

郭大侠

博客园 首页 新随笔 联系 订阅 管理

公告

昵称： 郭大侠1
园龄： 5年10个月
粉丝： 238
关注： 74
[+加关注](#)

| | | | | | | | |
|----|----------|----|----|----|----|----|---|
| < | 2023年11月 | | | | | | > |
| 日 | 一 | 二 | 三 | 四 | 五 | 六 | |
| 29 | 30 | 31 | 1 | 2 | 3 | 4 | |
| 5 | 6 | 7 | 8 | 9 | 10 | 11 | |
| 12 | 13 | 14 | 15 | 16 | 17 | 18 | |
| 19 | 20 | 21 | 22 | 23 | 24 | 25 | |
| 26 | 27 | 28 | 29 | 30 | 1 | 2 | |
| 3 | 4 | 5 | 6 | 7 | 8 | 9 | |

搜索

找找看

常用链接

[我的随笔](#)
[我的评论](#)
[我的参与](#)
[最新评论](#)
[我的标签](#)

积分与排名

积分 - 1197477
排名 - 228

随笔分类

[bat\(17\)](#)
[ELK\(28\)](#)
[etcd\(1\)](#)
[influxdb\(1\)](#)
[kafka\(4\)](#)
[linux\(44\)](#)

【sql server复制】sql server复制【最佳实践】

目录

- [【0】环境信息](#)
 - [【0.1】拓扑架构环境](#)
 - [【0.2】实例名与服务器名检查](#)
- [【1】前置环境配置](#)
 - [【1.1】修改hosts（配置DNS）](#)
 - [【1.2】修改防火墙（网络连通性）](#)
 - [【1.3】建立复制账户，测试连通性](#)
 - [【1.4】构建快照共享文件夹](#)
- [【2】分发、发布、订阅配置](#)
 - [【2.1】配置分发服务器](#)
 - [【2.2】配置发布服务器](#)
 - [【2.3】配置订阅服务器](#)
- [【3】核验与监视](#)
 - [【3.1】数据库查看](#)
 - [【3.2】复制监视器](#)
 - [【3.3】系统存储过程监视](#)
 - [【3.4】表：查看复制表监视](#)
- [【4】优化](#)
 - [【优化概述】（必看）](#)
 - [【4.0】官网使用参考](#)
 - [【4.1】快照初始化优化](#)
 - [【4.2】修改数据同步过程优化（日志代理与分发代理）](#)
 - [【4.3】不重新初始化新增数据库对象](#)
 - [【4.4】使用T-SQL，不初始化 增删表](#)
 - [不初始化，删除表案例](#)
 - [【4.5】解决大事务引起的发布订阅效率问题](#)
- [【5】基本原理](#)

[linux准备工作\(37\)](#)

[memcache\(3\)](#)

[mongodb\(8\)](#)

[MSSQL dba管理\(106\)](#)

[MSSQL dba开发\(94\)](#)

[MSSQL常用代码\(18\)](#)

[mysql dba管理\(46\)](#)

[mysql dba开发\(28\)](#)

[mysql 故障处理\(41\)](#)

[更多](#)

随笔档案

[2023年11月\(1\)](#)

[2023年10月\(5\)](#)

[2023年9月\(2\)](#)

[2023年8月\(4\)](#)

[2023年7月\(5\)](#)

[2023年6月\(11\)](#)

[2023年5月\(6\)](#)

[2023年4月\(7\)](#)

[2023年3月\(5\)](#)

[2023年2月\(2\)](#)

[2023年1月\(2\)](#)

[2022年12月\(5\)](#)

[2022年11月\(10\)](#)

[2022年10月\(5\)](#)

[2022年9月\(7\)](#)

[更多](#)

相册

[打赏\(4\)](#)

好友链接

[柠檬](#)

学习路径链接

[sqlserver_舒永春](#)

[mysql_知数堂](#)

阅读排行榜

1. [mysql字符串拆分实现split功能\(203888\)](#)

2. [Navicat破解版下载安装\(35358\)](#)

3. [mysql备份的4种方式\(33197\)](#)

- [【5.0】事务复制的原理](#)
- [【5.1】发布订阅的角色与概念](#)
- [【5.2】复制类型与订阅类型](#)
- [【5.3】复制代理](#)
- [【5.4】架构更改的注意事项](#)

• [【6】业务场景](#)

- [【6.1】跨机房、跨网段复制](#)
- [【6.2】跨机房、跨网段FTP复制](#)
- [【6.3】通过备份文件初始化复制](#)

• [【注意事项与疑问】](#)

- [\(1\) 注意事项](#)
- [\(2\) 疑问](#)

• [【常见问题】](#)

• [【故障排查】](#)

- [\(0\) 参考文档:](#)
- [\(1\) 由于出现操作系统错误 3, 进程无法读取文件](#)
- [\(2\) 订阅不存在](#)
- [\(3\) 找不到存储的过程 sp_MSins_tablename \(误删订阅服务器上的订阅\)](#)
- [\(4\) 无法禁用服务器 xxxx 上的发布与分发](#)
- [\(5\) 服务器名和主机名不一致](#)
- [\(6\) sql server复制需要实际的服务器名称才能连接到服务器](#)
- [\(7\) 强行删除订阅发布](#)
- [\(8\) 无法作为数据库主题执行, 因为主体dbo不存在](#)

• [【参考文档】](#)

- [1、概念与搭建](#)
- [2、优化与使用](#)
- [3、常见故障与问题](#)
- [4、深入解析与优化](#)

• [...](#)

auth: chaoqun.guo create by : 2020

官网: <https://docs.microsoft.com/zh-cn/sql/relational-databases/replication/sql-server-replication?view=sql-server-ver15>

[回到顶部](#)

【0】环境信息

【0.1】拓扑架构环境

- 4. mysql窗口函数、Mysql分析函数(32348)
- 5. 如何查看windows某个目录下所有文件/文件夹的大小? (TreeSize Free)(28527)

评论排行榜

- 1. 上传markdown文件到博客园(43)
- 2. Navicat破解版下载安装(16)
- 3. 博客园如何加目录(7)
- 4. 深入SQL Server优化【推荐】(5)
- 5. notepad++安装SQL格式化插件(4)

推荐排行榜

- 1. mysql字符串拆分实现split功能(17)
- 2. 上传markdown文件到博客园(12)
- 3. Navicat破解版下载安装(10)
- 4. 深入SQL Server优化【推荐】(10)
- 5. 深入Alertmanager 概念与配置介绍(5)

最新评论

- 1. Re:上传markdown文件到博客园
为啥到第二步我就改不了config.yaml,改完之后再打开还是原来的xxx
--be_like
- 2. Re:【sql server复制】sql server复制【最佳实践】
@MicroAnts 可以注明出处即可...
--郭大侠1
- 3. Re:【sql server复制】sql server复制【最佳实践】
你好可以转载吗?
--MicroAnts
- 4. Re:windows server 如要远程登录, 你需要具有通过远程桌面服务进行登录的权限。
添加用户或组的选项是灰色的怎么搞
--adawdq1e
- 5. Re:如何安装.net framework3.5
成功完成, 感谢博主分享!
--天成1687

| | |
|-----------|--|
| 角色 | 使用值 |
| 系统环境 | Winserver2016+sqlserver2016 |
| 发布服务器 | 192.168.175.145, 服务器名称: WIN-ST1AC985Q23 |
| 分发服务器 | 192.168.175.145, 服务器名称: WIN-ST1AC985Q23 |
| 订阅服务器 | 192.168.175.146, 服务器名称: WIN-UQH LGQ8DI5E |
| 发布数据库repl | 订阅数据库: repl |
| 数据库账户 | Repl/a123456! 权限: sysadmin |
| 本文针对复制类型 | 快照复制, 事务日志复制 |

【0.2】实例名与服务器名检查

检查sql server数据库实例名与服务器名是否一样



```
IF SERVERPROPERTY ('SERVERNAME') <>@@SERVERNAME

Print '服务器名和数据库实例名不同! 请修改操作后重启! '

--IF SERVERPROPERTY ('SERVERNAME') <>@@SERVERNAME
--BEGIN
-- DECLARE @server SYSNAME
-- SET @server=@@SERVERNAME
-- EXEC sp_dropserver @server=@server
-- SET @server=CAST (SERVERPROPERTY ('SERVERNAME') AS SYSNAME)
-- EXEC sp_addserver @server=@server,@local='LOCAL'
--END
```



如果不一样, 请把注释放开, 让后运行, 之后重启服务器

【1】前置环境配置

【1.1】修改hosts（配置DNS）

所有订阅发布机器都要配置： C:\Windows\System32\drivers\etc\hosts。也可以使用数据库客户端别名，这里就不介绍别名了，因为没有这个方便。

```
hosts - 记事本
文件(F)  编辑(E)  格式(O)  查看(V)  帮助(H)
# Copyright (c) 1993-2009 Microsoft Corp.
#
# This is a sample HOSTS file used by Microsoft TCP/IP for Windows.
#
# This file contains the mappings of IP addresses to host names. Each
# entry should be kept on an individual line. The IP address should
# be placed in the first column followed by the corresponding host name
# The IP address and the host name should be separated by at least one
# space.
#
# Additionally, comments (such as these) may be inserted on individual
# lines or following the machine name denoted by a '#' symbol.
#
# For example:
#
#       102.54.94.97       rhino.acme.com          # source server
#       38.25.63.10       x.acme.com              # x client host
#
# localhost name resolution is handled within DNS itself.
#       127.0.0.1         localhost
#       ::1               localhost
192.168.175.145 WIN-ST1AC985Q23
192.168.175.146 WIN-UQHLGQ8DI5E
```

【1.2】修改防火墙（网络连通性）

用命令配置吧，比较快，手动用界面配置防火墙入站规则，给对方IP开通数据库实例端口就好了，我们这里的实例端口是默认的1433；

打开cmd=>，然后运行下面的命令（这里为了省事我直接给局域网所有机器都开通了，要指定IP的话直接修改下面的IP地址即可）

```
netsh advfirewall firewall add rule name="mssql" dir=in action=allow remoteip="192.168.175.0/24" protocol=TCP localport="1433"
```

然后可以互相telnet 对方ip 1433，看看是否能访问对方1433端口，没问题就证明端口通了，连通测试这一步放到【1.3】下一步测试也可以。

【1.3】建立复制账户，测试连通性

所有发布订阅机器都建立该登录名和密码；给予了系统管理员权限；

```
USE [master]
```

```
GO
```

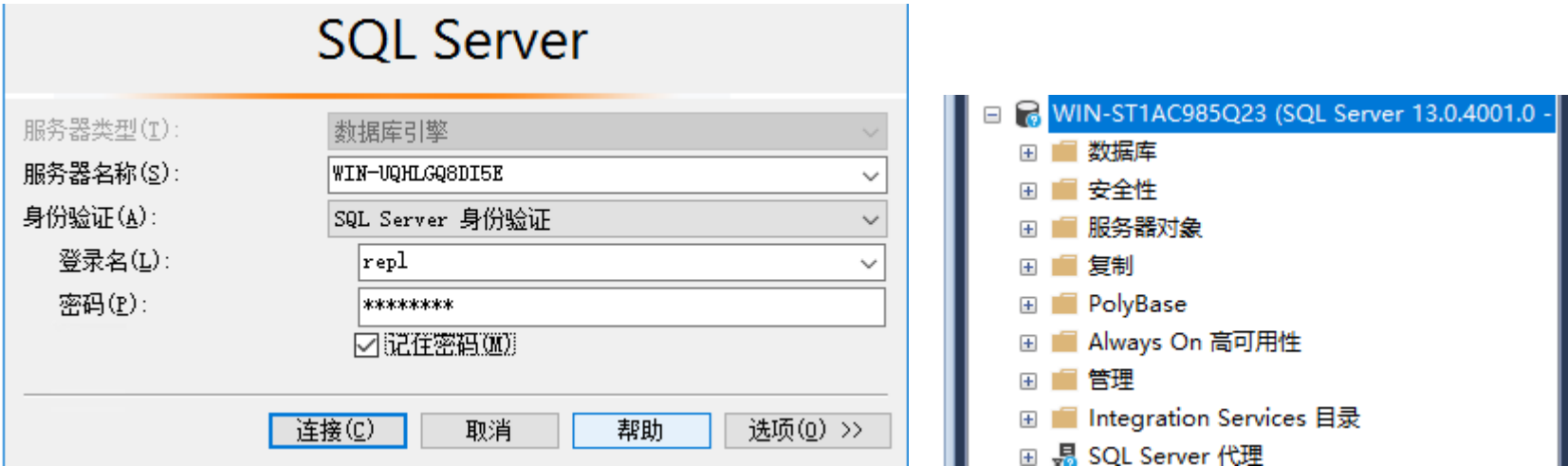
```
CREATE LOGIN [rep1] WITH PASSWORD=N'a123456!'
```

GO

ALTER SERVER ROLE [sysadmin] ADD MEMBER [repl]

GO

在两台机器上互相连接对方主机名：如下，可以连通就证明网络已经搞定了，账户名密码也是确认无误可以使用了。



【1.4】构建快照共享文件夹

(1) 修改发布端（192.168.175.145）agent启动账户为 localsystem，这个账户拥有本地系统访问权限。

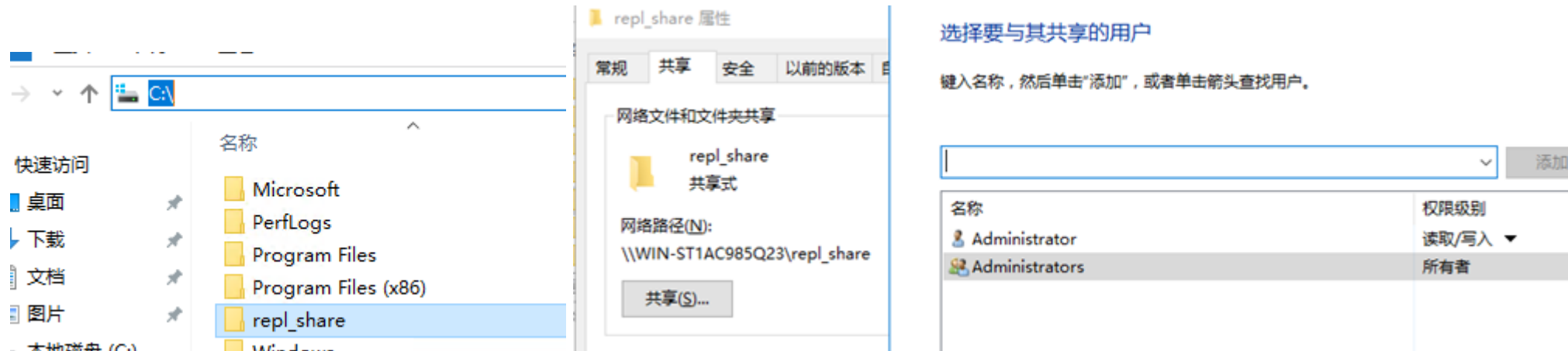
)

| 名称 | 状态 | 启动模式 | 登录身份为 | 进程 ID | 服务类型 |
|-----------------------|------|----------------|---------------------|-------|-----------------|
| SQL Server Inte... | 已停止 | 其他("引导"、"系统... | NT Service\MsDts... | 0 | |
| SQL Full-text Filt... | 正在运行 | 手动 | NT Service\MSSQ... | 3600 | |
| SQL Server (MS... | 正在运行 | 自动 | NT Service\MSSQ... | 2480 | SQL Server |
| SQL Server Ana... | 已停止 | 其他("引导"、"系统... | NT Service\MSSQ... | 0 | Analysis Server |
| SQL Server Rep... | 已停止 | 其他("引导"、"系统... | NT Service\Repor... | 0 | Report Server |
| SQL Server Bro... | 已停止 | 其他("引导"、"系统... | NT AUTHORITY\L... | 0 | |
| SQL Server 代理... | 正在运行 | 自动 | LocalSystem | 1240 | SQL Agent |

否则需要手动给共享目录添加NT Service\SQLSERVERAGENT 的读写权限（也不一定是这个账户，看这里显示的是什么）

(2) 新建共享文件夹

自己新建一个共享文件夹，用于存放发布快照；然后把共享开启就OK了。



如果想要使用推送订阅，那么发布、订阅服务器这里的登录身份均自己新建一个windows账户，且该账户可以访问unc路径。（是在不行就给把该账户加入管理员组）

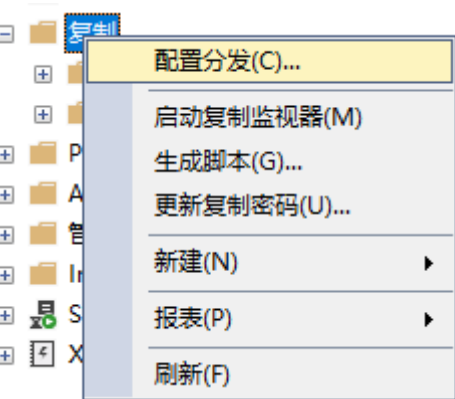
然后把这个文件夹的共享权限给予这个账户；

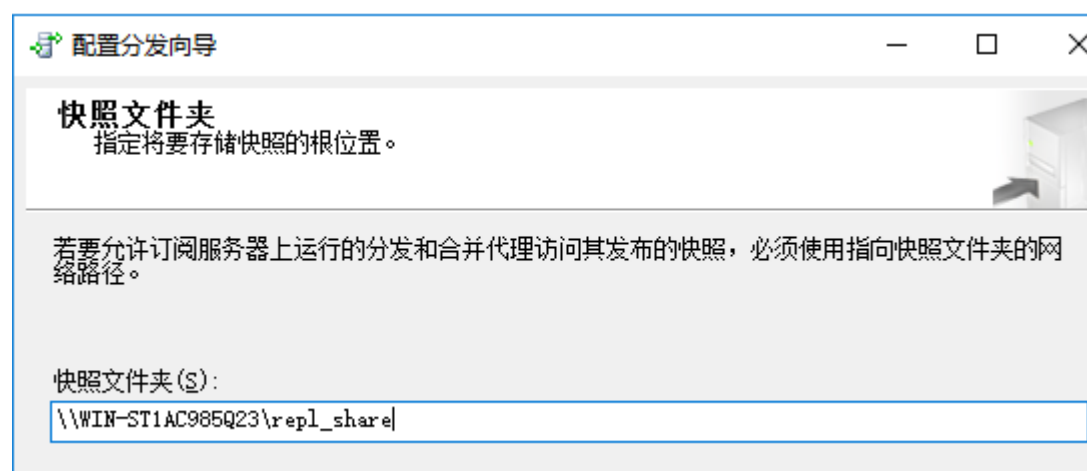
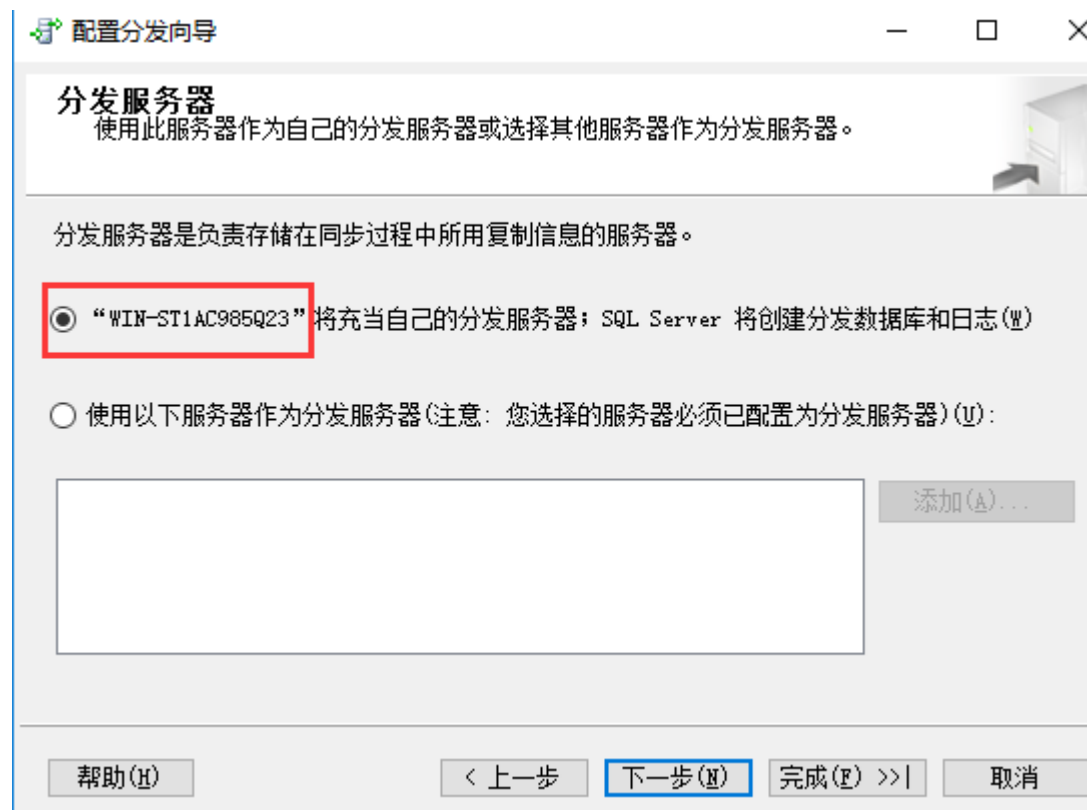
[回到顶部](#)

