**SQL replication with a Publisher Database in Always On Availability Groups**

**Prerequisites**:

1. Replication feature should be installed server
2. Remote distribution server.
3. publishers to use the remote distribution
4. Set up of Always on Availability Group
5. Configuring the SQL Server replication
6. Need to Enable replication for database going to use:

USE master

EXEC sp\_replicationdboption @dbname = 'AdventureWorks2017',

@optname = 'publish',

@value = 'true'

GO

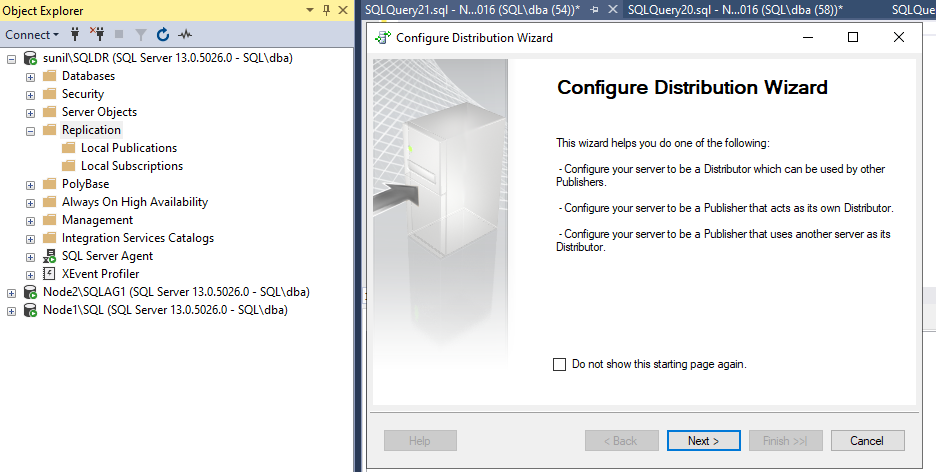
**Environment:**

* **Node1**\SQL– Primary server for AG & **Publisher**
* **Node2\**SQLAG1– Secondary server of AG
* **Node3\SQLDR**(Sunil)– Remote **distribution server**
* **Node3\SQLDR**(Sunil)– **Subscriber server**

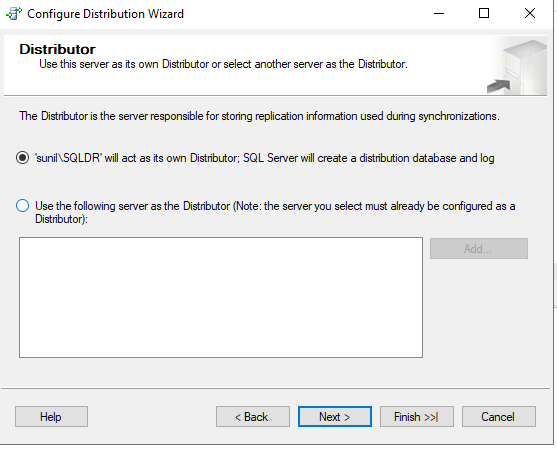
**STEPS:**

1. Configuring the remote distribution on Node3.

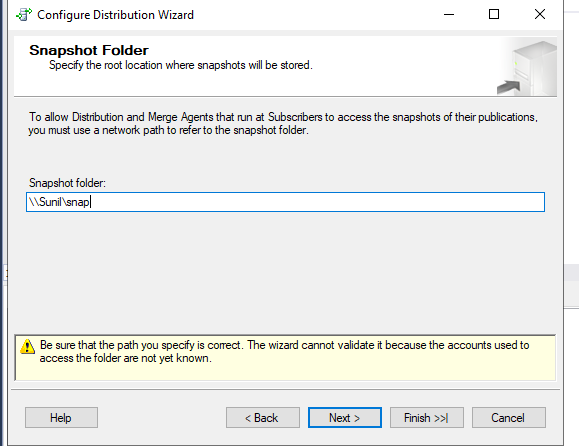
Log in to the distribution server **Node3** using SQL Server Management Studio, navigate to the **Replication** folder, Right-click and click on **Configure Distribution**. Click **Next** on Configure Distribution Wizard window.



