the correct server
select @@servername
GO

-- Fill in the name of the Distributor instance as needed:
-- The password is the "Administrative Link password" that you also see on the Distributor Properties page in
EXEC sp_adddistributor @distributor = @@ServerName, @password = N'$trongPa11word';
GO
EXEC sp_adddistributiondb @database = N'Distribution';
GO

-- Configure the remote Publisher at the Distributor
-- CRITICAL: Port 445 (TCP outbound) needs to be open in the security rules of the Managed Instance subnet to
EXEC sp_adddistpublisher @publisher = @@ServerName,
    @distribution_db = N'Distribution',
    @security_mode = 0,
    @login = N'TRANREPLADMIN',
    @password = N'$trongPa11word',
    @working_directory = N'\\yourstorageaccount.file.core.windows.net\snapshots',
    @storage_connection_string = N'DefaultEndpointsProtocol=https;AccountName=yourstorageaccount;AccountKey=9
GO

-- Configure the SQL user for the distribution database
USE [Distribution]
GO
CREATE USER TRANREPLADMIN FROM LOGIN TRANREPLADMIN
GO
EXEC sp_addrolemember N'db_owner', N'TRANREPLADMIN'
GO
USE [master]
GO
```

Create databases and tables

```sql
-- Create the publisher and subscriber databases
SET NOCOUNT ON
GO
USE [master]
GO
CREATE DATABASE [Repl_PUB]
GO
CREATE DATABASE [Repl_SUB]
GO

USE [Repl_PUB]
GO
CREATE USER TRANREPLADMIN FROM LOGIN TRANREPLADMIN
GO
EXEC sp_addrolemember N'db_owner', N'TRANREPLADMIN'
GO
USE [Repl_SUB]
GO
CREATE USER TRANREPLADMIN FROM LOGIN TRANREPLADMIN
GO
EXEC sp_addrolemember N'db_owner', N'TRANREPLADMIN'
GO
USE [master]
GO

-- Create the sample tables and add a few rows
USE [Repl_PUB]
GO
CREATE TABLE [dbo].[MainTable] (
    [ID] [int] PRIMARY KEY CLUSTERED NOT NULL ,
    [c1] [varchar](100) NULL
)
GO
CREATE TABLE [dbo].[SubTable] (
    [ID] [int] PRIMARY KEY CLUSTERED NOT NULL,
    [MainTableID] [int] NOT NULL,
    [c1] [varchar](100) NULL
    , CONSTRAINT FK_Main_Sub FOREIGN KEY (MainTableID) REFERENCES dbo.MainTable (ID) --NOT FOR REPLICATION
)
GO

INSERT INTO [dbo].[MainTable] (ID, c1) VALUES (1, 'original insert')
INSERT INTO [dbo].[SubTable] (ID, MainTableID, c1) VALUES (1, 1, 'original insert')
GO
SELECT * FROM MainTable
SELECT * FROM SubTable
GO
```

Publish the database

```
-- check if distribution has already been configured:
USE [master]
GO
exec sp_get_distributor
GO


-- Enable the replication database for publication
USE [master]
GO
exec sp_replicationdboption @dbname = N'Repl_PUB', @optname = N'publish', @value = N'true'
GO
-- configure the log reader agent
exec [Repl_PUB].sys.sp_addlogreader_agent @job_login = 'TRANREPLADMIN', @job_password = '$trongPa11word', @pub
    @publisher_login = 'TRANREPLADMIN', @publisher_password = '$trongPa11word'
GO


-- Add Publication
use [Repl_PUB]
exec sp_addpublication @publication = N'Publication_Tran', @description = N'Simple Transactional publication',
    @sync_method = N'native', @retention = 0,
    @allow_push = N'true', @allow_pull = N'true', @allow_anonymous = N'true', @enabled_for_internet = N'false
    @snapshot_in_defaultfolder = N'true', @compress_snapshot = N'false', @ftp_port = 21, @ftp_login = N'anony
    @allow_subscription_copy = N'false', @add_to_active_directory = N'false',
    @repl_freq = N'continuous', @status = N'active', @independent_agent = N'true',
    @immediate_sync = N'true',
    @allow_sync_tran = N'false', @autogen_sync_procs = N'false', @allow_queued_tran = N'false', @allow_dts =
    @replicate_ddl = 1, @allow_initialize_from_backup = N'false', @enabled_for_p2p = N'false', @enabled_for_h
GO
-- configure the Snapshot agent
-- (sets an inactive schedule so that you can start it manually for your tests)
exec sp_addpublication_snapshot @publication = N'Publication_Tran',
    @frequency_type = 1, @frequency_interval = 0, @frequency_relative_interval = 0, @frequency_recurrence_fac
    @active_start_time_of_day = 0, @active_end_time_of_day = 235959,
    @active_start_date = 0, @active_end_date = 20051231,
    @job_login = 'TRANREPLADMIN', @job_password = '$trongPa11word',
    @publisher_security_mode = 0, @publisher_login = 'TRANREPLADMIN', @publisher_password = '$trongPa11word'
GO
-- Grant permissions to publication
exec sp_grant_publication_access @publication = N'Publication_Tran', @login = N'sa'
GO
exec sp_grant_publication_access @publication = N'Publication_Tran', @login = N'TRANREPLADMIN'
GO