【2】分发、发布、订阅配置

【2.1】配置分发服务器

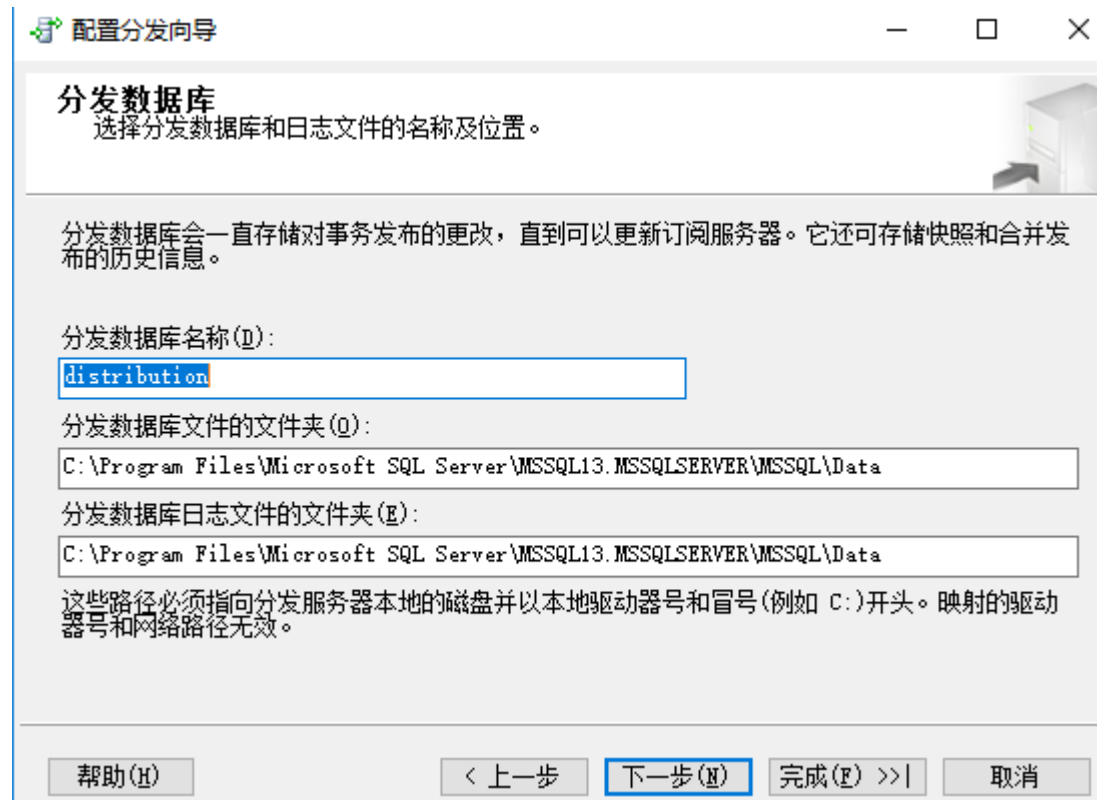
我们在主库，192.168.175.145 配置分发服务器；





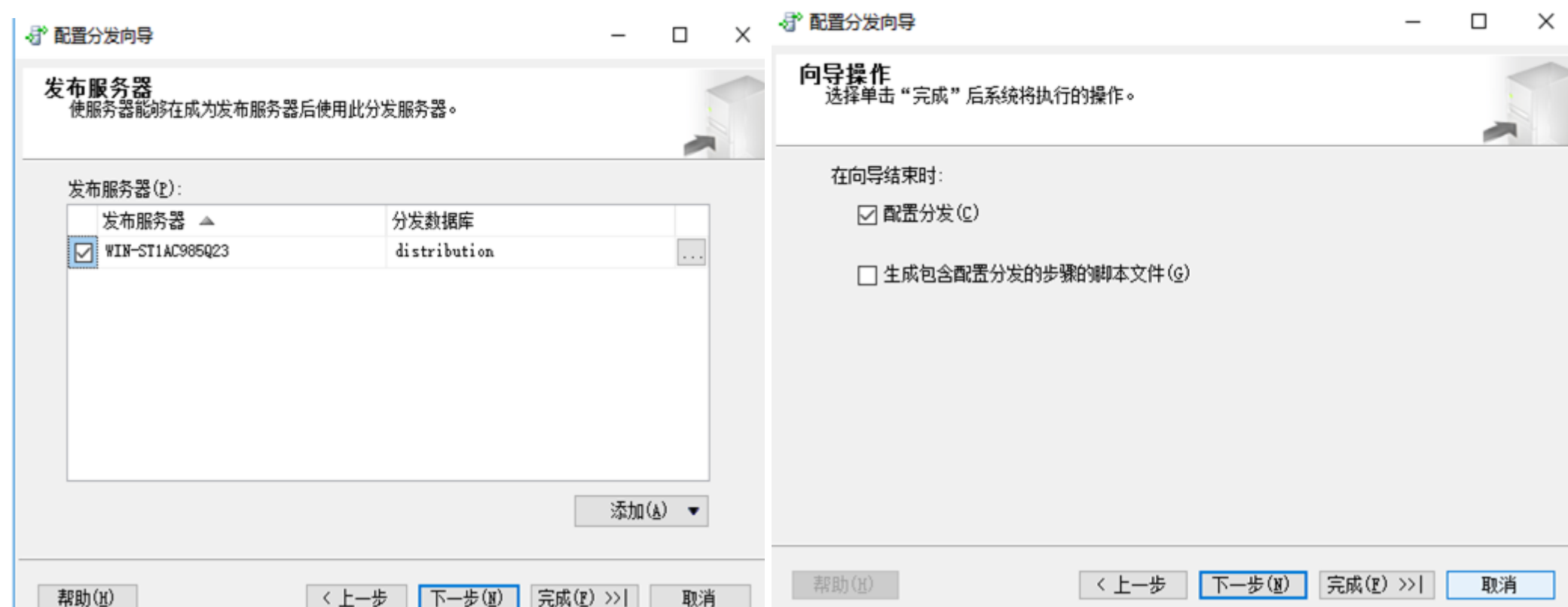
如果你设置快照文件夹路径为: C:\repl_share, 即使你的发布服务器本身就是分发服务器, 如果订阅服务器是另外一台机器, 那么在请求 (Pull) 订阅 (如果是推送 (Push) 订阅就没有这个限制) 模式下订阅代理是无法访问到这个快照文件的; 除非你发布服务器、分发服务器和订阅服务器都是同一台机器; 你应该设置快照文件夹路径为:

主机名\共享文件夹, 如上图; 然后继续下一步;

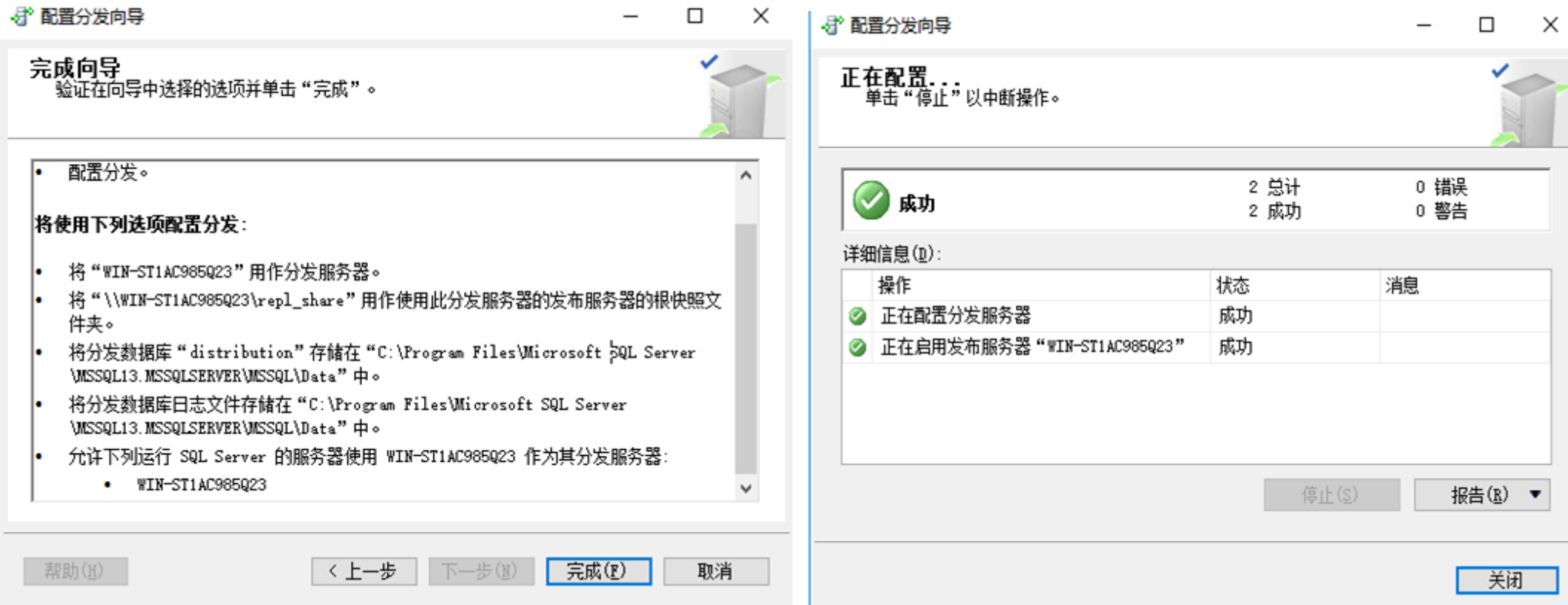


这里是配置分发数据库的名称和文件存放位置，这里我用的虚拟机，我就随便放了；

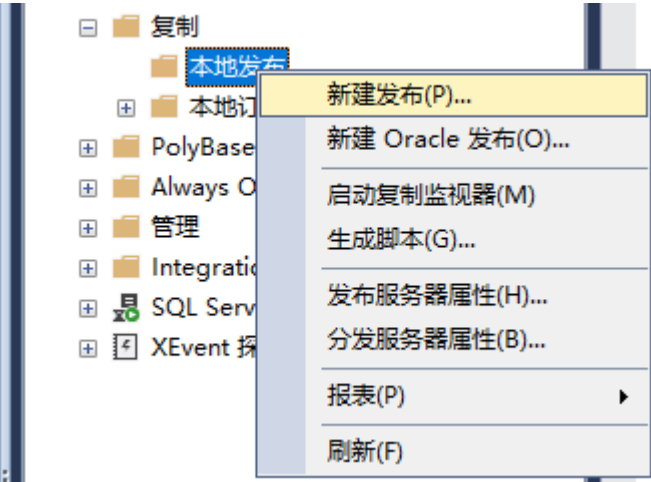
但请注意，务必要给这个文件夹添加 **sql server agent**启动账户读取权限



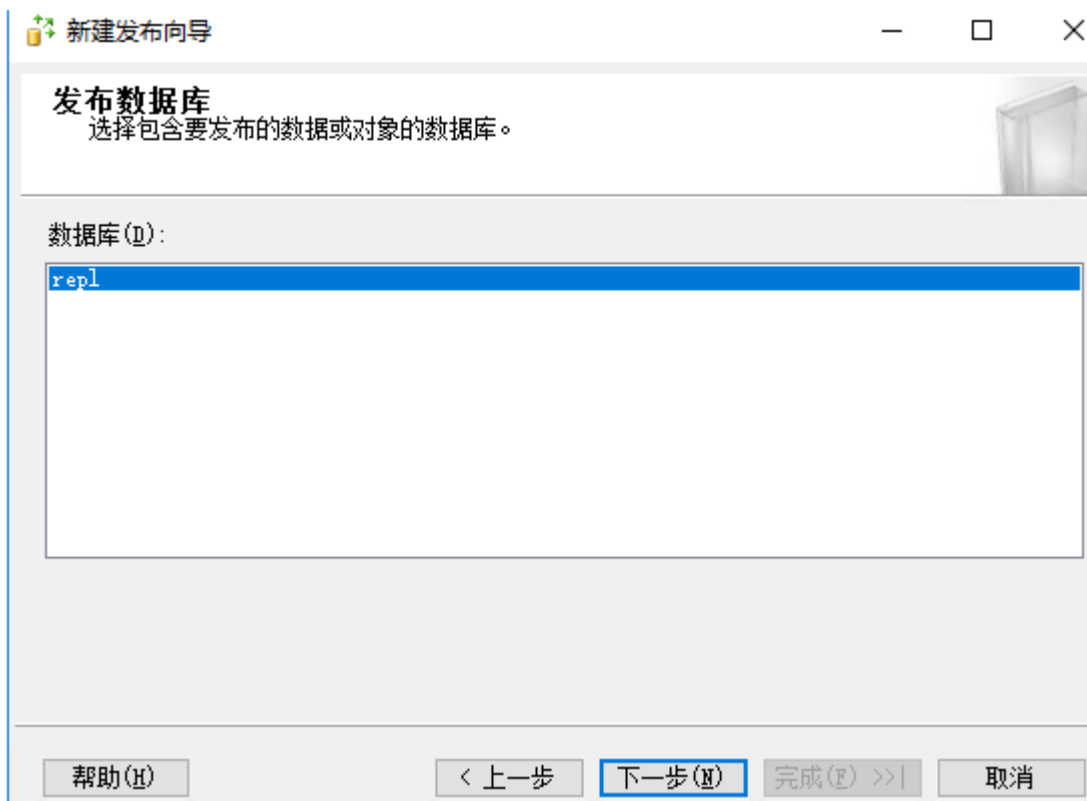
好，如上图，这就配置完成了；继续点下一步，如下图就会出现所有的配置信息。可以查阅一下，然后点击完成，成功。