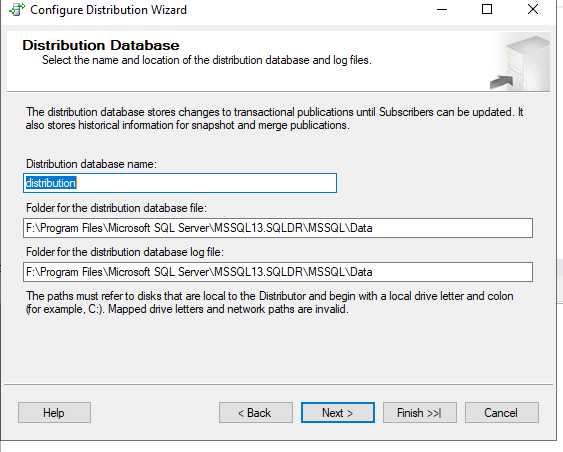
Select the first option i.e. **‘Node3 will act as its own Distributor; SQL Server will create a distribution database and log**and click **Next.**



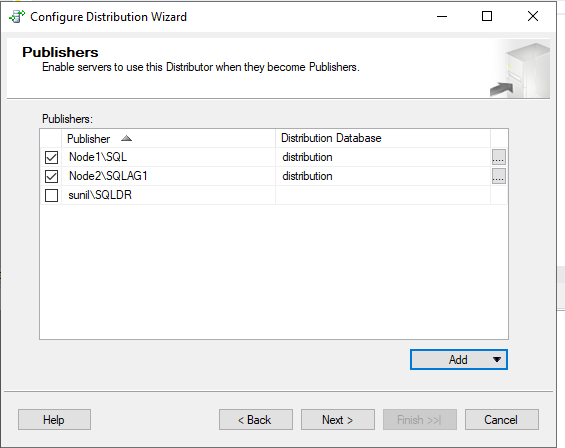
Provide **shared** Path of **SNAPSHOT** folder Next



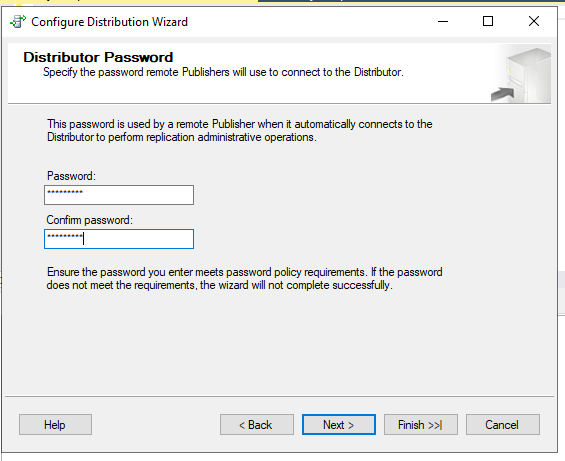
if you want a custom name, Enter the name of the distribution database else set it to default. Enter the location of the data and log file of the distribution database and click **Next**.



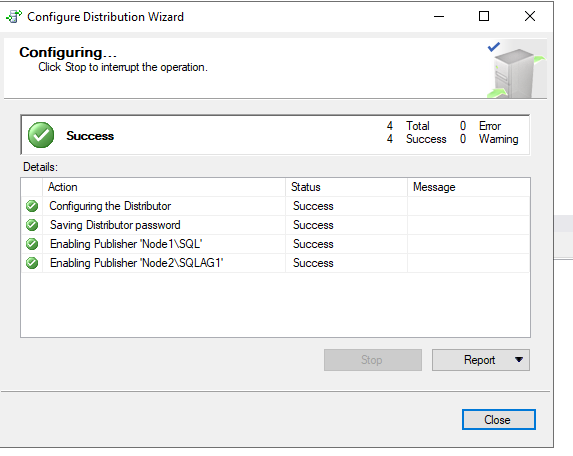
Add possible publisher server name I.e. **Node1\SQL** & **Node2\SQLAG1**



Specify password for remote **Publishers** will use to connect **Distributor**



Click **Next.**Select configure distribution, Click **Next,** and **Finish**,

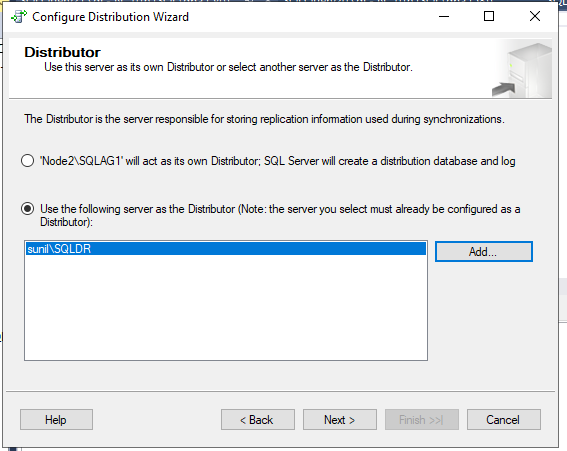


Now we need to configure the publishers Node1 and Node2 to use remote distribution server Node3 which was set up above