-- Add the articles
use [Repl_PUB]
exec sp_addarticle @publication = N'Publication_Tran',
    @article = N'MainTable', @source_owner = N'dbo', @source_object = N'MainTable', @destination_table = N'Ma
    @type = N'logbased', @description = N'', @creation_script = N'',
    @pre_creation_cmd = N'drop',
    @schema_option = 0x000000000803529F, @identityrangemanagementoption = N'none',
    @status = 24, @vertical_partition = N'false',
    @ins_cmd = N'CALL [dbo].[sp_MSins_dboMainTable]', @del_cmd = N'CALL [dbo].[sp_MSdel_dboMainTable]', @upd_
GO
exec sp_addarticle @publication = N'Publication_Tran',
    @article = N'SubTable', @source_owner = N'dbo', @source_object = N'SubTable', @destination_table = N'SubT
    @type = N'logbased', @description = N'', @creation_script = N'',
    @pre_creation_cmd = N'drop',
    @schema_option = 0x000000000803529F, @identityrangemanagementoption = N'none',
    @status = 24, @vertical_partition = N'false',
    @ins_cmd = N'CALL [dbo].[sp_MSins_dboSubTable]', @del_cmd = N'CALL [dbo].[sp_MSdel_dboSubTable]', @upd_cm
GO


-- Add the subscription
-- This is a Push subscription into a Managed Instance (using the same instance as the publisher)
use [Repl_PUB]
exec sp_addsubscription @publication = N'Publication_Tran',
    @subscriber = @@servername, @destination_db = N'Repl_SUB',
    @subscription_type = N'Push', @sync_type = N'automatic', @article = N'all',
```

```
        @update_mode = N'read only', @subscriber_type = 0
GO
-- Configure the Distribution Agent
exec sp_addpushsubscription_agent @publication = N'Publication_Tran',
        @subscriber = @@servername, @subscriber_db = N'Repl_SUB',
        @job_login = 'TRANREPLADMIN', @job_password = '$trongPa11word', @subscriber_security_mode = 0, @subscribe
        @frequency_type = 64, @frequency_interval = 0, @frequency_relative_interval = 0, @frequency_recurrence_fa
--      @frequency_type = 4, @frequency_interval = 1, @frequency_relative_interval = 0, @frequency_recurrence_f
GO

-- Start replication agents (if not already running)
USE [master]
GO
-- Snapshot agent
exec Repl_PUB..sp_startpublication_snapshot @publication = N'Publication_Tran'
--waitfor delay '00:00:30'
GO
-- Distribution agent
exec Repl_PUB..sp_startpushsubscription_agent @publication = N'Publication_Tran', @subscriber = @@servername,
--waitfor delay '00:00:10'
GO

-- Check Replication Monitor here for success agent execution

-- Check the results of the initial synchronization
SELECT * from Repl_PUB..MainTable
SELECT * from Repl_SUB..MainTable
SELECT * from Repl_PUB..SubTable
SELECT * from Repl_SUB..SubTable


/*** Initial configuration complete ****************/
```

◄ ▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬   ►

Add more sample data

```
INSERT INTO [Repl_PUB].[dbo].[MainTable] (ID, c1) VALUES (2, 'second insert')
GO
set nocount on
declare @i int = 101
while @i < 1000
begin
    INSERT INTO [Repl_PUB].[dbo].[SubTable] (ID, MainTableID, c1) VALUES (@i, 2, 'second insert')
    set @i = @i + 1
end
set nocount off
GO

-- Distribution agent - start if not already running in -Continuous mode
exec Repl_PUB..sp_startpushsubscription_agent @publication = N'Publication_Tran', @subscriber = @@servername,
go

-- wait a moment before verifying the outcome
SELECT count(*) from Repl_PUB..MainTable   -- 2
SELECT count(*) from Repl_SUB..MainTable   -- 2
SELECT count(*) from Repl_PUB..SubTable    -- 900
SELECT count(*) from Repl_SUB..SubTable    -- 900
```

◄ ▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬   ►

Simulate error 20598 Row not found

```sql
-- check existence of row
SELECT * from Repl_PUB..SubTable where ID = 404
SELECT * from Repl_SUB..SubTable where ID = 404

-- Simulate a simple but nasty failure - delete a row directly from the Subscriber
delete from Repl_SUB..SubTable where ID = 404
go
--- ... then update the corresponding row at the Publisher:
update Repl_PUB..SubTable set c1 = 'publisher update' where ID = 404
go
-- Distribution Agent failing after retries with:
-- "The row was not found at the Subscriber when applying the replicated UPDATE command for Table '[dbo].[SubT

-- show what commands are pending in distribution
exec Distribution..sp_browsereplcmds;

-- Pile up some further changes to demonstrate latency
INSERT INTO [Repl_PUB].[dbo].[MainTable] (ID, c1) VALUES (3, 'third insert')
GO
set nocount on
declare @i int = 1001
while @i < 2000
begin
        INSERT INTO [Repl_PUB].[dbo].[SubTable] (ID, MainTableID, c1) VALUES (@i, 3, 'third insert')
        set @i = @i + 1
end
set nocount off
GO
--delete [Repl_PUB].[dbo].[SubTable] where ID > 1000 and ID < 2000

-- Fix error 20598:
INSERT INTO [Repl_SUB].[dbo].[SubTable] (ID, MainTableID, c1) VALUES (404, 2, 'subscriber insert')
GO

-- check the outcome
-- note the value of the subscriber column
SELECT * from Repl_PUB..SubTable where ID = 404
SELECT * from Repl_SUB..SubTable where ID = 404
```
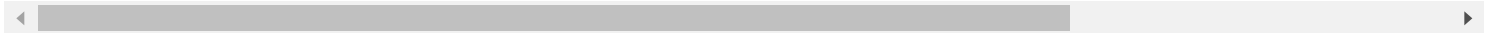
How good have you found this content?