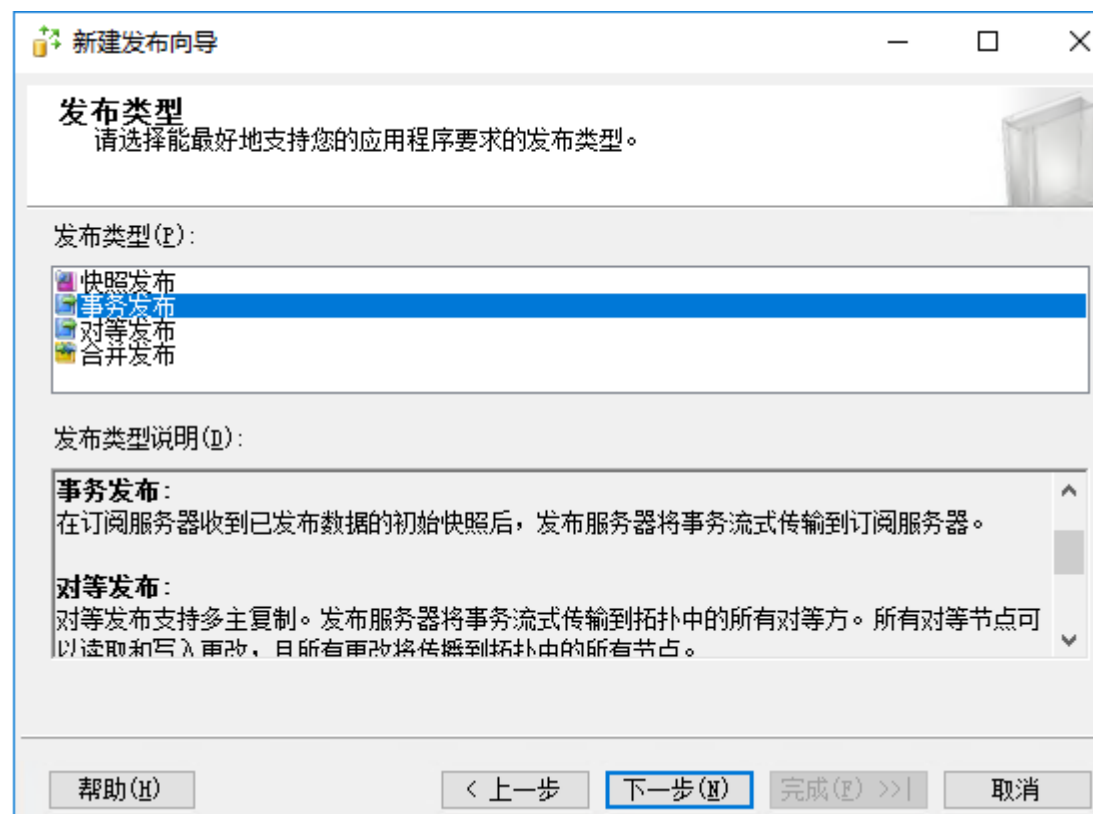
【2.2】配置发布服务器



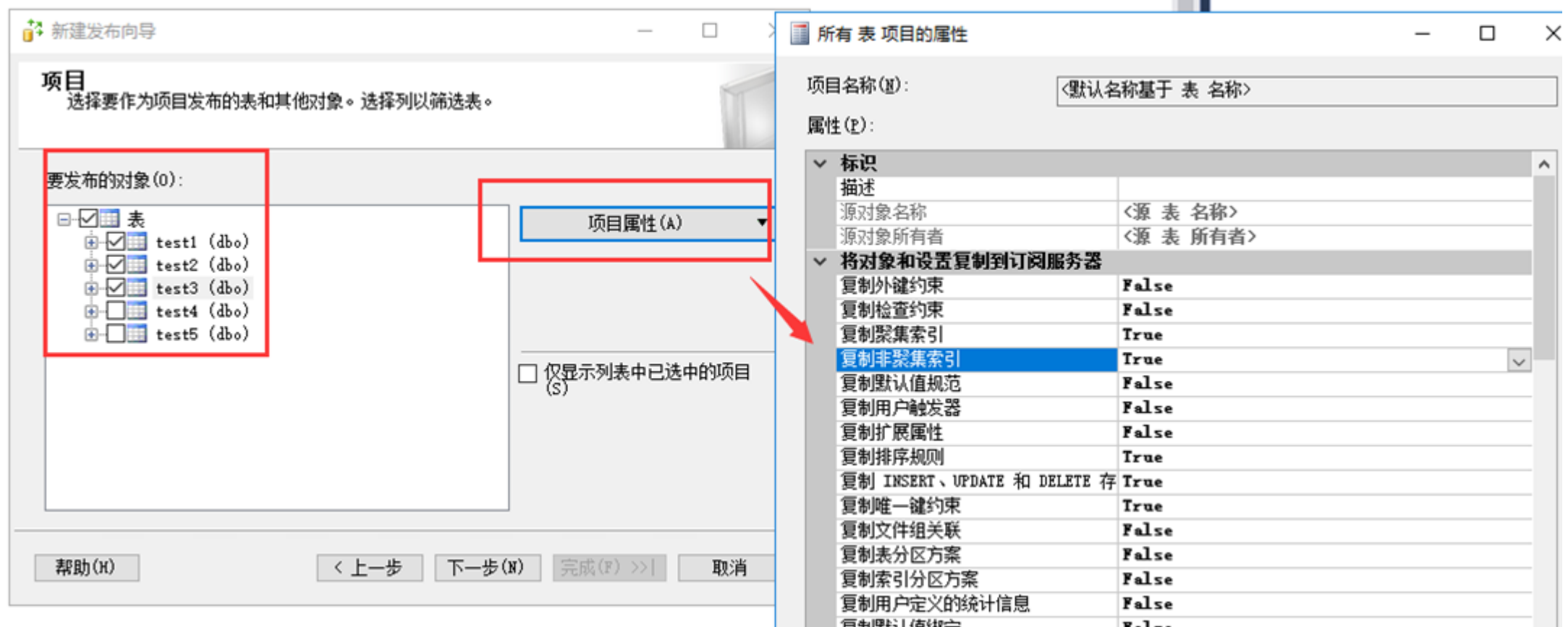
选择发布数据库



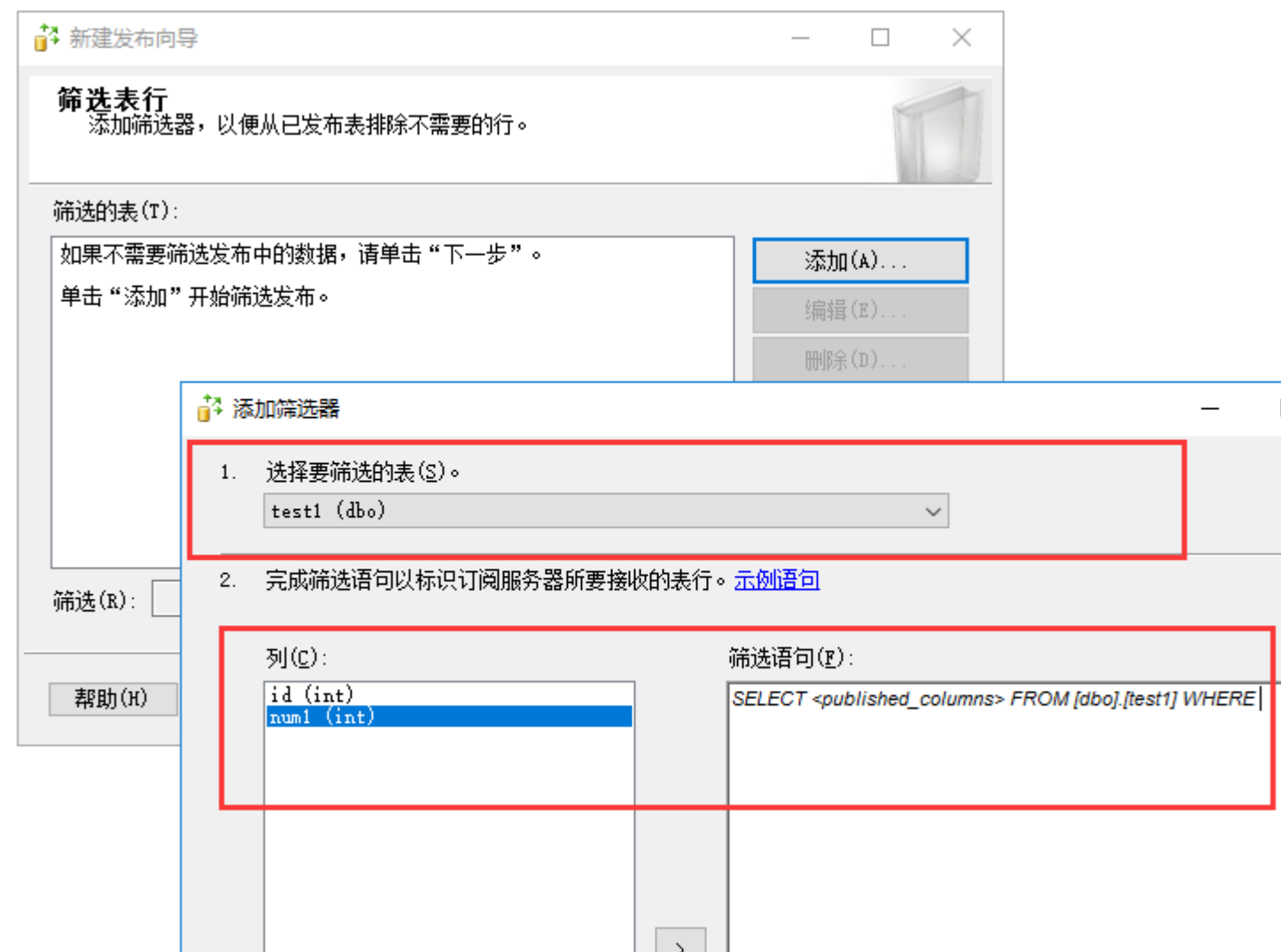
选择发布类型



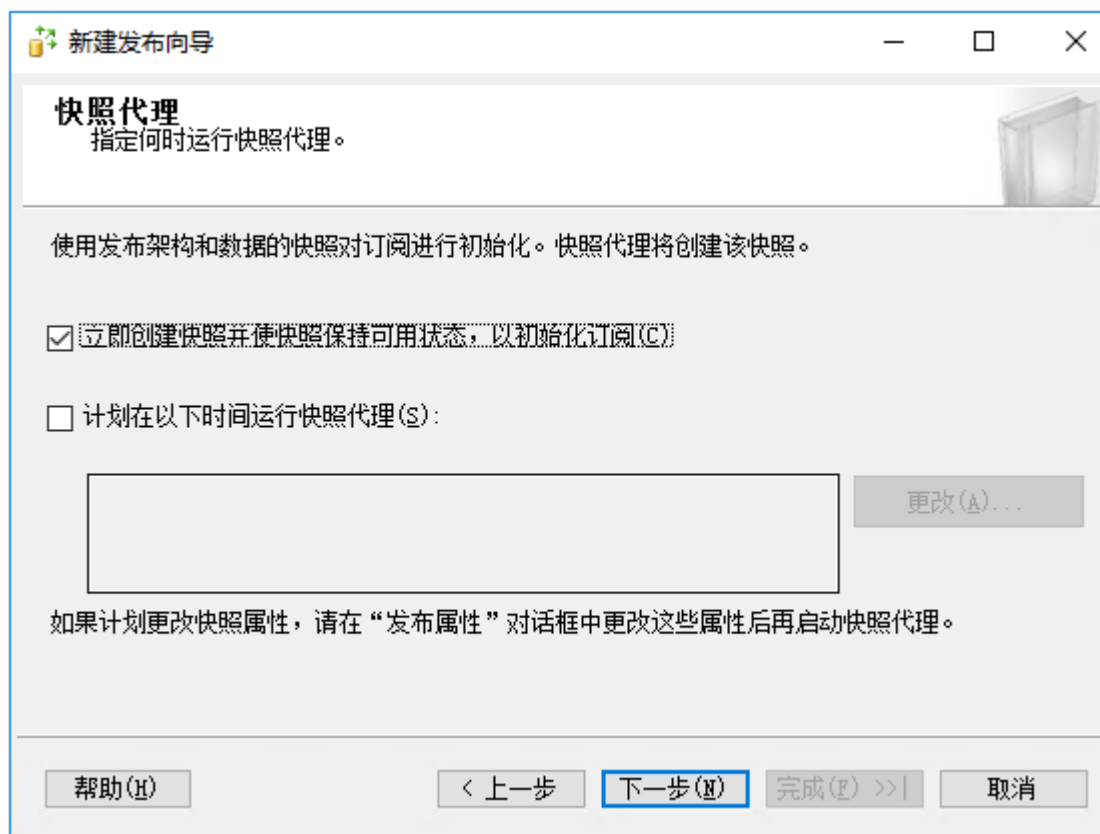
选择发布数据库对象，以及发布项目属性，这里我们把非聚集索引也添加到了发布中去，如下图



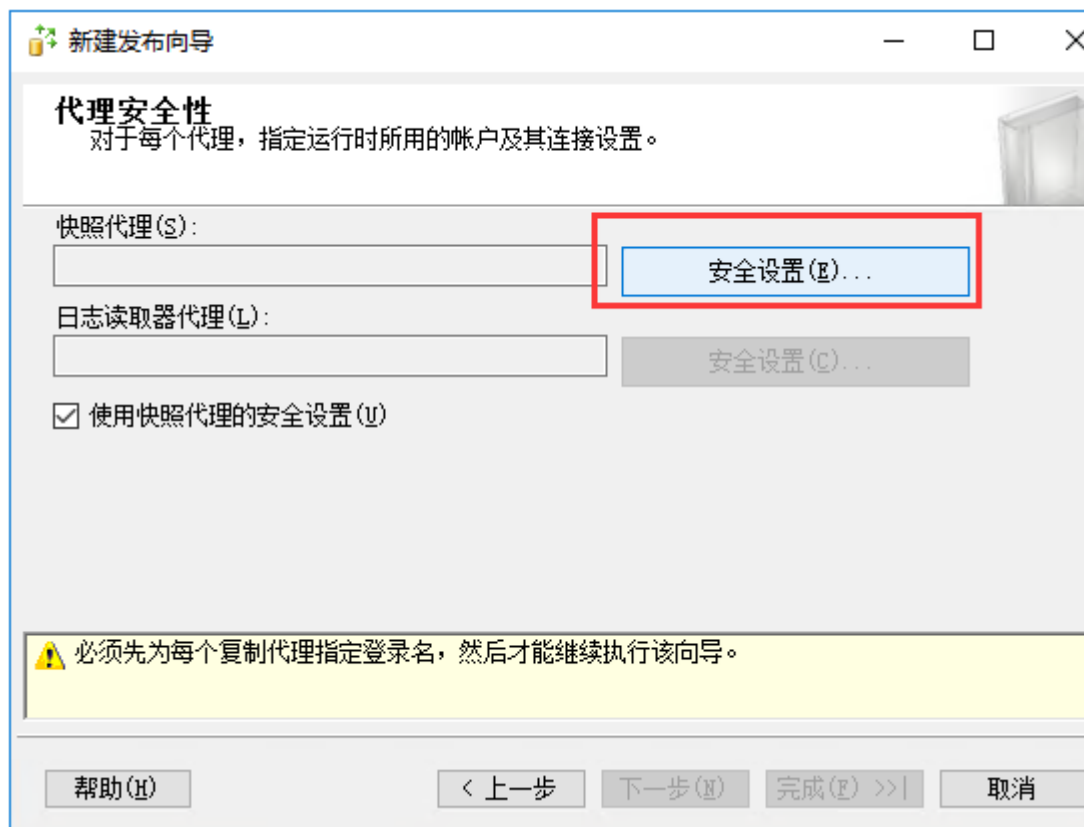
做数据筛选，这里我们没有做任何筛选，但其实是写sql针对某一部分数据筛选的，比如只同步id>500的数据.....等等



继续下一步，我们勾选上立即初始化快照



下一步，我们点击配置快照代理



快照代理安全性

指定将运行快照代理进程的域或计算机帐户。

☐ 在以下 Windows 帐户下运行 (R):

进程帐户 (A):

示例: 域\帐户

密码 (P):

确认密码 (C):

☒ 在 SQL Server 代理服务帐户下运行 (这不是我们推荐的最佳安全配置) (U)。

连接到发布服务器

☐ 通过模拟进程帐户 (E)

☒ 使用以下 SQL Server 登录名 (U):

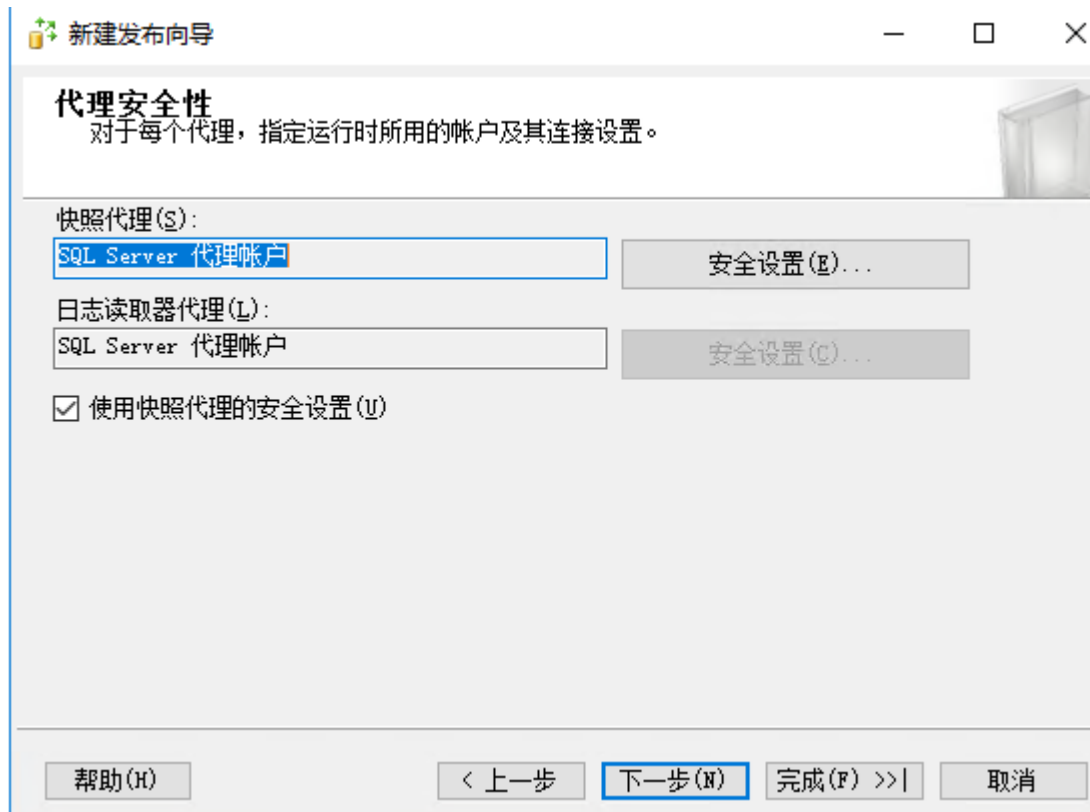
登录名 (L):

密码 (P):

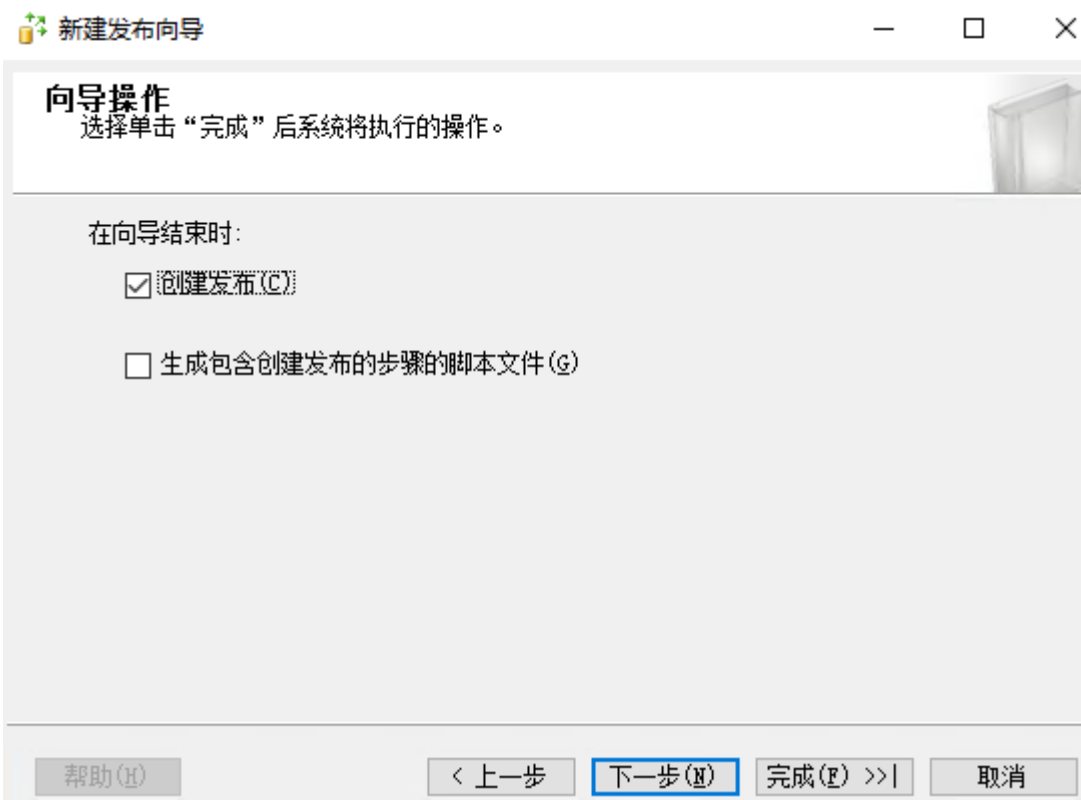
确认密码 (C):

确定 取消 帮助 (H)

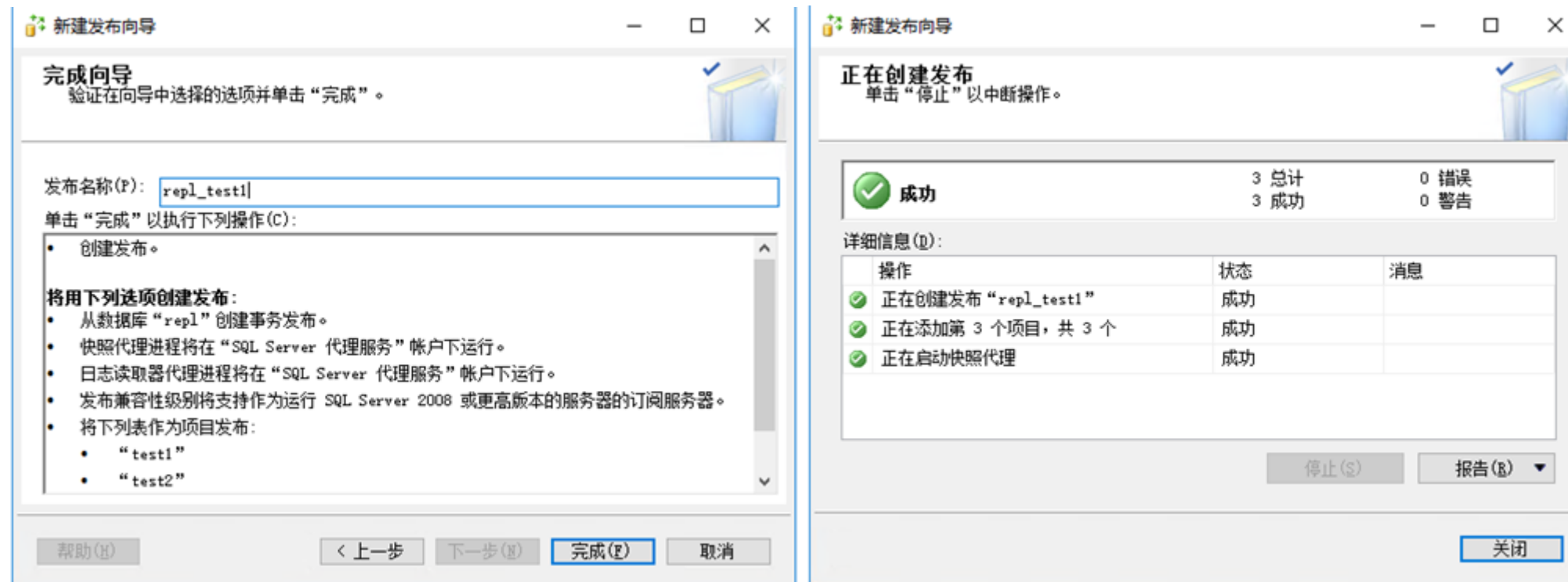
用我们之前配置好的数据库账户填充上图。配置完后如下图，勾选上使用快照代理的安全设置，然后下一步