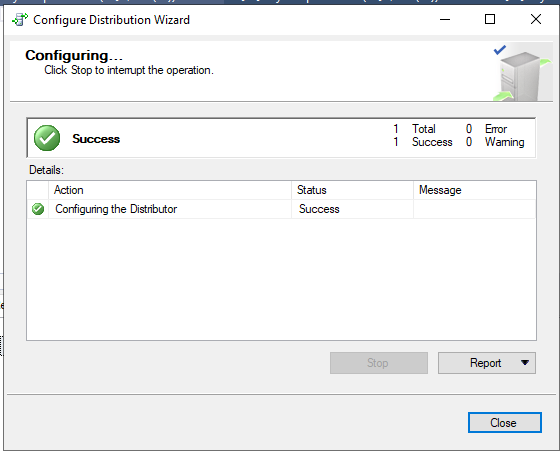
1. **Now Configure Distribution on Publisher server:**

Log in to the primary node Node1 using SQL Server Management Studio. Navigate to the **Replication** folder. Right-click on the **Replication** folder and click on **Configure Distribution**.

Click **Next.**select “**Use the following server as distributor**”. Click on **Add** and add the remote distribution server Node3 which was configured above. Click **Next.**



Enter the administrative password you set above. Click **Next ->**Enable **Configure Distribution ->**Click **Next**and Finish.

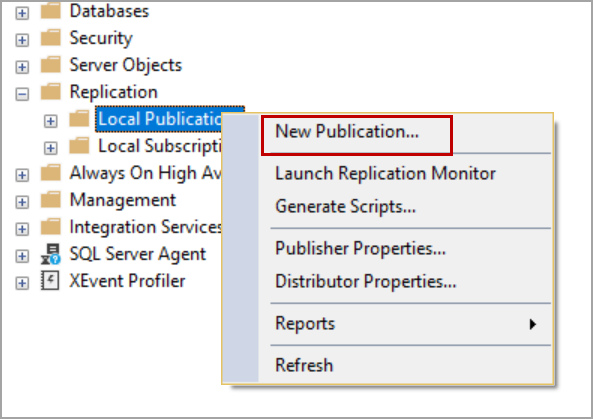


Similarly, configure the secondary node Node2 to use the same remote distribution server. If you have more nodes in the Availability group,

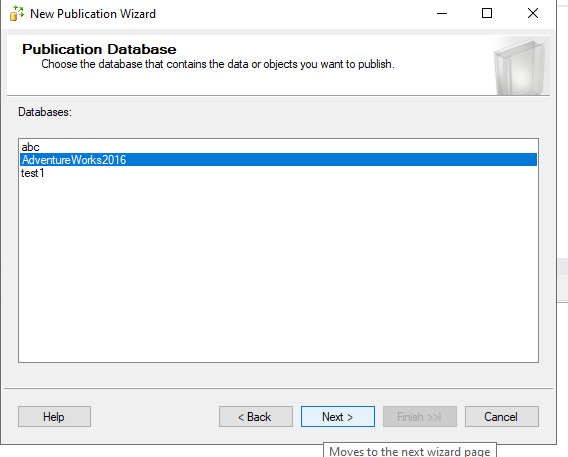
Configure all the possible failover nodes to use the same remote distribution server.

## Configuring the SQL Server replication :

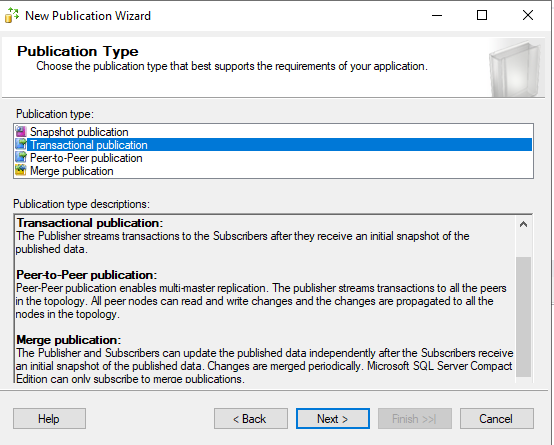
Log in to primary node Node1 using SQL Server Management Studio, Navigate to the **Replication** folder and then **Local Publications**. Right-click and click on **New Publication**.