然后勾选创建发布，点击下一步

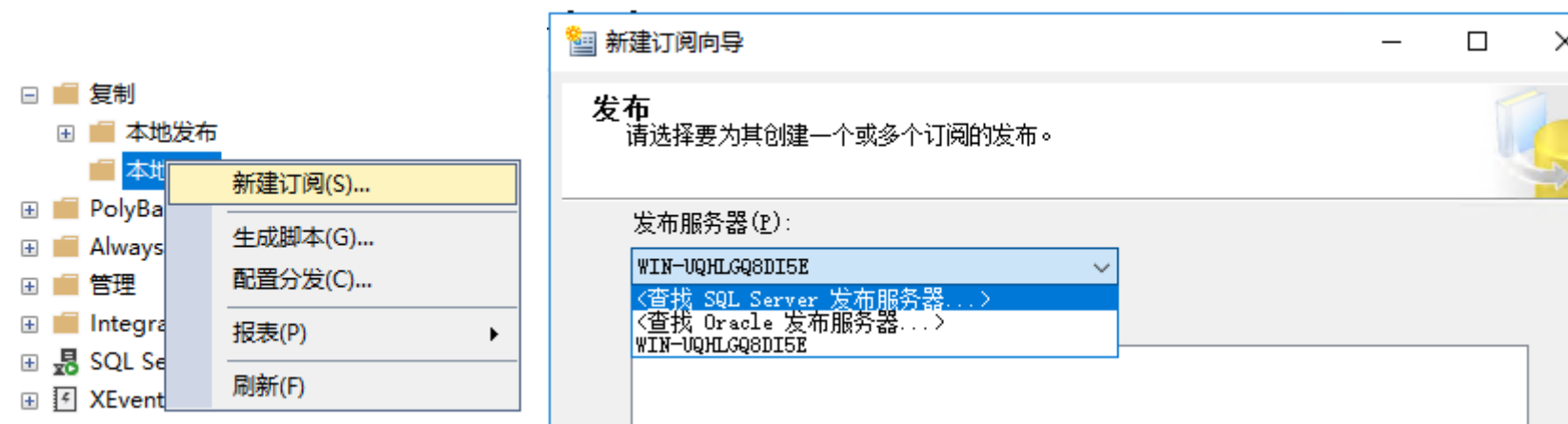


给发布取个名字，比如我这里就是repl_test1，然后点击完成；这就配置好了数据库发布；

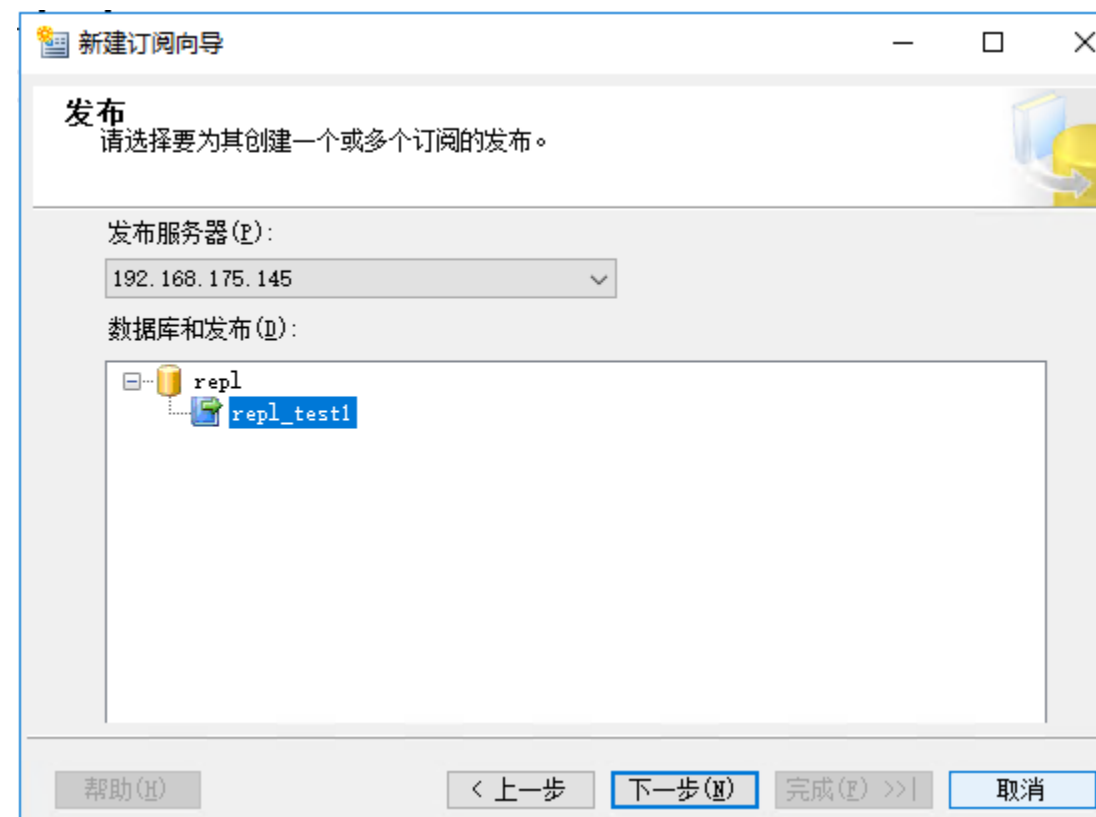


【2.3】配置订阅服务器

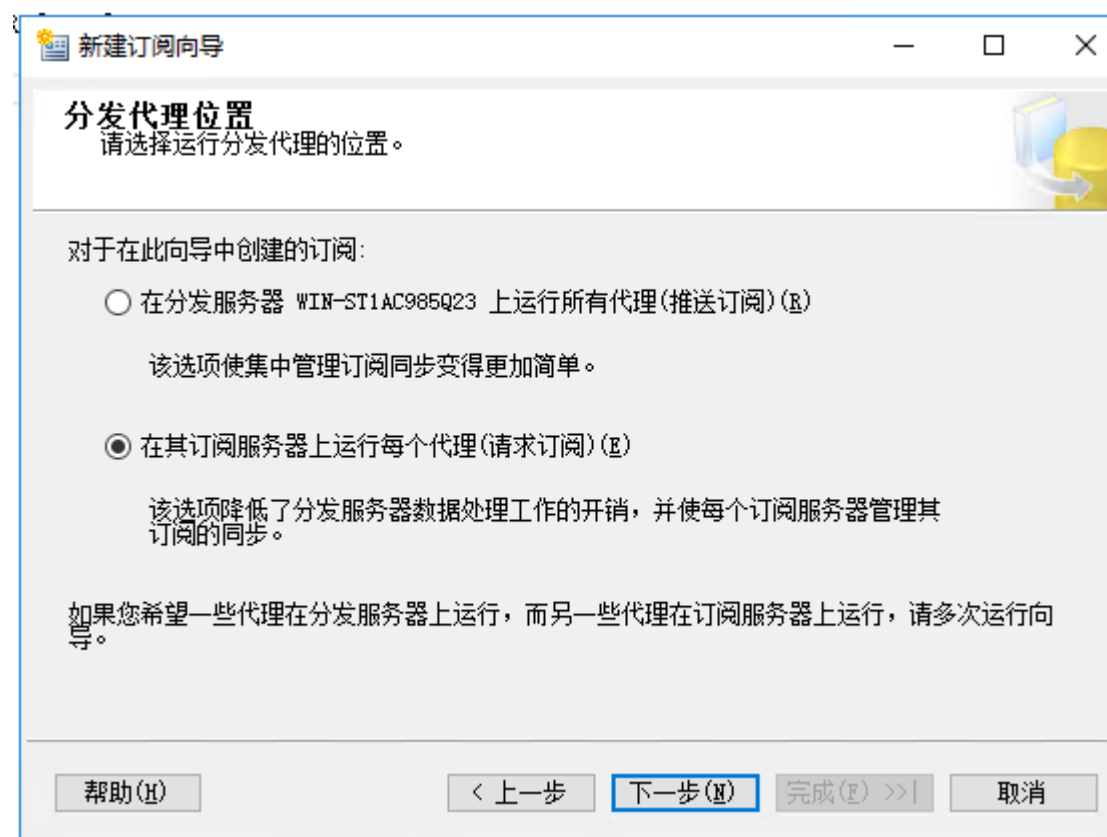
然后我们跑到订阅服务器上操作，即192.168.175.146，忘记了的看最前面的拓扑架构表信息



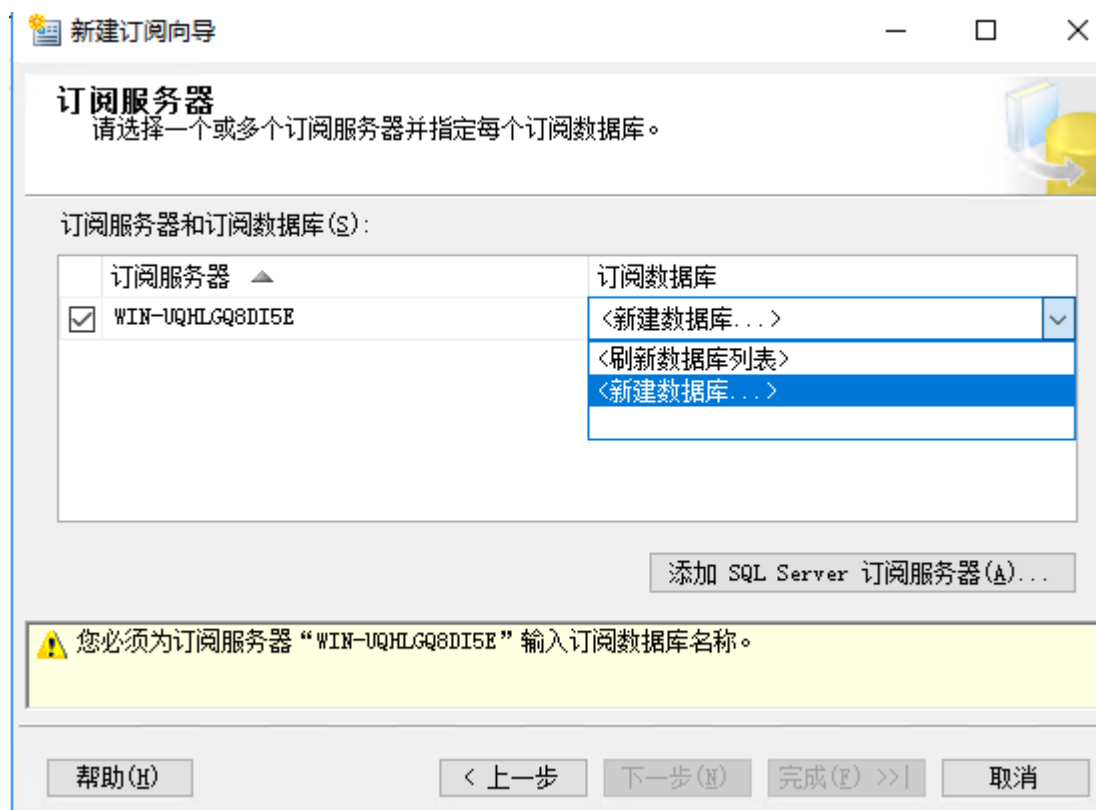
然后我们输入IP，遭到我们的发布服务器

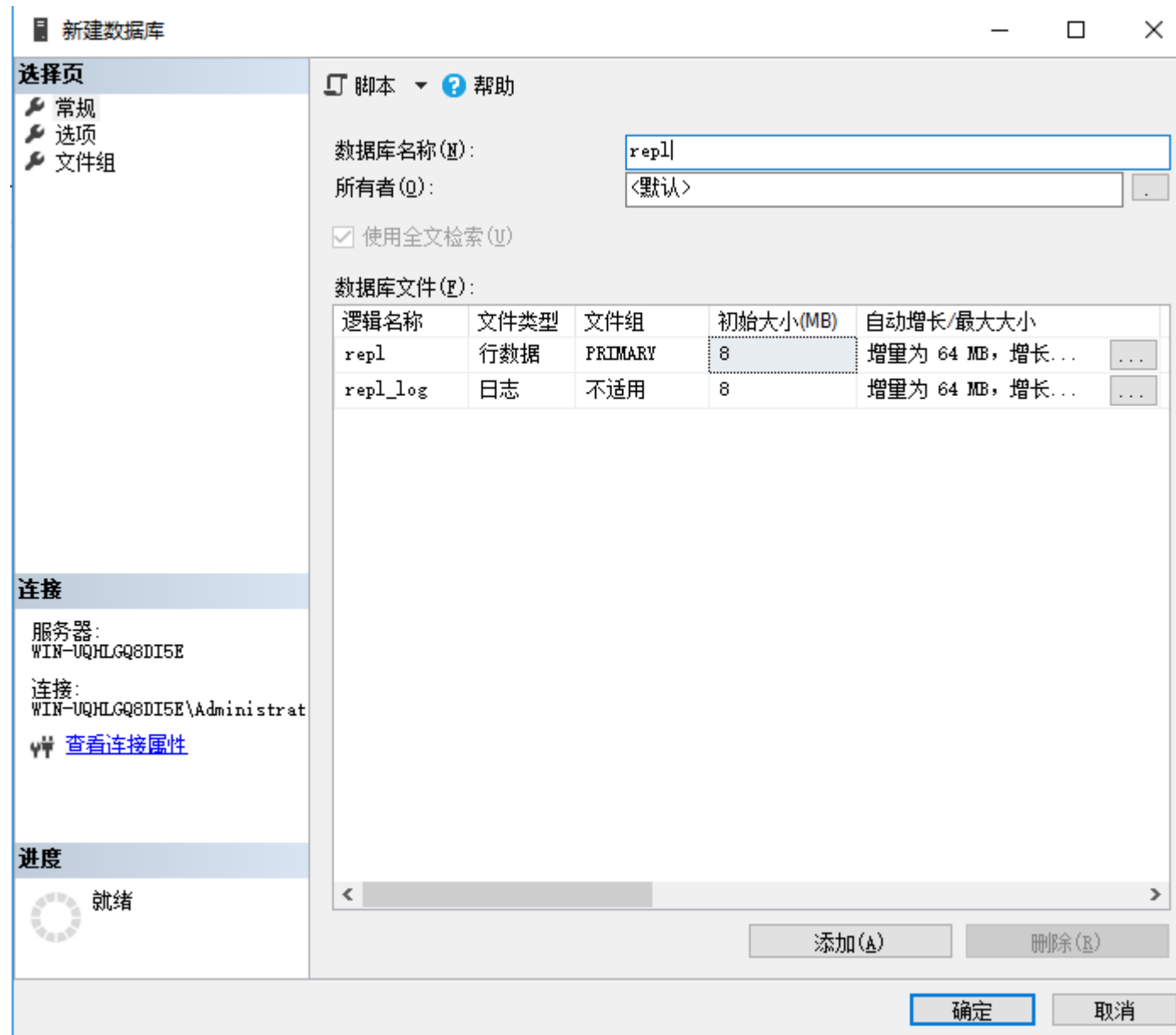


然后我们就下一步，选择请求订阅

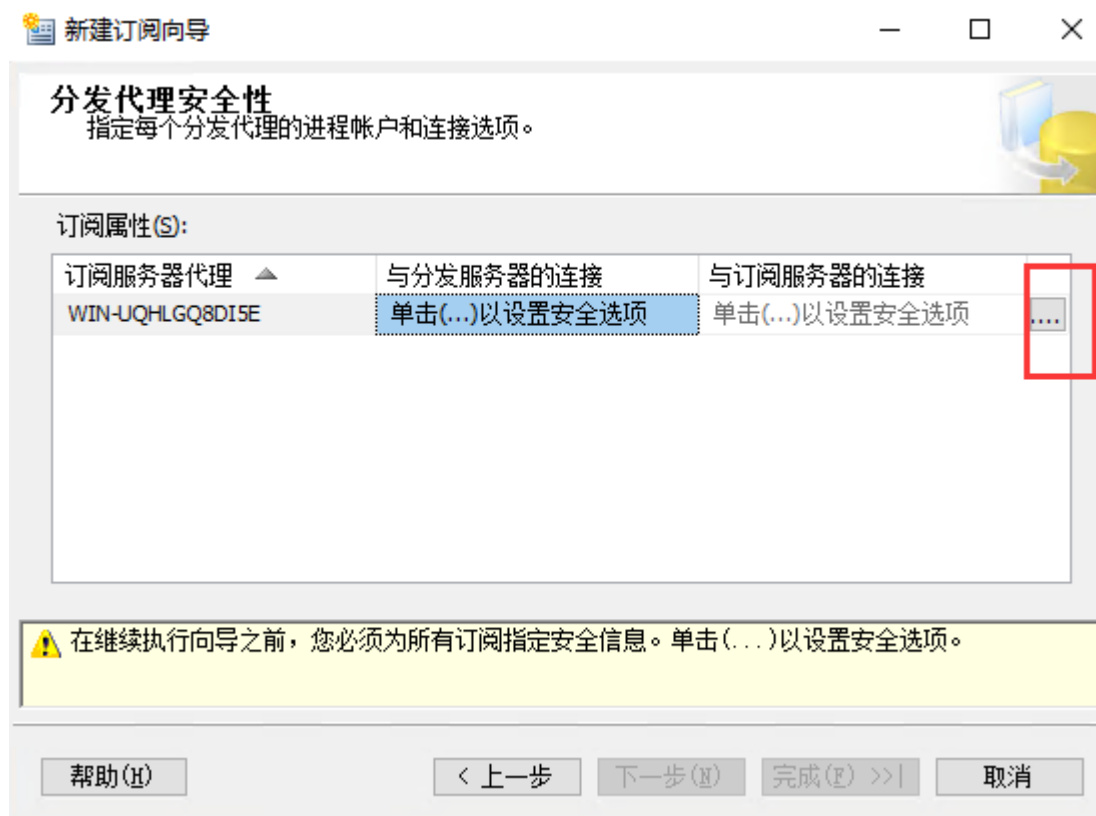


然后继续下一步，我们现在没有的话，我们可以新建订阅数据库





然后确认，然后下一步，点击右边的点



然后进行配置：

分发代理安全性

指定在同步此订阅时运行分发代理进程的域或计算机帐户。

☐

在以下 Windows 帐户下运行(B):

进程帐户(A):

示例: 域\帐户

密码(P):

确认密码(C):

☒

在 SQL Server 代理服务帐户下运行 (这不是我们推荐的最佳安全配置)(D)。

连接到分发服务器

☐ 通过模拟进程帐户(M)

☒ 使用以下 SQL Server 登录名(S):

登录名(L):

repl

密码(W):

确认密码(E):

用于连接发布服务器的登录名必须是发布访问列表的成员。

连接到订阅服务器

☒ 通过模拟进程帐户(B)

☐ 使用 SQL Server 登录名(U)

与运行代理的服务器的连接必须模拟进程帐户。进程帐户必须是订阅数据库的数据库所有者。

确定

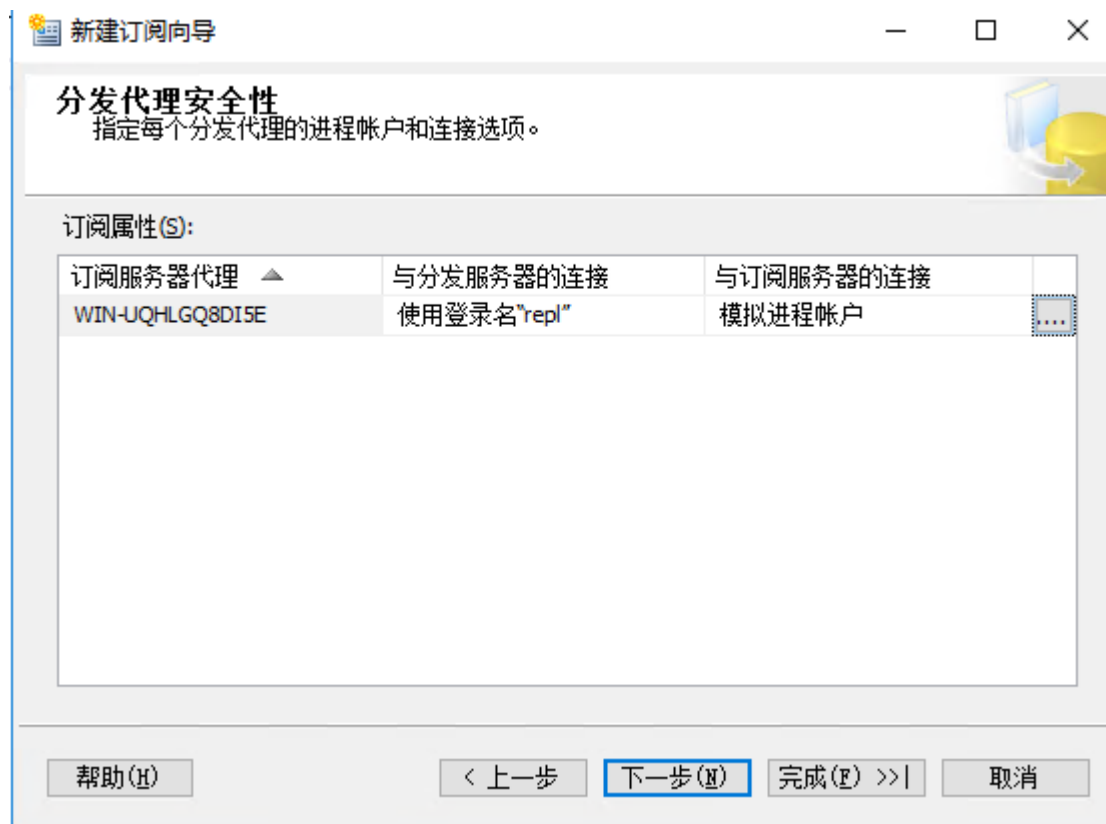
取消

帮助(H)

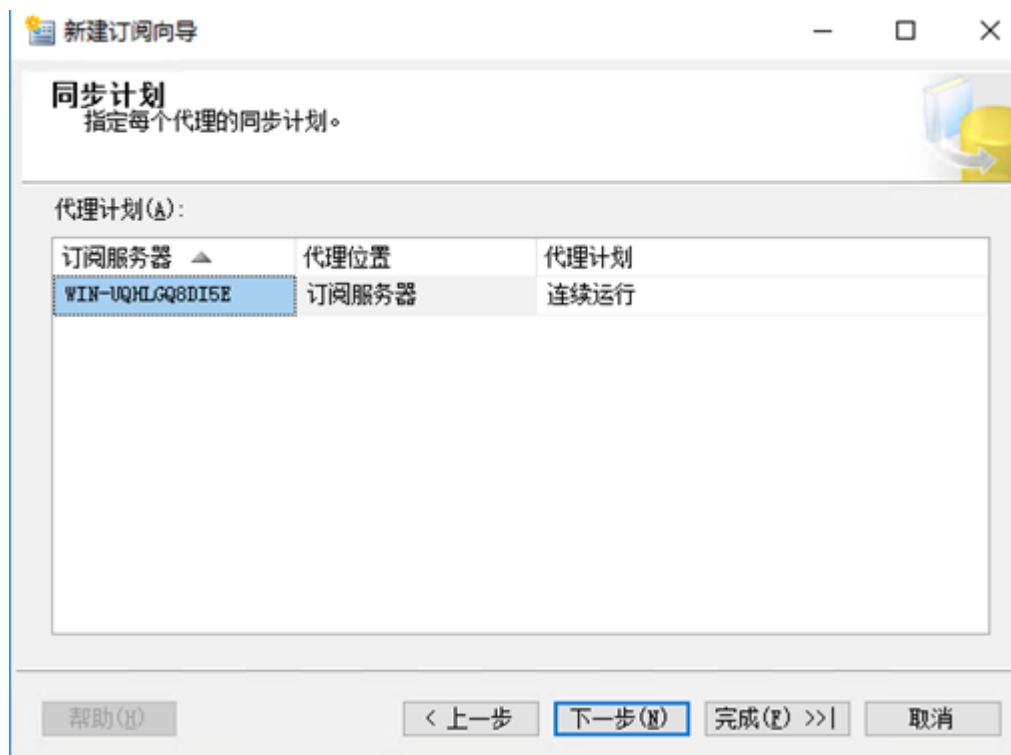
配置完后:

第21页 共55页

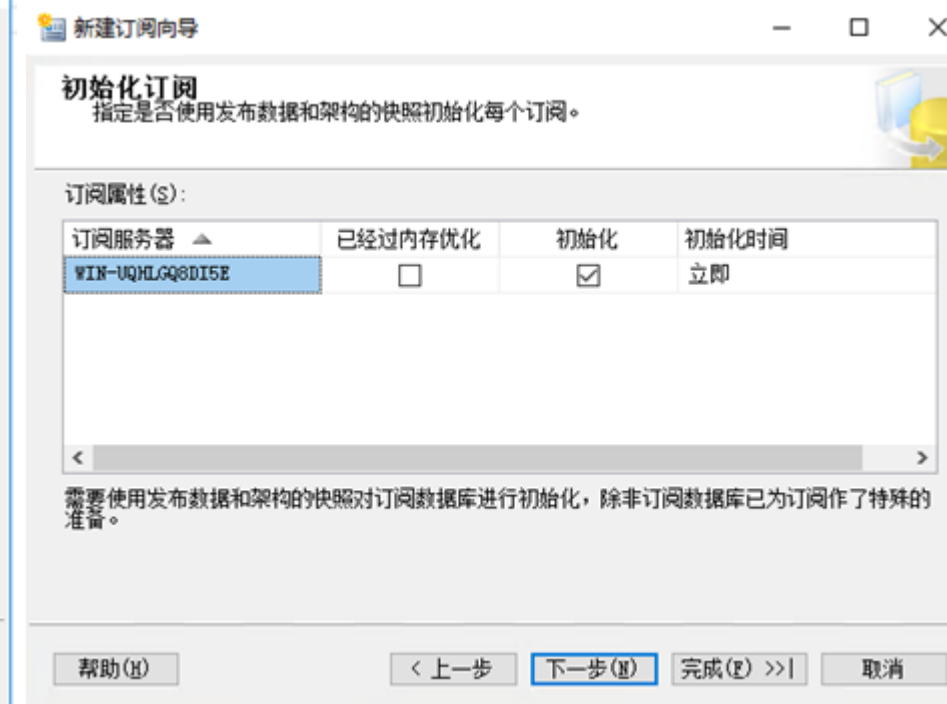
2023/11/3 21:42

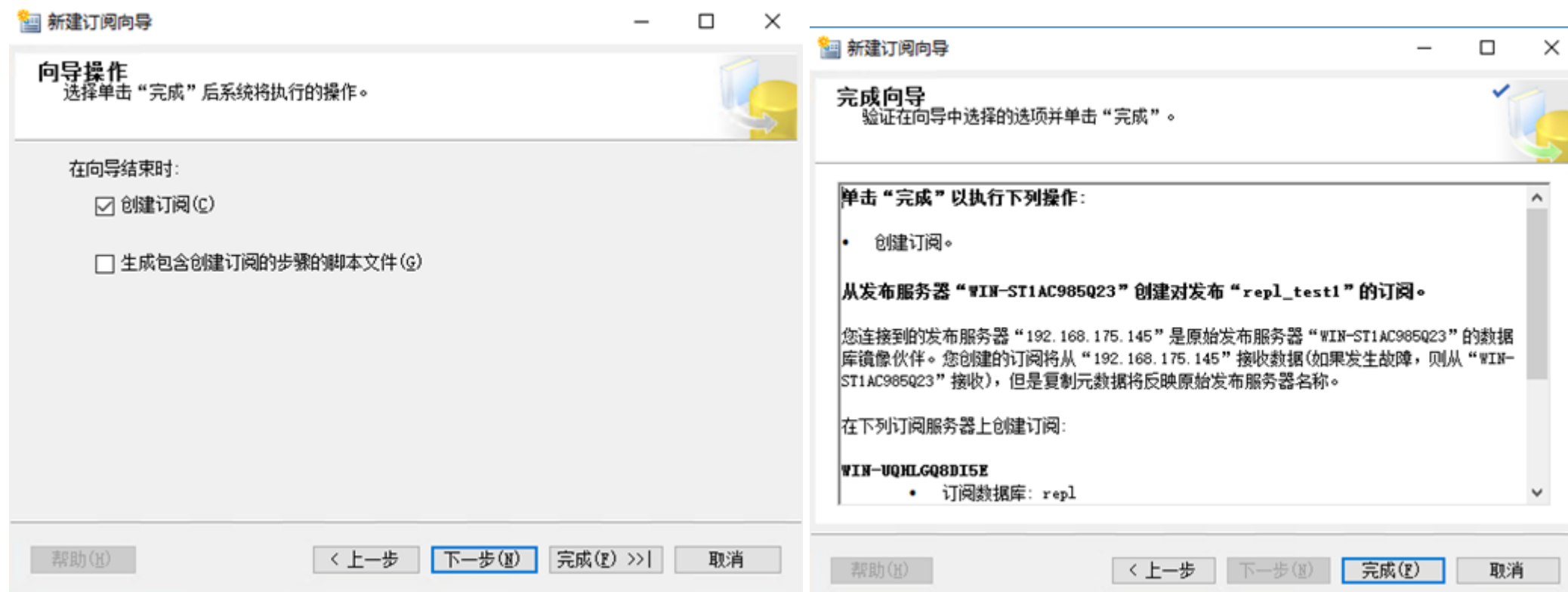


下一步,下一步

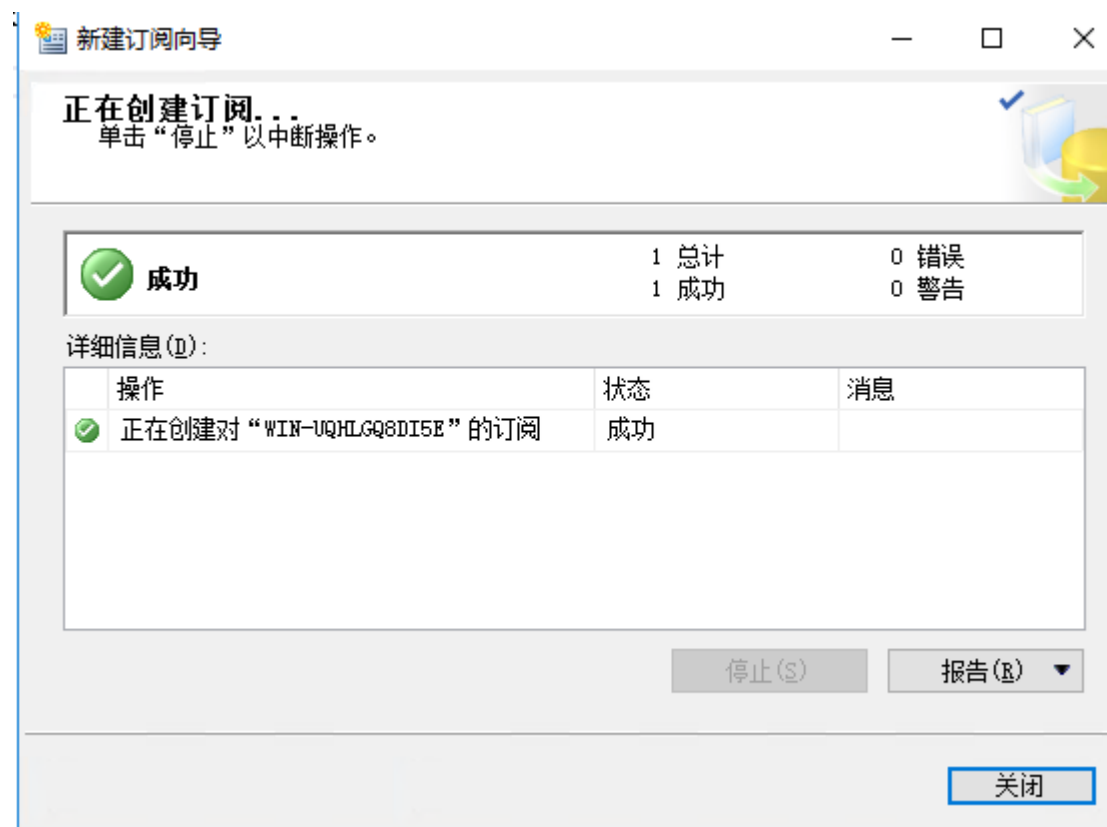


下一步, 完成





最终没有问题, 完成

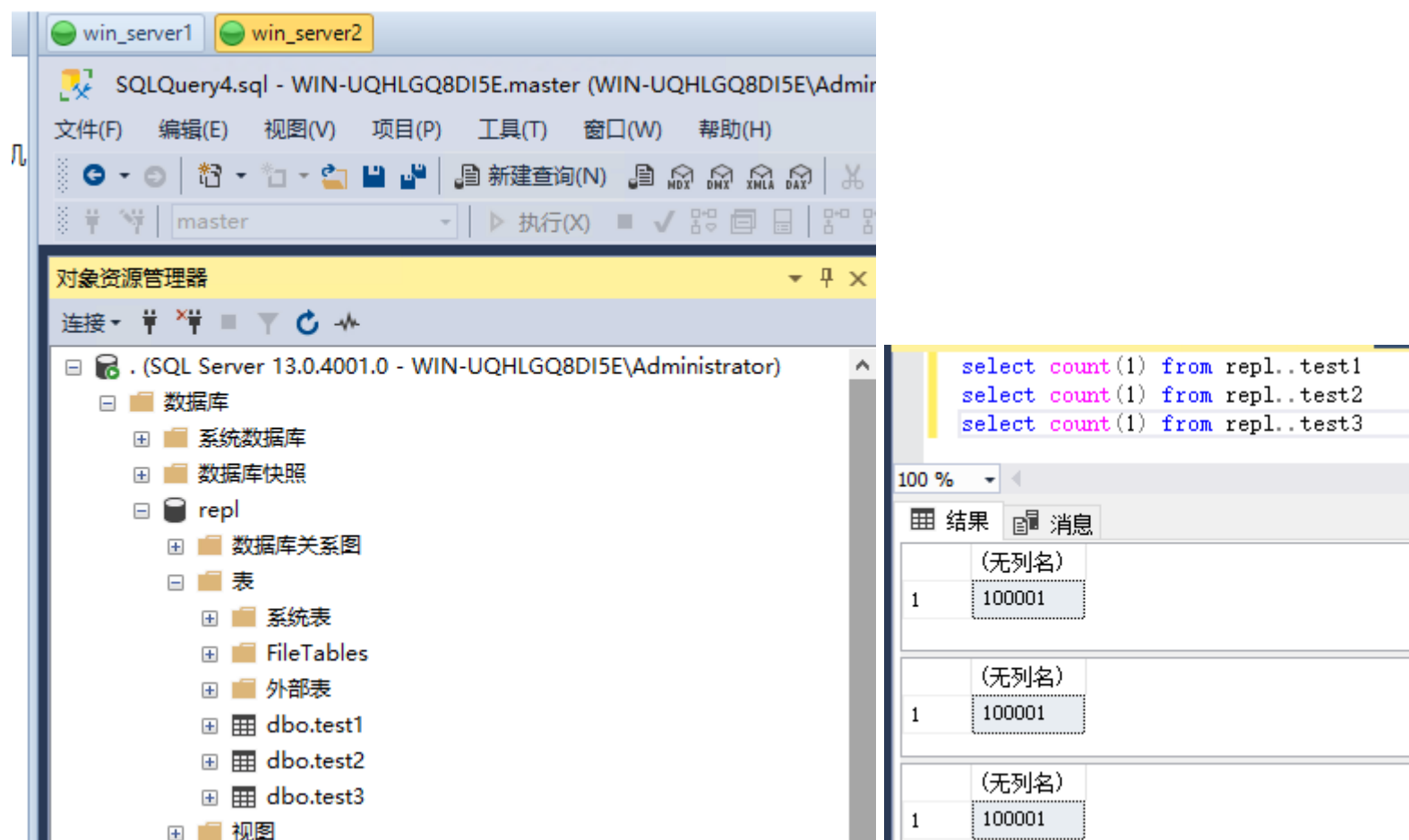


[回到顶部](#)

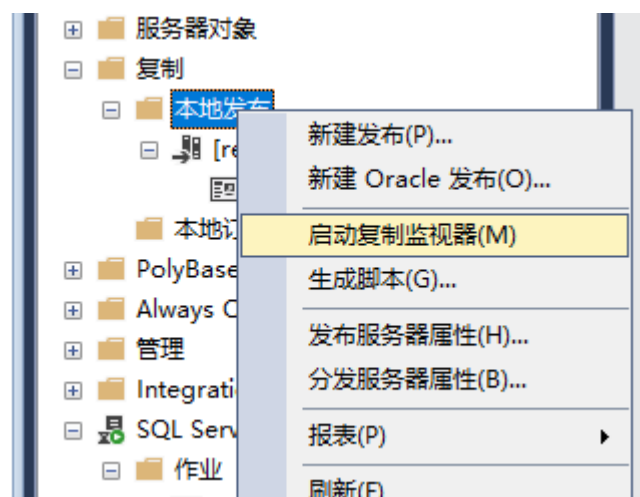
【3】核验与监视

【3.1】数据库查看

我们发布的3个表都已经复制过来了。如下图；数量也是对的

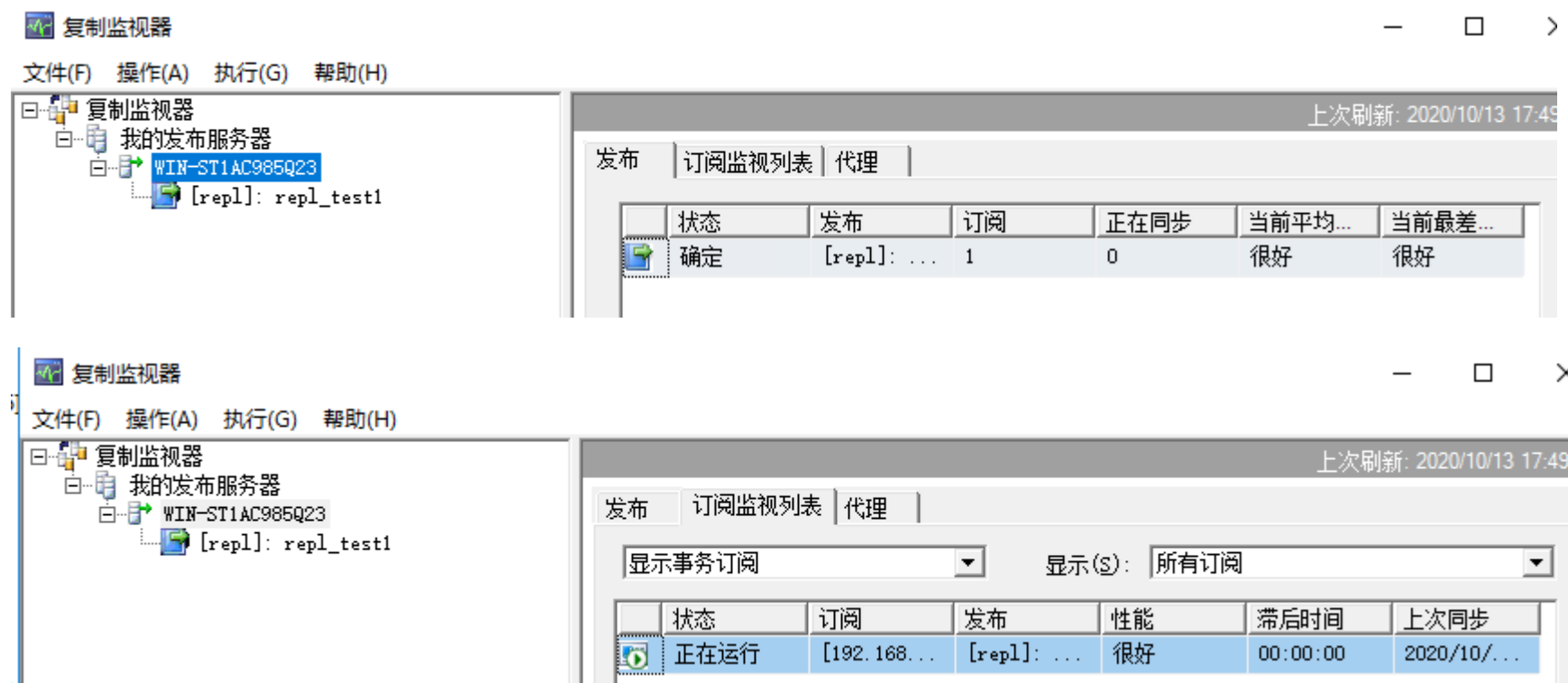


【3.2】复制监视器

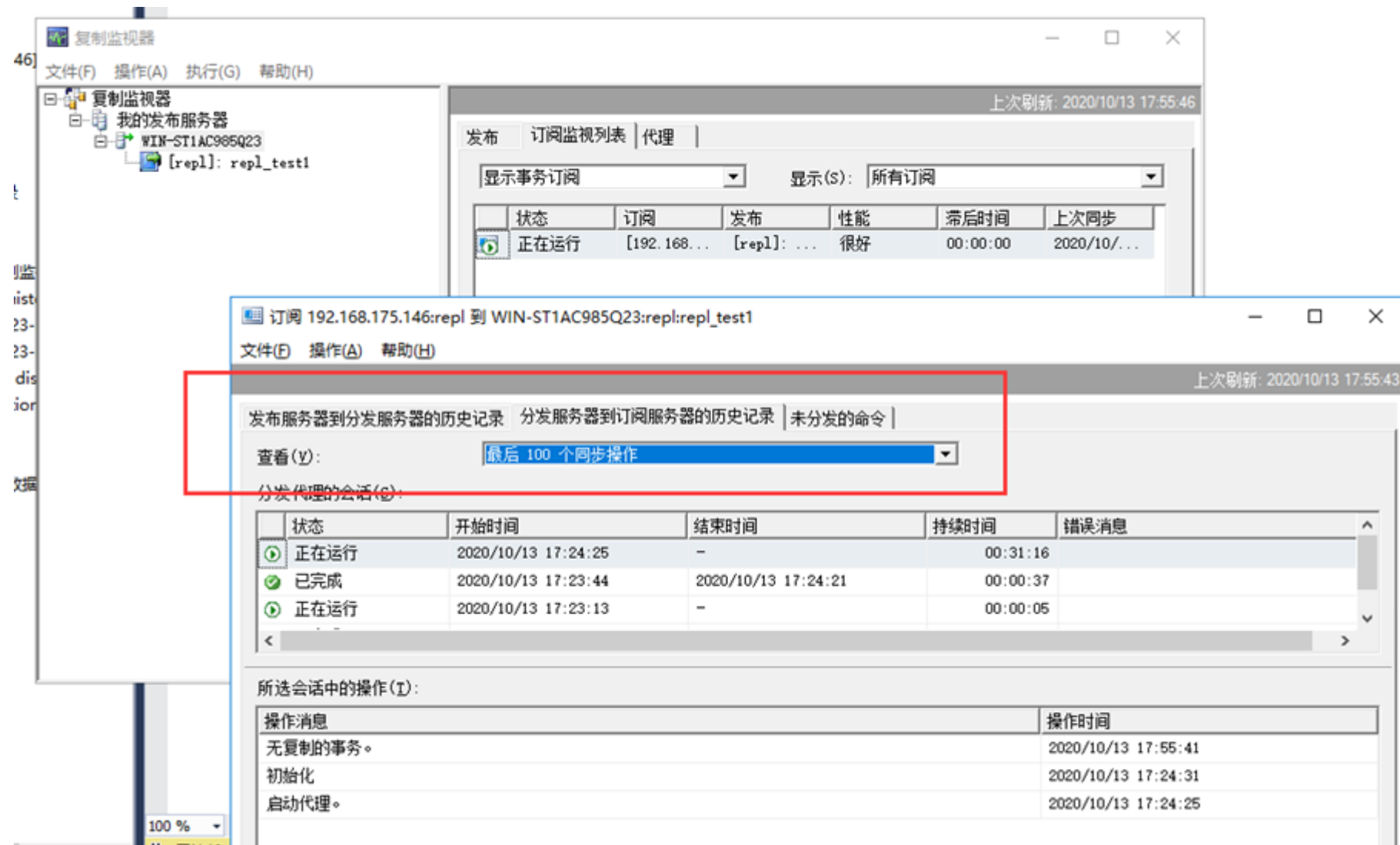


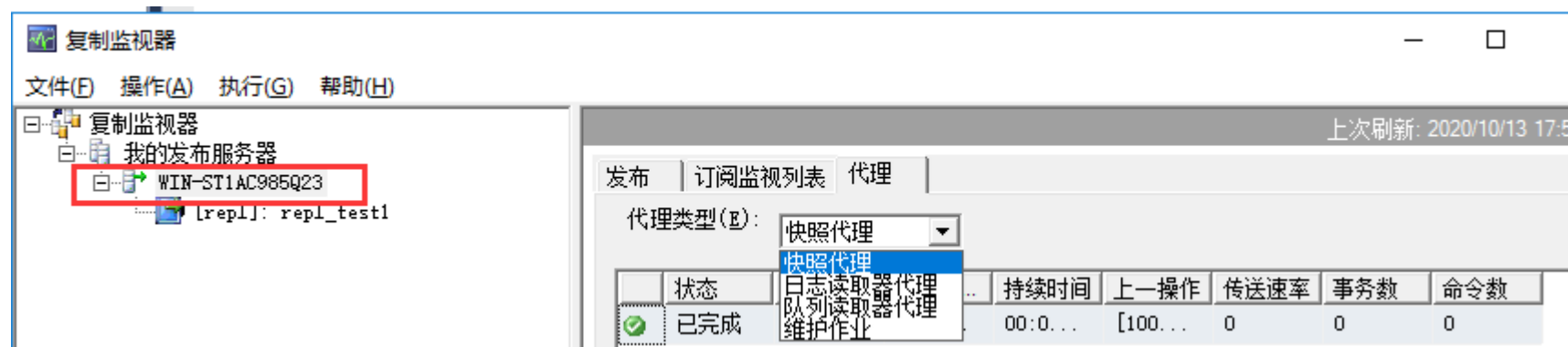
我们这里可以看发布、订阅、代理等信息

(1) 发布服务器查看



在这里，可以双击订阅监视列表数据，然后双击进去可以查看该行订阅的同步情况；





注意，这里的代理也是发布服务器的代理哦；

(2) 具体发布对应的信息查看

我们可以选中具体发布名称，然后就可以查看到和它相关的所有订阅，代理情况，发生的警告等等



【3.3】系统存储过程监视

参考：<https://docs.microsoft.com/zh-cn/sql/relational-databases/replication/monitor/programmatically-monitor-replication?view=sql-server-ver15>