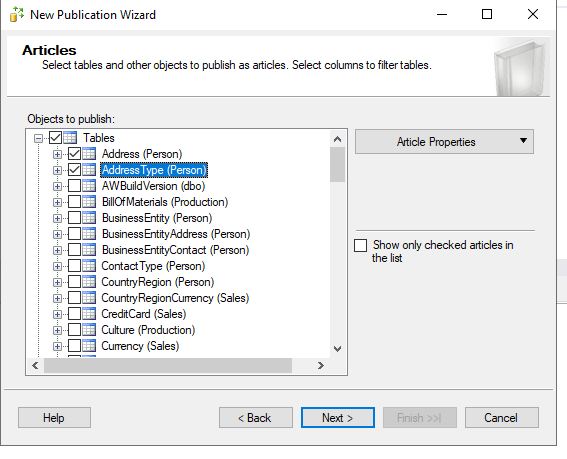
Click Next and select the **database** that you want to replicate.



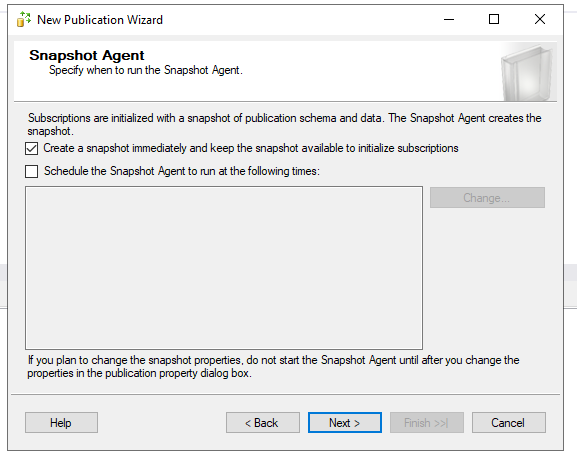
Click **Next** and select the type of replication.

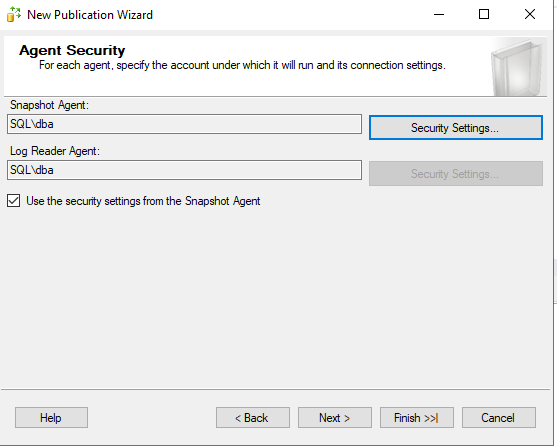


Select the tables you want to replicate and click **Next**.



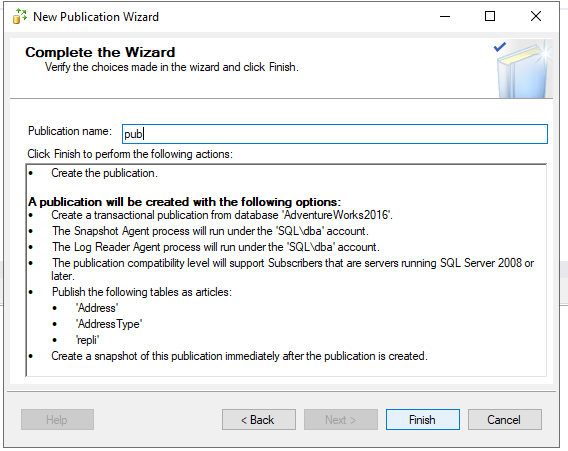
Enable **Create Snapshot Immediately** and Click **Next**.