| | |
|---|---|
| 分发服务器上运行 | |
| Use distribution; exec sp_replmonitorhelppublisher | 为与分发服务器关联的一个或多个发布服务器返回当前状态信息。 在分发服务器的分发数据库上执行此存储过程，用于监视复制。 |
| Use distribution; exec sp_replmonitorhelppublication | 返回发布服务器上一个或多个发布的当前状态信息。 在分发服务器的分发数据库上执行此存储过程，用于监视复制 |
| Use distribution; exec sp_replmonitorhelpsubscription Use distribution; exec sp_replmonitorhelpsubscription null,'repl',@publication_type=0 | 返回发布服务器上属于一个或多个发布的订阅的当前状态信息，并为每个返回的订阅返回一行。 在分发服务器的分发数据库上执行此存储过程，用于监视复制。 |

【3.4】表：查看复制表监视

(1) MSDB

```
select * from msdb..MSagentparameterlist --代理配置参数列表

select * from msdb..MSagent_parameters --代理配置文件对于当前配置参数

select * from msdb..MSagent_profiles --代理配置文件信息

select * from msdb..MSdistpublishers --发布服务器信息

select * from msdb..MSdistributiondbs --分发数据库及配置

select * from msdb..MSdistributor --心跳

select * from msdb..MSreplmonthresholdmetrics --报警选项阈值

select * from msdb..sysreplicationalerts --复制报警信息
```

(2) 分发数据库 distribution

```
select * from distribution..MSarticles --复制：发布的数据库对象

select * from distribution..MSrepl_commands --复制：复制的命令行

select * from distribution..MSrepl_errors --复制：复制错误
```

```
select * from distribution..MSrepl_transactions           --复制：事物的历史纪录

select * from distribution..MSreplication_monitordata --复制监视器参数/刷新时间/刷新信息

select * from distribution..MSdistribution_agents         --分发代理：客户端

select * from distribution..MSdistribution_history        --分发代理：历史记录

select * from distribution..MSlogreader_agents           --日志代理：客户端

select * from distribution..MSlogreader_history          --日志代理：历史记录

select * from distribution..MSsnapshot_agents            --快照代理：与本地分发服务器关联的每个

select * from distribution..MSsnapshot_history           --快照代理：历史记录

select * from distribution..MSsubscriber_info            --订阅服务器：信息

select * from distribution..MSsubscriber_schedule        --订阅服务器：代理运行计划

select * from distribution..MSsubscriptions             --订阅：每个数据库对象一行

select * from distribution..MSpublication_access         --账户：有权利访问发布订阅的

select * from distribution..MSpublications               --发布：每个发布一行

select * from distribution..MSpublicationthresholds      --发布：告警法制

select * from distribution..MSPublisher_databases       --发布服务器：发布数据库信息和实例引擎版本
```

(3) 发布数据库

```
select * from repl..sysarticles           --该数据库：参与复制的表

select * from repl..syspublications       --该数据库：参与的发布

select * from repl..syssubscriptions      --该数据库：被订阅的表对于的订阅信息

select * from repl..systranschemas       --该数据库：参与复制的表

select * from repl..sysarticleupdates    --该数据库：参与复制的表

select * from repl..sysarticleupdates    --该数据库：参与复制的表
```

(4) 订阅数据库

```
select * from repl..MSreplication_objects --该数据库：复制的对象信息

select * from repl..MSreplication_subscriptions --该数据库：订阅信息
```

[回到顶部](#)

【4】优化

【优化概述】（必看）

1.硬件、数据库设计：

- 使用SSD磁盘，有钱的可以使用PCI-e卡，SSD做RAID10，所有节点部署在内网。
- 更改事务隔离级别为 READ_COMMITTED_SNAPSHOT。说实话，SQL Server 默认事务隔离级别是读已提交，但是没有多版本的概念，需要设置为 READ_COMMITTED_SNAPSHOT才是我们理解上的读已提交。如果不设置这个，照样会读写互相阻塞。
- 在创建订阅之前，先将目标数据库的恢复模式改成简单，能够减少日志

2.发布、订阅设计：

- 仅发布需要的数据，对于将日志写入数据库而且还要同步复制的，DBA可直接拒绝。
- 表中存在大量非聚集索引，发布时可以不勾选复制非聚集索引选项。经过测试，虽然数据库引擎是在应用完快照最后创建索引的，但创建时间很长，有时甚至会出错。可以在后期手工建立。我们采用导出脚本方式，然后只保留非聚集索引的脚本在订阅库中建立。
- 如果资源允许，单独建立一个分发服务器。发布和分发服务器在一个服务器上会增加性能损耗，尤其是分发服务器下面有多个订阅服务器时。
- 分发库默认是distribution库，会有专门的作业对该库进行清理，名字就叫"分发清除：distribution"。但这个进程清除分发库的效率太低了，会导致分发库积累大量数据，需要修改运行间隔和每次清楚的数据量。详见之前的随笔：distribution库过大的问题
- 合理使用订阅方式，如果选择推送订阅，势必加大分发服务器的压力，但延迟低；如果选择请求订阅，代理服务是在订阅服务器上，减少了分发服务器压力，但延迟会增加。
- 快照代理的生成。对于生成几百GB数据库的快照，采用SSD盘可能在10分钟左右。但会造成很高的性能损耗，要在业务非高峰期生成快照。快照代理文件也要放在SSD盘上，如果资源允许，单独一个驱动盘也可以。
- 如果已经有备份了，那么可以通过备份直接来初始化订阅，省去了生成快照和应用快照的步骤。
- 新发布新的表对象，对于发布库很大的情况下重新初始化是不实际的，可以增量发布，详见sqlserver同步后在不重新初始化快照的情况下新增表
- 对于已发布的表有批处理更新，一种方法是将批量更新分成若干小批量执行，减小事务处理的数据量；另一种方法是通过存储过程执行更改，并将存储过程发布。推荐第一种
- 将项目分布在多个发不上，这样相当于变向的实现了多线程并行发布，当然也可以通过参数实现并行，下面会讲到

3.参数优化：

- 降低复制代理的详细级别，在初始测试、监视或调试期间除外。减小分发代理或合并代理的 -HistoryVerboseLevel 参数和 -OutputVerboseLevel 参数。这样可以减少为跟踪代理历史记录和输出而插入的新行数。相反，具有相同状态的以前的历史记录消息将更新为新的历史记录信息。提高测试、监视和调试的详细级别，这样就可以获得有关代理活动的尽可能多的信息。在系统开销不是很大的情况下，建议使用默认值1，本身这点对性能消耗可以忽略。
- 增大快照代理的BcpBatchSize参数，-BcpBatch参数表示一个批事务的行数。
- 将快照代理的- MaxNetworkOptimization设置为1，减少将不必要的信息发送到订阅服务器上。
- 使用快照代理、分发代理的 -MaxBcpThreads 参数（指定的线程数不应超过计算机上的处理器数）。此参数指定创建和应用快照时可以并行执行的大容量复制操作的数目。
- 为分发代理使用 -SubscriptionStreams 参数。-SubscriptionStreams 参数可以显著提高聚合复制吞吐量。它使到一台订阅服务器的多个连接可以并行应用批量更改，同时在使用单线程时保持多个事务特征的存在。如果有一个连接无法执行或提交，则所有连接将中止当前批处理，而且代理将用单独的流重试失败的批处理。在重试阶段完成之前，订阅服务器上会存在临时事务不一致。失败的批处理成功提交后，订阅服务器将恢复到事务一致状态。官方文档有这个参数，但增加这个参数报错。
- 增大分发代理的 -CommitBatchSize 和CommitBatchThreshold参数的值。提交一组事务的开销是固定的；通过不经常提交大量事务，就可以将开销分散在大量数据上。但是，增大此参数的优势因应用更改的开销由其他因素（例如包含日志的最大磁盘 I/O）限制而降低。另外，需要考虑以下权衡问题：任何导致分发代理重新开始的失败必须回滚并重新应用大量事务。对于不可靠的网络，越小的值导致失败的几率越小，如果导致失败，将回滚并重新应用较少量事务。
- 增大日志读取器代理的 -ReadBatchSize 参数的值。-ReadBatchSize 表示日志读取器代理和分发代理支持事务读取和提交操作的批大小。批大小的默认值为 500 个事务。日志读取器代理从日志中读取特定数量的事务，而不管这些事务是否标记为复制。将大量事务写入发布数据库，而其中只有一小部分标记为复制时，则应使用 -ReadBatchSize 参数增加日志读取器代理的读取批大小。此参数不适用于 Oracle 发布服务器。
- 为日志读取器代理使用 -MaxCmdsInTran 参数。-MaxCmdsInTran 参数指定日志读取器向分发数据库写入命令时组成一个事务的语句的最大数量。使用此参数能够使日志读取器

代理和分发代理在订阅服务器上应用命令时将发布服务器上的大事务（由许多命令组成）分成若干个较小的事务。指定此参数可以减少分发服务器的争用问题并缩短发布服务器与订阅服务器之间的滞后时间。由于初始事务是以较小的单元应用的，订阅服务器可以在初始事务结束之前访问一个较大的逻辑发布服务器事务的行，因而会破坏事务的原子性。默认值为 0，这将保持发布服务器的事务边界。官方文档有该参数，但实际设置里面没有。

- 减小日志读取器代理和分发代理的 `-PollingInterval` 参数的值。`-PollingInterval` 参数指定为要复制的事务查询已发布数据库的事务日志的频率。默认值为 5 秒。如果减小此值，日志的轮询将更加频繁，这会使事务从发布数据库传递到分发数据库的滞后时间较低。但是，应该对较低滞后时间的需要和因频繁轮询导致的服务器上增加的负荷之间进行平衡。

【4.0】官网使用参考

代理文件概念及参数官网：<https://docs.microsoft.com/zh-cn/sql/relational-databases/replication/agents/replication-agent-profiles?view=sql-server-ver15>

综合使用参考官网：<https://docs.microsoft.com/zh-cn/sql/relational-databases/replication/agents/work-with-replication-agent-profiles?view=sql-server-ver15>

T-SQL代码操作代理官网：<https://docs.microsoft.com/zh-cn/sql/relational-databases/replication/agents/work-with-replication-agent-profiles?view=sql-server-ver15#TsqlProcedure>

【4.1】快照初始化优化

--(0) 相关DMV

```
select * from msdb.dbo.MSagent_profiles --查看代理文件
```

```
select * from msdb..MSagentparameterlist --查看不同代理的默认参数及相关值
```

```
select * from msdb..msagent_parameters --查看当前所有代理文件的参数值
```

--(1) 创建一个新的代理文件

```
--查看最大ID是多少，没有设置的话，默认是16 select max(profile_id) from msdb.dbo.MSagent_profiles
```

--默认最大是16，我们新建的文件id就给17好了

```
sp_add_agent_profile @profile_id= 17
```

```
    , @profile_name= '优化快照代理'
```

```
    , @agent_type= 1 --快照代理
```

```
    , @profile_type= 1
```

```
    , @description= '自定义修改配置参数，优化快照代理'
```

```
    , @default= default
```

--（2）修改新增配置文件的配置参数

/*

修改配置文件，这里新增了

MaxBcpThreads（最大bcp使用线程），MaxNetworkOptimization（网络自动优化）参数，

修改了BcpBatchSize（bcp一次性操作中的最大行数），HistoryVerboseLevel参数，

对于性能提高最大的参数就是MaxBcpThreads，其他参数调整对性能调高不大，但MaxNetworkOptimization参数一定要调整到1，可以减少传输到订阅服务器上不必要的操作。

*/

```
USE msdb
```

```
EXEC sp_add_agent_parameter @profile_id = 17,
```

```
@parameter_name = 'MaxBcpThreads', @parameter_value = 4 --多现场会占用较多资源，开这个要合理，慎重
```

```
EXEC sp_add_agent_parameter @profile_id = 17,
```

```
@parameter_name = 'MaxNetworkOptimization', @parameter_value = 1
```

```
EXEC sp_change_agent_parameter @profile_id = 17,
```

```
@parameter_name = 'BcpBatchSize', @parameter_value = 200000 --2016默认已经为10W
```

```
EXEC sp_change_agent_parameter @profile_id = 17,
```

```
@parameter_name = 'HistoryVerboseLevel', @parameter_value = 1 -- 1（性能较好）：如果更新状态与上一次相同，则更新上一历史信息，否则新建 --2：默认，每次操作都记录成一条新的信息
```

【4.2】修改数据同步过程优化（日志代理与分发代理）

*（1）日志代理



--新建日志读取代理配置文件

```
sp_add_agent_profile @profile_id= 20
```

```
, @profile_name= '优化日志代理'
```

```
, @agent_type= 2 --日志代理
```

```
, @profile_type= 1
```

```
, @description= '自定义修改配置参数，优化日志代理'
```

```
        , @default= default

-- 查询该代理的profile_id
SELECT * FROM msdb.dbo .MSagent_profiles WHERE agent_type =2

EXEC sp_change_agent_parameter @profile_id = 20,
    @parameter_name = 'ReadBatchSize', @parameter_value = 2000

EXEC sp_change_agent_parameter @profile_id = 20,
    @parameter_name = 'PollingInterval', @parameter_value = 1
```

(2) 分发代理

新建分发参数配置文件，命名为"分发代理参数优化参数"

--新建日志读取代理配置文件

```
sp_add_agent_profile @profile_id= 18

    , @profile_name= '优化分发代理'

    , @agent_type= 3 --分发代理

    , @profile_type= 1

    , @description= '自定义修改配置参数，优化分发代理'

    , @default= default

EXEC sp_change_agent_parameter @profile_id = 18,
    @parameter_name = 'PollingInterval', @parameter_value = 1

EXEC sp_change_agent_parameter @profile_id = 18,
    @parameter_name = 'CommitBatchSize', @parameter_value = 1000

EXEC sp_change_agent_parameter @profile_id = 18,
    @parameter_name = 'CommitBatchThreshold', @parameter_value = 10000
```

【4.3】不重新初始化新增数据库对象

immediate_sync: 指定每次运行快照代理时是否为发布创建同步文件。 immediate_synchronization 的数据类型为 nvarchar(5), 默认值为 FALSE。

如果为 True, 表示每次运行快照代理时都创建或重新创建同步文件。 如果快照代理在订阅创建前完成, 则订阅服务器可以立即获得同步文件。 新订阅将获取最近一次执行快照代理所生成的最新同步文件。 independent_agent 必须为 true, 以便于 immediate_synchronization 为 true。 如果为 False, 则仅当有新订阅时, 才创建同步文件。

当以增量方式向现有发布添加新项目时, 必须为每个订阅调用 sp_addsubscription。 订阅后订阅服务器无法接收同步文件, 直到启动并完成快照代理为止。

allow_anonymous: 指定是否可为给定发布创建匿名订阅。 allow_anonymous 的数据类型为 nvarchar(5), 默认值为 FALSE。

如果为 True, 则 immediate_synchronization 也必须设置为 True。 如果为 False, 则表示不允许对该发布创建匿名订阅。

从定义中看出来, immediate_sync为false时, 新的项目(表、存储过程等)可以以增量方式发布, 而allow_anonymous必须为false, immediate_sync才能为false。



```
--查阅发布对应的这2个参数当前值 select * from distribution..MSpublications

use msterdb ;
go

EXEC sp_changepublication @publication = 'repl_master', @property = 'allow_anonymous' ,@value = 'false'
GO

EXEC sp_changepublication @publication = 'repl_master', @property = 'immediate_sync' ,@value = 'false'
GO
```



修改前: 更新发布, 项目中新增表之后, 启动快照代理=》如下图, 发现所有发表表全部初始化了

| | | | | |
|-------|---------------------|---------------------|----------|--|
| ✔ 已完成 | 2020/10/14 18:07:10 | 2020/10/14 18:07:34 | 00:00:24 | |
| ✔ 已完成 | 2020/10/13 17:24:32 | 2020/10/13 17:24:34 | 00:00:02 | |
| ✔ 已完成 | 2020/10/13 17:23:38 | 2020/10/13 17:23:43 | 00:00:05 | |

所选会话中的操作(T):

| 操作消息 | 操作时间 |
|---|---------------------|
| [100%] 已生成 4 个项目的快照。 | 2020/10/14 18:07:34 |
| [99%] 在生成快照时锁定已发布的表 | 2020/10/14 18:07:33 |
| [99%] 已将用于在并发快照期间禁用约束/触发器项目“test4”的存储过程插入到分发数据库中。 | 2020/10/14 18:07:33 |
| [99%] 已将用于在并发快照期间禁用约束/触发器项目“test3”的存储过程插入到分发数据库中。 | 2020/10/14 18:07:33 |
| [99%] 已将用于在并发快照期间禁用约束/触发器项目“test2”的存储过程插入到分发数据库中。 | 2020/10/14 18:07:33 |
| [99%] 已将用于在并发快照期间禁用约束/触发器项目“test1”的存储过程插入到分发数据库中。 | 2020/10/14 18:07:33 |
| [99%] 已将项目“test4”的 bcp 命令插入到分发数据库中。 | 2020/10/14 18:07:33 |
| [99%] 已将项目“test3”的 bcp 命令插入到分发数据库中。 | 2020/10/14 18:07:33 |
| [99%] 已将项目“test2”的 bcp 命令插入到分发数据库中。 | 2020/10/14 18:07:33 |
| [99%] 已将项目“test1”的 bcp 命令插入到分发数据库中。 | 2020/10/14 18:07:33 |
| [99%] 已将项目“test4”的索引创建命令插入到分发数据库中。 | 2020/10/14 18:07:33 |
| [99%] 已将项目“test3”的索引创建命令插入到分发数据库中。 | 2020/10/14 18:07:33 |
| [99%] 已将项目“test2”的索引创建命令插入到分发数据库中。 | 2020/10/14 18:07:33 |
| [99%] 已将项目“test1”的索引创建命令插入到分发数据库中。 | 2020/10/14 18:07:33 |

修改后：更新发布，项目中新增表之后，启动快照代理=》如下图，发现只有 test5表更新过去了

动

| | 状态 | 开始时间 | 结束时间 | 持续时间 | 错误消息 |
|---|-----|---------------------|---------------------|----------|------|
| ✔ | 已完成 | 2020/10/14 18:53:31 | 2020/10/14 18:53:35 | 00:00:04 | |
| ✔ | 已完成 | 2020/10/14 18:07:10 | 2020/10/14 18:07:34 | 00:00:24 | |
| ✔ | 已完成 | 2020/10/13 17:24:32 | 2020/10/13 17:24:34 | 00:00:02 | |
| ✔ | 已完成 | 2020/10/13 17:23:38 | 2020/10/13 17:23:43 | 00:00:05 | |

所选会话中的操作(T):

| 操作消息 | 操作时间 |
|---|---------------------|
| [100%] 已生成 1 个项目的快照。 | 2020/10/14 18:53:35 |
| [19%] 在生成快照时锁定已发布的表 | 2020/10/14 18:53:35 |
| [19%] 已将用于在并发快照期间禁用约束/触发器项目“test5”的存储过程插入到分发数据库中。 | 2020/10/14 18:53:35 |
| [19%] 已将项目“test5”的 bcp 命令插入到分发数据库中。 | 2020/10/14 18:53:35 |
| [19%] 已将项目“test5”的索引创建命令插入到分发数据库中。 | 2020/10/14 18:53:35 |
| [19%] 已将项目“test5”的架构命令插入到分发数据库中 | 2020/10/14 18:53:35 |
| [19%] 正在将快照命令投递到分发数据库中 | 2020/10/14 18:53:35 |
| [19%] 正在处理系统快照前脚本 | 2020/10/14 18:53:35 |
| [19%] 正在为项目“test5”生成架构脚本 | 2020/10/14 18:53:34 |
| [19%] 分析非项目对象的依赖关系 | 2020/10/14 18:53:34 |
| [19%] 分析所有已复制的检查约束和默认约束的外部引用 | 2020/10/14 18:53:34 |

【4.4】使用T-SQL，不初始化 增删表



```
-- 若提前“禁止架构更改”，新增的列不会自动添加大发布，此时应使用 sp_articlecolumn 添加列
EXEC sp_changepublication @publication = N'publication', @property = N'replicate_ddl', @value = 0

-- 【移除和添加发布而不初始化所有项目，但是添加的项目须初始化】

-- 禁止匿名访问
Exec sp_changepublication 'publication','allow_anonymous',false
GO

-- 禁止立即更新
Exec sp_changepublication 'publication','immediate_sync',false
GO

-- 终止单个项目复制（该表在发布可正常操作，订阅应禁止操作）
Exec sp_dropsubscription @publication = N'publication', @article = N'article', @subscriber = N'subscriber', @destination_db = N'destination_db'
GO

-- 从发布中删除项目（界面操作将会重新初始化所以项目）
Exec sp_droparticle @publication = N'publication', @article = N'article'
GO

-- 添加项目到发布中（@schema_option 参考个人要求更改）
Exec sp_addarticle @publication = 'publication', @article = N'article', @source_object = N'article', @source_owner = N'dbo',@schema_option = 0x0000000
GO


-- 添加未发布的列（对已发布中的表添加多一列，一般不用）
Exec sp_articlecolumn @publication = 'publication' ,@article = N'article',@column = N'column',@operation = N'add'
GO

-- 进行行筛选（用于筛选发布）
Exec sp_articlefilter @publication = N'publication', @article = N'article', @filter_name = N'FLTR_article_1__1', @filter_clause = N'id % 2 = 0'
GO

-- 进行行筛选（用于筛选发布）
Exec sp_articleview @publication = N'publication', @article = N'article', @view_name = N'SYNC_article_1__1', @filter_clause = N'id % 2 = 0'
GO

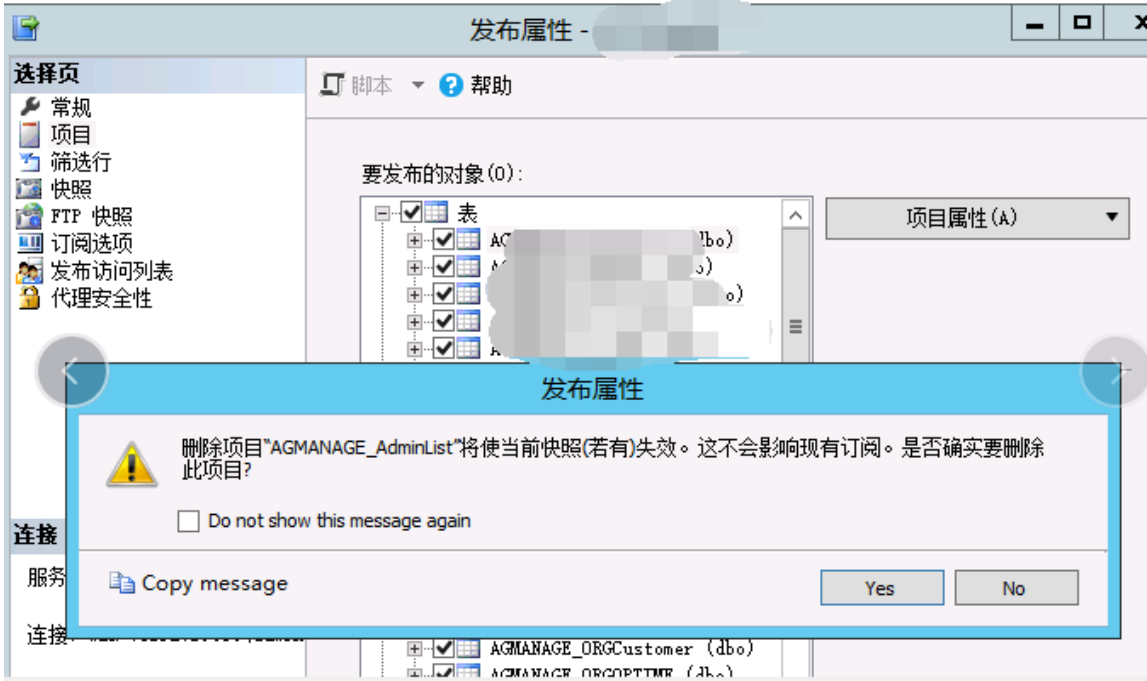
-- 刷新订阅以添加项目
Exec sp_refreshsubscriptions 'publication'
GO

-- 启动快照代理（初始化新增的项目）
Exec sp_startpublication_snapshot 'publication'
GO
```



不初始化，删除表案例

不初始化，删除表案例：（注意，这样会导致新建订阅时无法应用现有的快照连接发布，必须重新运行快照代理后，才可以,如下图）



这种办法，不会阻塞，不需要发布的项目全表初始化；（当然，直接用界面操作也是没有问题的）

```
select * from distribution..mspublications -- 查看 publication
select * from distribution..mssubscriber_info -- 查看 subscriber
select * from distribution..MSsubscriptions -- 查看 destination_db
select * from distribution..MSarticles -- 查看所有发布的项目（即表）

-- 终止单个项目复制（该表在发布可正常操作，订阅应禁止操作）
Exec sp_dropsubscription @publication = N'publication', @article = N'article', @subscriber = N'subscriber', @destination_db = N'destination_db'
GO

-- 从发布中删除项目（界面操作将会重新初始化所以项目）
Exec sp_droparticle @publication = N'publication', @article = N'article'
GO

--主从都要
drop table article
```

【4.5】解决大事务引起的发布订阅效率问题

(1) 大事务引起的问题

在 SQLServer 复制中，当在发布数据库执行1个大事务时，如一次性操作十万行或百万行以上的数据。当操作数据在发布数据库执行完成后，日志读取器代理将扫描事务日志，一次性传递到分发数据库中。

若上个事务未传递完成，连续执行多个事务，日志读取器代理将扫描日志中多个事务同时传递到分发数据库中，默认最大扫描500个事务。如果执行多次上百万或千万的数据将堵塞很久。

(2) 解决

日志读取器代理可配置将大事务划分为多个小事务进行传递到分发数据库中，分发队列则按照小事务分发到订阅数据库中，这样数据就很快同步！

在没改代理参数之前，本人执行1次插入30万的数据到发布表中。插入完成后，监控发布到分发的记录如下：

| 发布服务器到分发服务器的历史记录 | | | | | |
|---|------|--------------------|--------------------|------------|--------------------|
| 查看 (V): 最后 100 个同步操作 | | | | | |
| 日志读取器代理的会话 (S): | | | | | |
| | 状态 | 开始时间 | 结束时间 | 持续时间 | 错误消息 |
| ▶ | 正在运行 | 2015-5-30 18:00:46 | - | 00:02:00 | |
| ✔ | 已完成 | 2015-5-30 17:08:50 | 2015-5-30 18:00:41 | 00:51:51 | |
| ✔ | 已完成 | 2015-5-30 12:16:36 | 2015-5-30 17:08:44 | 04:52:08 | |
| ✔ | 已完成 | 2015-5-29 11:09:26 | 2015-5-30 12:16:32 | 1.01:07:06 | |
| 所选会话中的操作 (I): | | | | | |
| 操作消息 | | | | | 操作时间 |
| 传递了 1 个事务，使用了 300000 个命令。 | | | | | 2015-5-30 18:02:46 |
| 日志读取器代理正在扫描事务日志，从中查找要复制的命令。在第 3 遍中扫描了大约 500000 个日志记录... | | | | | 2015-5-30 18:02:46 |
| 初始化 | | | | | 2015-5-30 18:00:47 |
| 启动代理。 | | | | | 2015-5-30 18:00:46 |

可以看到，这1个事务的命令都得一次传递完才能分发，而分发又消耗时间，这里等待太久影响事务的实时性。如果还有其他事务，默认500（参考参数：-ReadBatchSize），也将一起传递，不过耗时较长。

现在更改参数，扫描到 1000 左右的命令就即时分发，需要设置如下参数：

-MaxCmdsInTran number_of_commands

步骤名称 (S):
运行代理。

类型 (T):
复制事务-日志读取器

运行身份 (R):
[REPL] Sync-1

命令 (M):
utor [kk-DB] -DistributorSecurityMode 1 -MaxCmdsInTran 1000 -Continuous

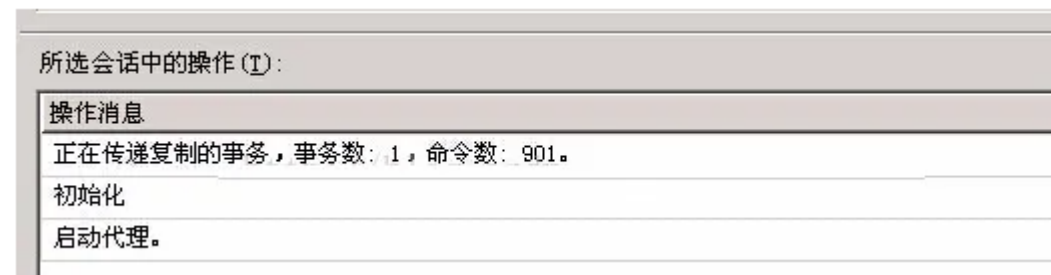
全选 (A)

复制 (C)

粘贴 (P)

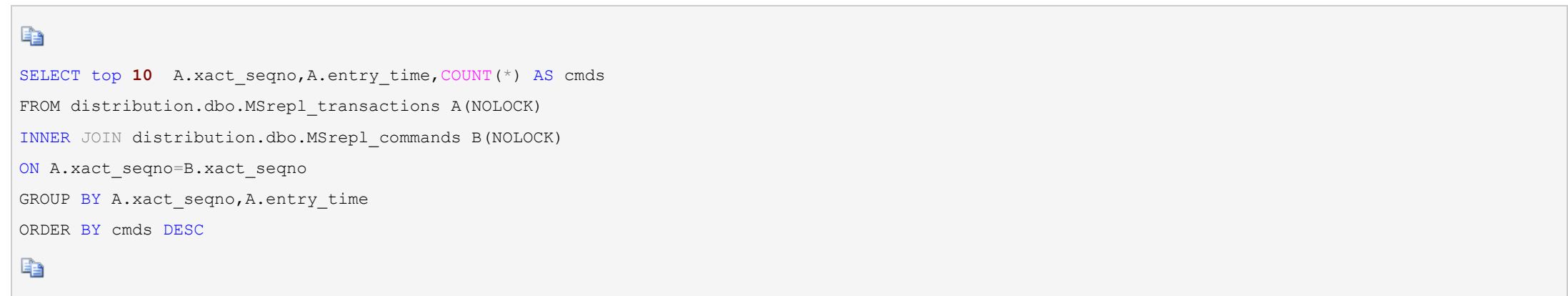
注：该参数只能添加到日志读取器代理中，在代理配置文件没有此参数的设置。

添加后重启日志读取器代理。再次插入 30 万的数据，到监视器查看。



可以看到，命令达到 1000 左右就进行分发了，此时查看订阅数据库，数据也同步过来了，这样就省去了较多扫描命令的时间。

更详细查看每个事务的命令数，如下：



| | xact_seqno | entry_time | cmds |
|----|-------------------------------|-------------------------|------|
| 1 | 0x000DA7140000B3B00A500000018 | 2015-07-01 16:14:51.673 | 1000 |
| 2 | 0x000DA7140000B3B00A50000000C | 2015-07-01 16:14:45.060 | 1000 |
| 3 | 0x000DA7140000B3B00A50000000A | 2015-07-01 16:14:43.670 | 1000 |
| 4 | 0x000DA7140000B3B00A500000004 | 2015-07-01 16:14:40.400 | 1000 |
| 5 | 0x000DA7140000B3B00A5 | 2015-07-01 16:14:37.300 | 1000 |
| 6 | 0x000DA7140000B3B00A50000001C | 2015-07-01 16:14:53.973 | 1000 |
| 7 | 0x000DA7140000B3B00A50000001E | 2015-07-01 16:14:55.100 | 1000 |
| 8 | 0x000DA7140000B3B00A500000024 | 2015-07-01 16:14:58.857 | 1000 |
| 9 | 0x000DA7140000B3B00A500000026 | 2015-07-01 16:15:01.240 | 1000 |
| 10 | 0x000DA7140000B3B00A500000028 | 2015-07-01 16:15:02.517 | 1000 |

这个参数虽好，但是也可能引起数据的一致性。如：在发布更新了一批数据，但是订阅查询时却有不同。分发事务遇到冲突或者死锁，也导致这部分的数据不一致。

[回到顶部](#)

【5】基本原理

【5.0】事务复制的原理

(1) 当publication database的article发生更新时，会产生相应的日志，Log reader会读取这些日志信息，将他们写入到Distribution 数据库的msrepl_transactions和msrepl_commands中。

(2) Msrepl_transactions中的每一条记录都有一个唯一标识xact_seqno，xact_seqno对应日志中的LSN。所以可以通过xact_seqno推断出他们在publication database

中的生成顺序，编号大的生成时间就晚，编号小的生成时间就早。

(3) Distributionagent包含两个子进程，reader和writer。Reader负责从Distribution 数据库中读取数据，Writer负责将reader读取的数据写入到订阅数据库。

(4) Reader是通过sp_MSget_repl_commands来读取Distribution数据库中（读取Msrepl_transactions表和Msrepl_Commands表）的数据。

(5) 大致逻辑是：Reader读取subscription database的MSreplication_subscriptions表的transaction_timestamp列，获得更新的上一次LSN编号，然后读取分发数据库中LSN大于这个编号的数据。

Writer将读取到的数据写入订阅，并更新MSreplication_subscriptions表的transaction_timestamp列。然后Reader会继续用新的LSN来读取后续的数据，再传递给Writer，如此往复。

注意：如果我们手工更新transaction_timestamp列,将这个值设置为当前正在执行的大事务的LSN，那么distribution agent就会不读取这个大事务，而是将其跳过了。

【5.1】发布订阅的角色与概念

复制使用出版业术语表示复制拓扑中的组件，其中包括发布服务器、分发服务器、订阅服务器、发布、项目和订阅。可借助杂志的概念来帮助理解 Microsoft SQL Server 复制：

- 杂志出版商（发布服务器）生产一种或多种刊物（发布） publisher
- 刊物（发布）包含文章（项目） articles
- 出版商（发布服务器）可以直接发行（分发）杂志，也可以使用发行商（分发服务器） distributor
- 订阅者（订阅服务器）接收订阅的刊物（发布） subscriber

虽然杂志术语有助于理解复制，但重要的是要注意到 SQL Server 复制包含有这套术语未予以表述的功能，尤其是订阅服务器进行更新的功能以及发布服务器将增量更改发送到发布中的项目的功能。

"复制拓扑" 定义了服务器和数据副本间的关系，并阐明了决定数据如何在服务器之间流动的逻辑。有若干复制进程（称为"代理"）负责在发布服务器和订阅服务器之间复制和移动数据。下图为复制中所涉及的组件和进程的概述。

复制组件和数据流

发布者publisher

发布服务器是一种数据库实例，它通过复制向其他位置提供数据。发布服务器可以有一个或多个发布，每个发布定义一组要复制的具有逻辑关系的对象和数据。

分发服务器distributor

分发服务器也是一种数据库实例，它起着存储区的作用，用于复制与一个或多个发布服务器相关联的特定数据。每个发布服务器都与分发服务器中的单个数据库（称作分发数据库）相关联。分发数据库存储复制状态数据和有关发布的元数据，并且在某些情况下为从发布服务器向订阅服务器移动的数据起着排队的作用。在很多情况下，一个数据库服务器实例充当发布服务器和分发服务器两个角色。这称为"本地分发服务器"。当发布服务器和分发服务器按各自的数据库服务器实例配置时，把分发服务器称为"远程分发服务器"。

订阅服务器subscriber

订阅服务器是接收复制数据的数据库实例。订阅服务器可以接收来自多个发布服务器和发布的数据。根据所选的复制类型，订阅服务器还可以将数据更改传递回发布服务器或者将数据重新发布到其他订阅服务器。

项目articles

项目用于标识发布中包含的数据库对象。一次发布可以包含不同类型的项目，包括表、视图、存储过程和其他对象。当把表作为项目发布时，可以用筛选器限制发送到订阅服务

器的数据的列和行。

发布publication

发布是一个数据库中的一个或多个项目的集合。 将多个项目分组成一个发布，使得更便于指定一组作为一个单元复制的、具有逻辑关系的数据库对象和数据。

订阅subscribe

订阅是把发布副本传递到订阅服务器的请求。 订阅定义将接收的发布和接收的时间、地点。 有两种类型的订阅：推送订阅和请求订阅。

有关推送订阅和请求订阅的详细信息，请参阅订阅发布。

【5.2】复制类型与订阅类型

| 类型 | 说明 |
|-----------------------|---|
| 事务复制 | 在发布服务器上进行的更改会在发生时（几乎实时）传递给订阅服务器。 数据更改将按照其在发布服务器上发生的顺序和事务边界应用于订阅服务器。 |
| 合并复制 | 可以在发布服务器和订阅服务器上更改数据，并且可以通过触发器进行跟踪。 订阅服务器在连接到网络时将与发布服务器进行同步，并交换自上次同步以来发布服务器和订阅服务器之间发生更改的所有行。 |
| 快照复制 | 将来自发布服务器的快照应用于订阅服务器，这完全按照数据在特定时刻的状态分发数据，而不监视数据是否更新。 发生同步时，将生成完整的快照并将其发送到订阅服务器。 |
| 对等 | 对等复制建立在事务复制的基础之上，以事务方式近乎实时地在多个服务器实例之间传播一致的更改。 |
| 双向 | 双向事务复制是一种特定的事务复制拓扑，它允许两台服务器相互交换更改：每台服务器均发布数据，然后从另一台服务器订阅包含相同数据的发布。 |
| 可更新订阅 | 以事务复制为基础而构建，在订阅服务器上针对可更新订阅更新了数据时，会首先传播到发布服务器，然后传播到其他订阅服务器。 |

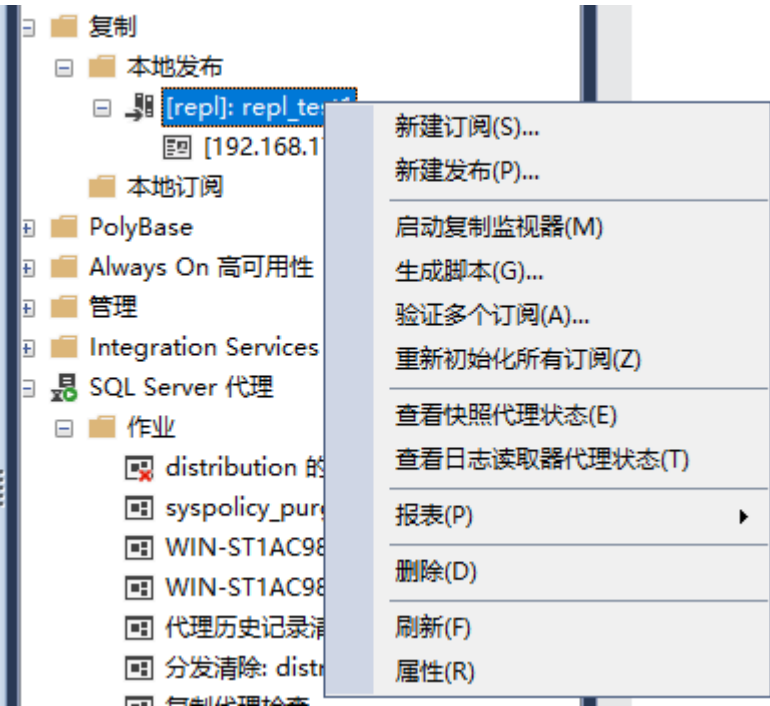
| 订阅 | 特征 | 使用时间 |
|------|--|---|
| 推送订阅 | 对于推送订阅，发布服务器将更改传播到订阅服务器，而无需订阅服务器发出请求。 更改可以按需、连续地或按照计划推送到订阅服务器。 分发代理或合并代理在分发服务器上运行。 | 通常，数据将连续同步或按照经常重复执行的计划同步。 发布要求数据近似实时地移动。 |

| | | |
|------|---|--|
| | | 分发服务器上较高的处理器开销不会影响性能。 通常与快照和事务复制一起使用。 |
| 请求订阅 | 对于请求订阅，订阅服务器请求在发布服务器上所做的更改。请求订阅允许订阅服务器上的用户确定同步数据更改的时间。分发代理或合并代理在订阅服务器上运行。 | 数据通常按需或按计划同步，而非连续同步。 发布具有大量订阅服务器，并且/或在分发服务器上运行所有代理会消耗大量资源。 订阅服务器是自主的、断开连接的和/或移动的。订阅服务器将确定连接和同步更改的时间。 通常与合并复制一起使用。 |