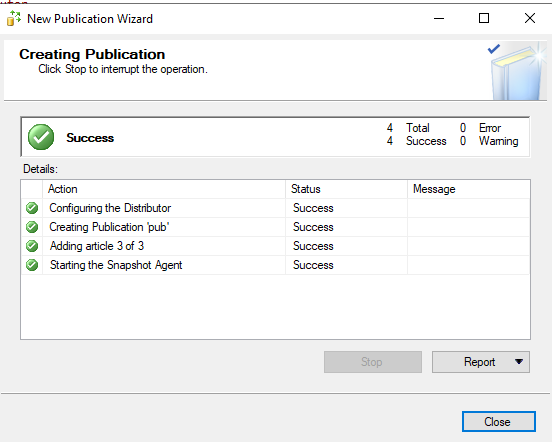
Configure **Agent Security**. click **Next.** 

Enable **Create the Publication.**Click **Next,**

**E**nter the name of the publication & in SQL Replication and click **Finish**.



Once you click on Finish, the publication is created, and the snapshot agent is started which generates the snapshot.



1. **Configure Node2 for Use remote distributor:**

Configure the secondary node Node2 to use the same remote distribution server. If you have more nodes in the Availability group,

Configure all the possible failover nodes to use the same remote distribution server. Using below query.

EXEC sp\_adddistributor

@distributor = 'sunil\SQLDR',

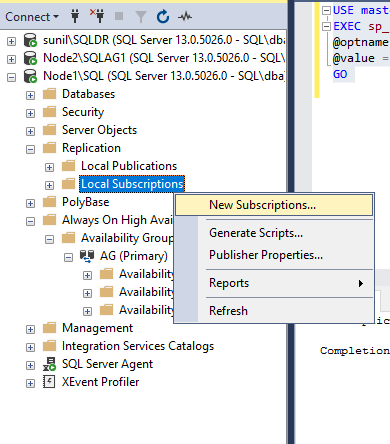
@password = 'Admin$123'

1. **Configure Subscriber:**

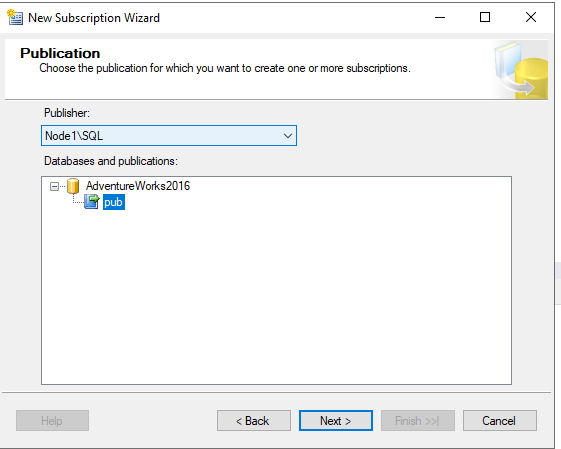
To add the subscriber, log in to the publisher **Node1** and

Navigate to **Replication** -> **Local Publications**.

Right-click on the publication you created above and click on **New Subscriptions**.

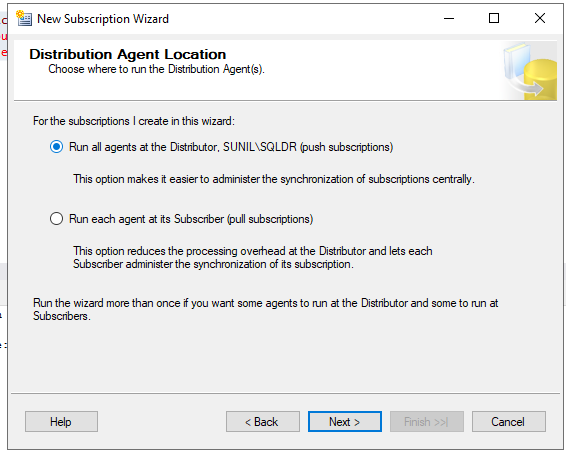


Click **Next.**Select the publication and click **Next**.

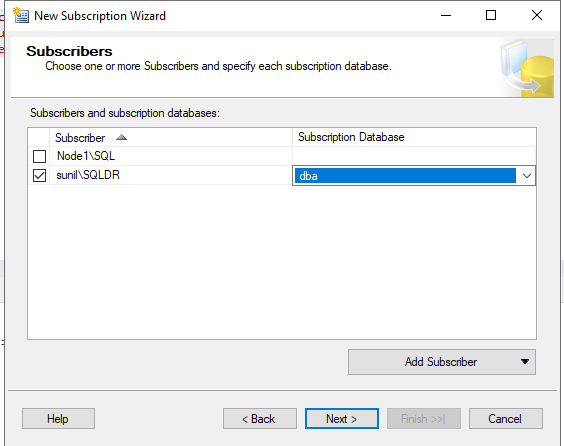


Select the Distribution Agent location and click **Next**.

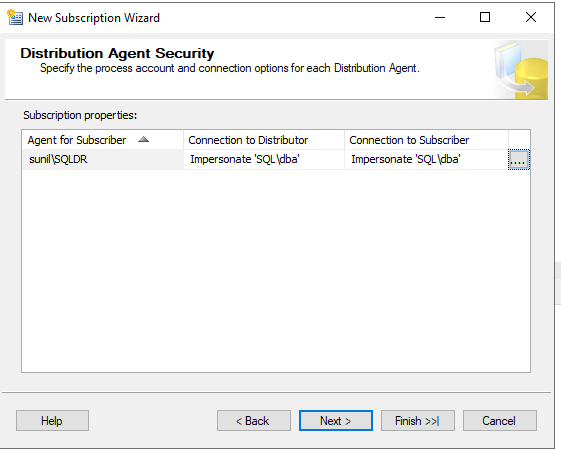
Choose subscriber type **Pull or Push subscriptions**



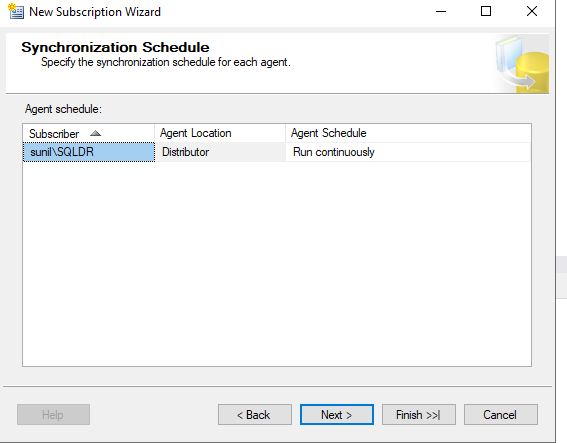
Click on Add SQL Server subscriber and add Node3 as a subscriber.



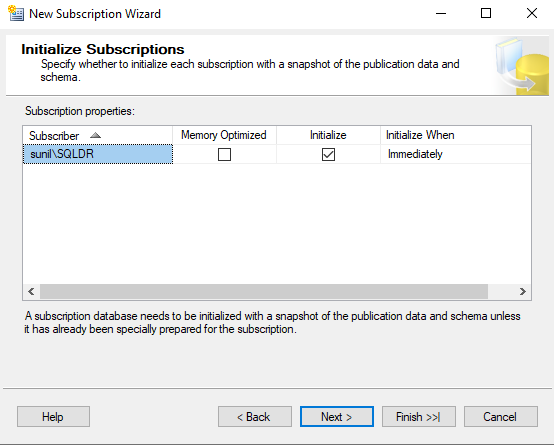
Configure agent security and click **Next.**



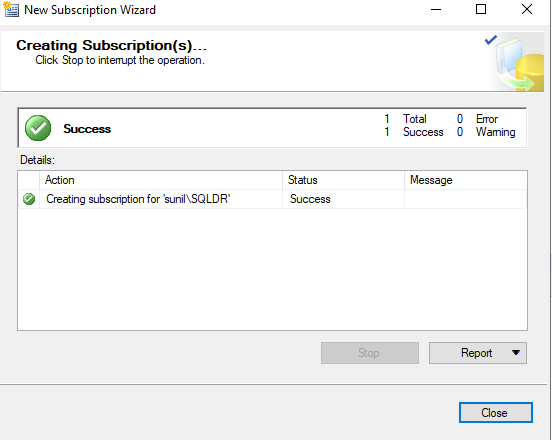
Schedule the SQL replication agent to run as per your choice.



Initialize the subscription **Immediately**to apply the snapshot immediately after creation. Select the subscription type as **Server** if you want to republish the subscriber else select the subscription type as **Client** and create the subscriber.



Once the subscriber is created the initial snapshot is applied and all the incremental data changes are replicated to the subscriber from the publisher and vice versa.



## Redirect the original publisher to the Always on Availability Group Listener:

Log in to the remote distribution server Node3 using SQL Server management studio.

Execute the following script on the distribution database to redirect the original publisher to Always on the listener.

--Run On distribtion server

USE distribution

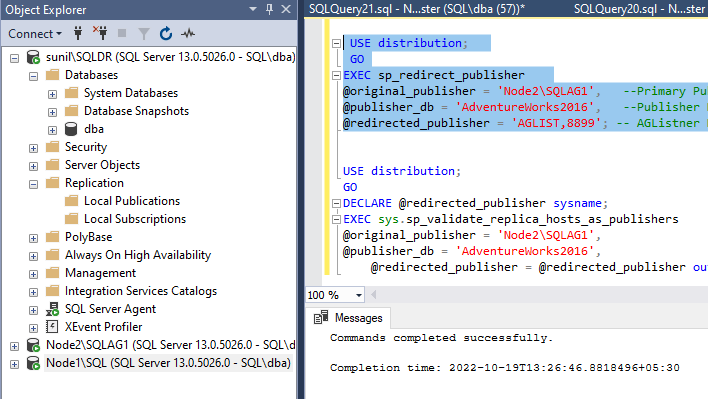
GO

EXEC sp\_redirect\_publisher

@original\_publisher = 'Node2\SQLAG1', --Primary Publisher Name

@publisher\_db = 'AdventureWorks2016', --Publisher Database name

@redirected\_publisher = 'AGLIST,8899'; --AGListner Name



**At the distribution database, Execute the following script to validate the redirection.**

USE distribution;

GO

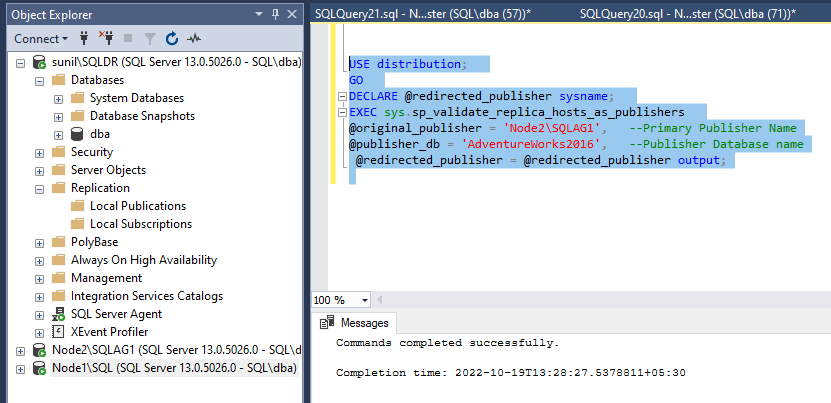
DECLARE @redirected\_publisher sysname;

EXEC sys.sp\_validate\_replica\_hosts\_as\_publishers

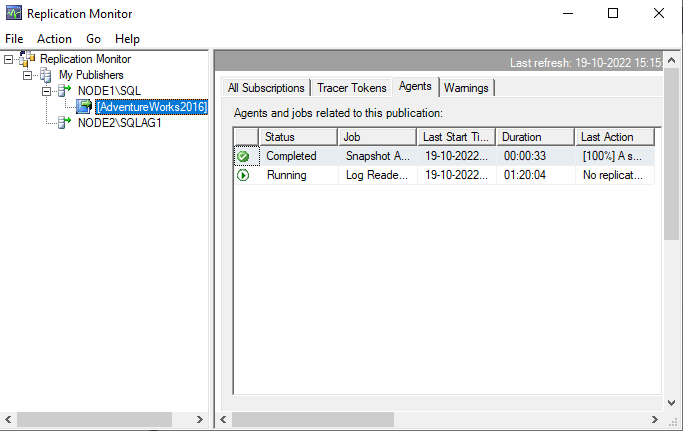
@original\_publisher = 'Node2\SQLAG1', --Primary Publisher Name

@publisher\_db = 'AdventureWorks2016', --Publisher Database name

@redirected\_publisher = @redirected\_publisher output;



Once the validation is completed, manually failover the primary node to the secondary node and test the replication agent synchronization and verify the data on the SQL replication publishers and the subscribers.



**Trace Flage 1448 enabled in SQL Server**

https://blogs.lessthandot.com/index.php/datamgmt/dbadmin/asynchronous-replicas-and-transactional-replication/