【5.3】复制代理

参考：<https://docs.microsoft.com/zh-cn/sql/relational-databases/replication/agents/replication-agent-profiles?view=sql-server-ver15>

我们右击发布可以看到许多代理，如下图，那么他们到底是什么呢？



(1) 快照代理

快照代理通常与各种类型的复制一起使用。快照代理准备已发布表的架构和初始数据文件以及其他对象、存储快照文件并记录分发数据库中的同步信息。

快照代理在分发服务器上运行。有关详细信息，请参阅 Replication Snapshot Agent。

(2) 日志读取器代理

日志读取器代理与事务复制一起使用。它将发布服务器上的事务日志中标记为复制的事务移至分发数据库中。使用事务复制发布的每个数据库都有自己的日志读取器代理，该代理运行于分发服务器上并与发布服务器连接（分发服务器与发布服务器可以是同一台计算机）。有关详细信息，请参阅 Replication Log Reader Agent。

(3) 分发代理

分发代理与快照复制和事务复制一起使用。它将初始快照应用于订阅服务器，并将分发数据库中保存的事务移至订阅服务器。分发代理既可以运行于分发服务器（对于推送订阅），也可运行于订阅服务器（对于请求订阅）。有关详细信息，请参阅 Replication Distribution Agent。

(4) 合并代理

合并代理与合并复制一起使用。它将初始快照应用于订阅服务器，并移动和协调所发生的增量数据更改。每个合并订阅都有自己的合并代理，该代理同时连接到发布服务器和订阅服务器并对它们进行更新。合并代理既可以运行于分发服务器（对于推送订阅），也可以运行于订阅服务器（对于请求订阅）。默认情况下，合并代理将订阅服务器上的更改上载到发布服务器，然后将发布服务器上的更改下载到订阅服务器。有关详细信息，请参阅 Replication Merge Agent。

(5) 队列读取器代理

队列读取器代理与包含排队更新选项的事务复制一起使用。该代理运行于分发服务器，并将订阅服务器上所做更改移回至发布服务器。与分发代理和合并代理不同，只有一个队列读取器代理的实例为给定分发数据库的所有发布服务器和发布提供服务。有关队列读取器代理的详细信息，请参阅 Replication Queue Reader Agent。有关可更新订阅的详细信息，请参阅 Updatable Subscriptions for Transactional Replication。

(6) 复制维护作业

创建复制之后，会在很代理中创建很多作业。复制包含许多执行计划维护和按需维护的维护作业。

【5.4】架构更改的注意事项

架构更改的注意事项

复制架构更改时，请牢记下列注意事项：

一般注意事项

- 架构更改需遵守 Transact-SQL规定的所有限制。例如，ALTER TABLE 不允许对主键列执行 ALTER 语句。
- 仅对初始快照执行数据类型映射。架构更改不会映射到以前版本的数据类型。例如，如果在 ALTER TABLE ADD datetime2 column 中使用 SQL Server 2012 (11.x)语句，则不会将数据类型转换为 ssVersion2005 订阅服务器的 SQL Server 2005 (9.x)。在某些情况下，架构更改在发布服务器上受到阻止。
- 如果发布被设置为允许传播架构更改，则不论为发布中的项目设置的相关架构选项是什么，都会传播架构更改。例如，如果选择不复制某个表项目的外键约束，但随后又发出 ALTER TABLE 命令，将外键添加到发布服务器中的表中，那么外键将被添加到订阅服务器中的表中。若要防止出现这种情况，请在发出 ALTER TABLE 命令之前禁用架构更改的传播。
- 架构更改只应在发布服务器中进行，不应在订阅服务器中（包括重新发布订阅服务器）上进行。合并复制禁止在订阅服务器中进行架构更改。事务复制虽然不禁止更改，但更改可能会导致复制失败。

- 传播到重新发布订阅服务器的更改默认情况下会传播到它的订阅服务器。
- 如果架构更改引用存在于发布服务器中而不存在于订阅服务器中的对象或约束，则架构更改将在发布服务器中成功，但在订阅服务器上失败。
- 添加外键时引用的订阅服务器中的所有对象必须与发布服务器中的相应对象具有相同的名称和所有者。
- 不会复制显式添加、删除或更改索引，并且涉及显式索引的任何更改都需要在每个副本集上单独运行。支持为约束（如主键约束）隐式创建的索引。
- 不支持更改或删除由复制管理的标识列。有关自动管理标识列的详细信息，请参阅[复制标识列](#)。
- 不支持包含不确定性函数的架构更改，因为它们可能会导致发布服务器和订阅服务器中的数据不同（称为无法收敛）。例如，如果在发布服务器发出以下命令：ALTER TABLE SalesOrderDetail ADD OrderDate DATETIME DEFAULT GETDATE()，则将此命令复制到订阅服务器并执行时，会得到不同的值。有关不确定性函数的详细信息，请参阅 [Deterministic and Nondeterministic Functions](#)。
- 建议显式命名约束。如果不显式命名约束，SQL Server 将为约束生成名称，并且这些名称在发布服务器和每个订阅服务器中都不同。这在复制架构更改时会引发问题。例如，如果在发布服务器中删除一列并删除了一个相关约束，则复制将尝试在订阅服务器中删除该约束。由于约束名称不同，因此在订阅服务器上的删除将失败。如果由于约束命名问题而导致同步失败，可以在订阅服务器中手动删除约束然后重新运行合并代理。
- 如果为复制发布了一个表，并且已生成了发布快照，则无法将该表中的列更改为 XML 数据类型。若要更改列，必须先删除复制。
- 对已发布的表执行 DDL 时，"未提交读"不是受支持的隔离级别。
- 不应使用SET CONTEXT_INFO 来修改已对发布的对象执行架构更改的事务的上下文。

[回到顶部](#)

【6】业务场景

【6.1】跨机房、跨网段复制

<http://www.cnblogs.com/gaizai/p/3328511.html>

【6.2】跨机房、跨网段FTP复制

<http://www.cnblogs.com/gaizai/p/3337434.html>

【6.3】通过备份文件初始化复制

<http://www.cnblogs.com/gaizai/p/3309567.html>

[回到顶部](#)

【注意事项与疑问】

(1) 注意事项

1. 在SQL SERVER下实现发布服务器和订阅服务器的通信正常（即可以互访），打开1433端口，在防火墙中设置入站规则；
2. 发布服务器与订阅服务器的SQL Server Agent代理帐号必须设置的一样，否则不能互访；
3. 如果你希望在复制的过程中一并复制非聚集索引，可以对发布属性-项目进行如下设置，修改完之后需要重新生成快照；
- 4.使用请求订阅，分发作业是在订阅服务器上创建的；使用推送订阅，分发作业是在分发服务器上创建；
5. 分发服务器上的快照文件会删除，在所有订阅服务器应用快照后，复制清理将自动删除初始快照的关联bcp文件

6.在分发服务器和订阅服务器上设置别名的时候，别名应该跟服务器的实例名要一致，不然会报下面的错误：

(2) 疑问

1. SQL Server 只有在完整日志模式下才能使用复制嘛？

解惑：在简单模式下一样可以使用复制；

2. 如果是跨网段（跨机房）的发布与订阅，有没办法实现？需要注意什么？

解惑：可以通过修改host文件、别名的方式搭建复制，请参考：[SQL Server跨网段（跨机房）复制](#)

3. 如果说上面的情况可以在host设置，但是如果有端口映射的，host也无法设置吧？

解惑：在SQL Server配置管理器里建立别名，同样可参考：[SQL Server跨网段（跨机房）复制](#)

4. 订阅的形式可以选择推送订阅或者请求订阅，请求订阅降低分发服务器处理工作的开销，这个开销有多大呢？怎么计算影响？

解惑：只有在有很多订阅服务器的时候才比较明显，推送订阅与请求订阅更大的区别是在管理方面的不同；

[回到顶部](#)

【常见问题】

<https://docs.microsoft.com/zh-cn/sql/relational-databases/replication/administration/frequently-asked-questions-for-replication-administrators?view=sql-server-ver15>

活动在发布时是否需要在数据库上停止？

快照生成期间是否锁定表？

订阅何时可用？何时可以使用订阅数据库？

订阅何时可用？何时可以使用订阅数据库？

如果分发代理或合并代理启动时快照代理尚未完成，会出现什么情况？

是否应为复制配置编写脚本？

复制数据库需要什么样的恢复模式？

复制为什么要向已复制的表添加一列；如果表并未发布，是否会删除该列？

如何管理已发布表的约束？

如何管理标识列？

是否可以在不同的发布中发布相同的对象？

多个发布是否可以使用同一个分发数据库？

如何在分发服务器和发布服务器中查找信息，如已经发布了数据库中的哪些对象？

复制是否加密数据？

如何在 Internet 上复制数据？

如果删除连接，复制是否可以继续？

复制是否可以在低带宽连接上工作？ 它是否使用压缩？

复制是否可以在低带宽连接上工作？ 它是否使用压缩？

如果使用 IP 地址连接到服务器，是否可以配置复制？

是否复制登录名和密码？

什么是架构，如何复制架构？

如何配置订阅数据库上的授权，以使其与发布数据库上的授权相匹配？

如果重新初始化订阅，订阅数据库中授予的权限会怎样？

为什么无法对已发布的表运行 TRUNCATE TABLE？

在复制的数据库上运行大容量插入命令，会产生什么效果？

对于备份和还原，是否存在复制注意事项？

复制是否影响事务日志的大小？

如何在复制的数据库中重新生成索引或表？

如何在发布数据库和订阅数据库上添加或更改索引？

如何对复制所涉及的数据库的文件进行移动和重命名？

如何删除正在复制的表？

如何在已发布的表中添加或删除列？

如何确定订阅服务器的数据是否与发布服务器的数据同步？

如何向现有发布添加表？

如何从发布中删除表？

哪些操作需要重新初始化订阅？

什么操作会导致快照失效？

如何删除复制？

如何确定是否存在要复制的事务或行？

分发代理滞后多少？ 是否应重新初始化？

分发代理滞后多少？ 是否应重新初始化？

复制是否可与日志传送和数据库镜像一起进行？

复制是否可与聚类分析一起进行？

[回到顶部](#)

【故障排查】

(0) 参考文档:

<https://www.cnblogs.com/gered/p/9831781.html>

(1) 由于出现操作系统错误 3，进程无法读取文件

上次刷新: 2020/10/13 16:53:22

发布服务器到分发服务器的历史记录 分发服务器到订阅服务器的历史记录 未分发的命令

查看(V): 最后 100 个同步操作

分发代理的会话(S):

| | 状态 | 开始时间 | 结束时间 | 持续时间 | 错误消息 |
|--|----|---------------------|---------------------|----------|-----------------------|
| | 错误 | 2020/10/13 16:52:53 | 2020/10/13 16:52:55 | 00:00:02 | 由于出现操作系统错误 5，进程无法读... |
| | 错误 | 2020/10/13 16:51:48 | 2020/10/13 16:51:48 | 00:00:00 | 由于出现操作系统错误 5，进程无法读... |
| | 错误 | 2020/10/13 16:50:43 | 2020/10/13 16:50:43 | 00:00:00 | 由于出现操作系统错误 5，进程无法读... |

所选会话中的操作(I):

| 操作消息 | 操作时间 |
|--|---------------------|
| 由于出现操作系统错误 5，进程无法读取文件 “\\WIN-ST1AC985Q23\repl_share\unc\WIN-ST1... | 2020/10/13 16:52:55 |
| 初始化 | 2020/10/13 16:52:53 |

所选会话的错误详细信息或消息(E):

尝试的命令:
\\WIN-ST1AC985Q23\repl_share\unc\WIN-ST1AC985Q23 REPL REPL TEST1\20201013162446\test3 4.pre
(事务序列号: 0x0000002900004B6E002100000000, 命令 ID: 7)

错误消息:
• 由于出现操作系统错误 5，进程无法读取文件 “\\WIN-ST1AC985Q23\repl_share\unc\WIN-ST1AC985Q23 REPL REPL TEST1\20201013162446\test3 4.pre”。(源: MSSQL_REPL, 错误号: MSSQL_REPL20024)
获取帮助: http://help/MSSQL_REPL20024

解决:

推送订阅: 快照文件夹 给予 sql server代理启动账户访问权限

请求订阅: 发布服务器与订阅服务器的SQL Server Agent代理帐号必须设置的一样, 且对快照unc路径共享文件夹有权限, 否则不能互访;

核心原理

就是订阅服务器读取不到快照文件夹位置.这也导致了"订阅服务器无法重新初始化非immediate_sync发布的订阅".

可能的原因较多, 1、没设置共享或者没共享给跟订阅服务器共同的账户权限; 2、IPC连接被关闭, 订阅服务器无法连接到发布服务器共享资源。

解决办法:

可以将快照文件夹D:\XX复制到订阅服务器F:\XX

某个订阅属性-》快照位置-》"备用文件夹", 快照文件夹-》F:\XX。应该就可以了。

1.首先, 在创建发布时需要指定一个发布快照位置, 并且会提示你订阅服务器可能无法访问该目录 (不知你有没有注意), 每次生成快照都会将快照文件放在该目录下。

2.其次, 在创建订阅时, 需要你指定快照位置, 一般为发布缺省位置 (不知你这次注意没有)

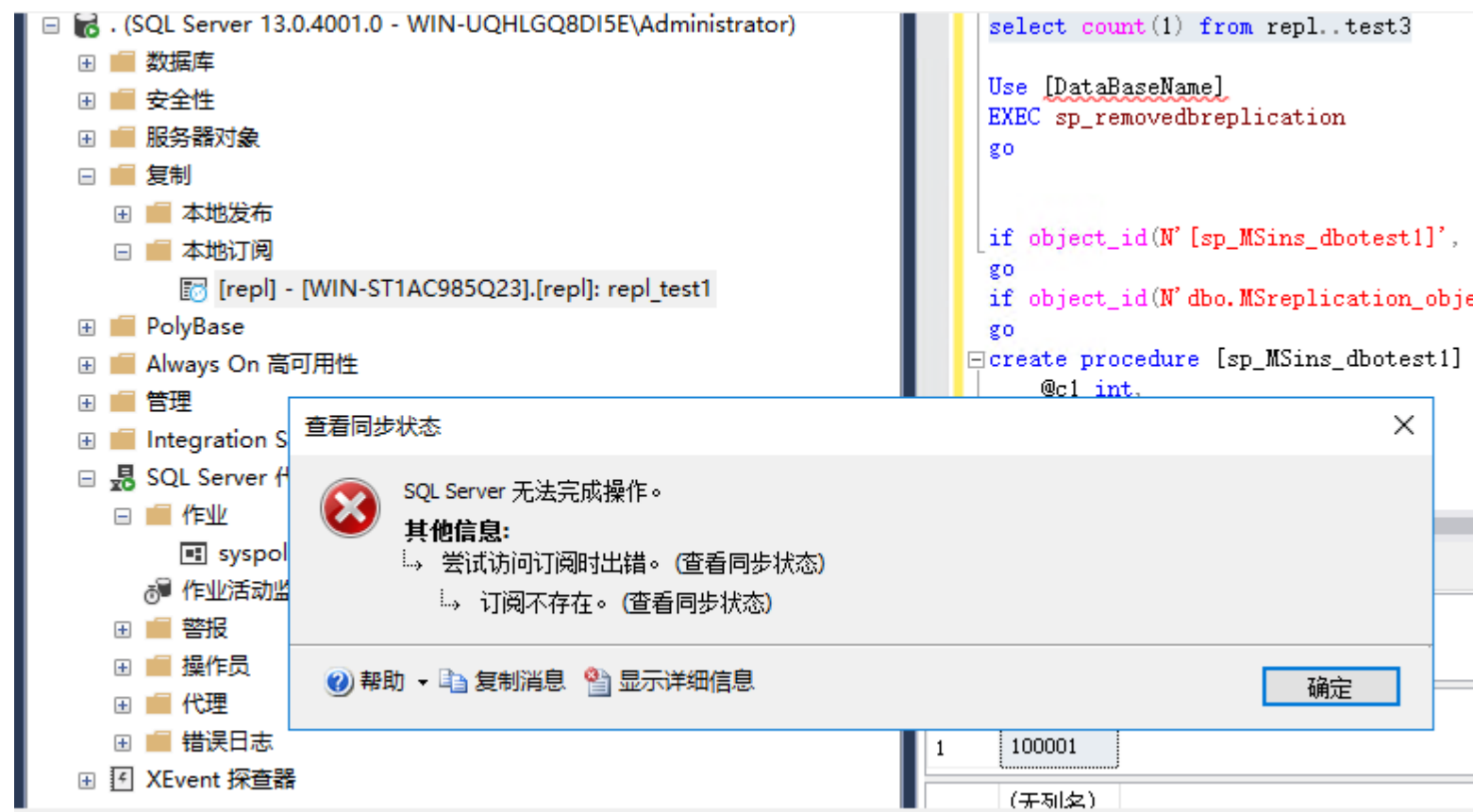
3.那么, 要解决你的问题只有两种方式

1)第一种: 在创建发布和订阅时指定一个两个服务器都能访问的目录。 (上面案例就是此步遇到障碍)

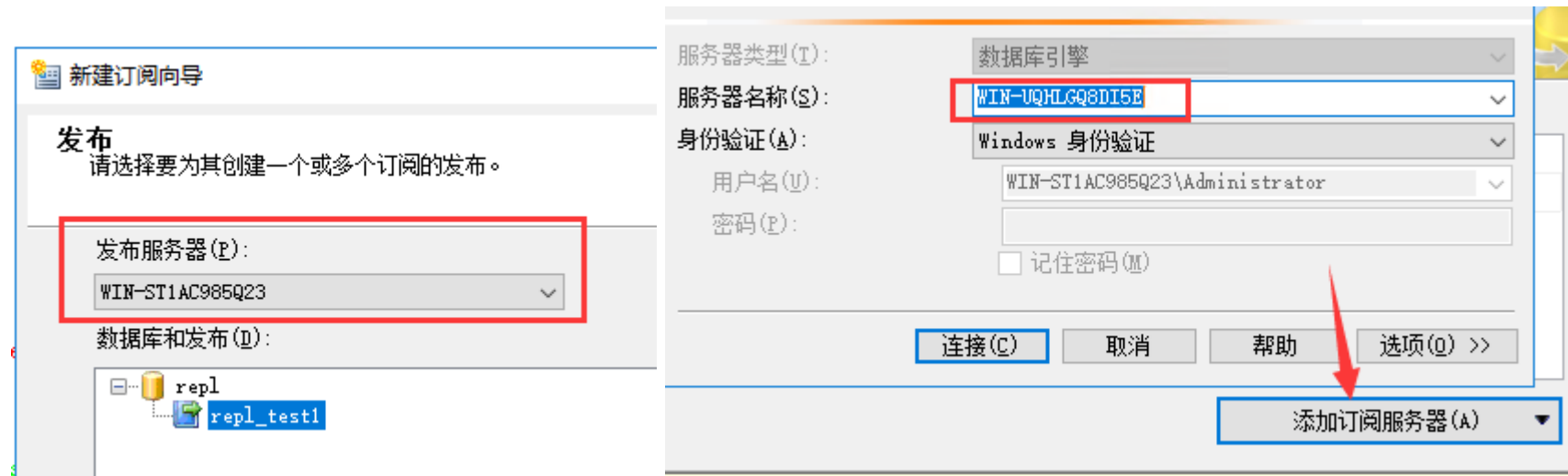
2)第二种: 每次自己将发布快照目录中的文件通过网络 (或移动设备) 拷贝到订阅指定的快照目录中。 (上面解决办法就用该种)

(2) 订阅不存在

明明有, 却提示订阅不存在, 且数据同步也是正常的, 但是就是查看的时候报错;

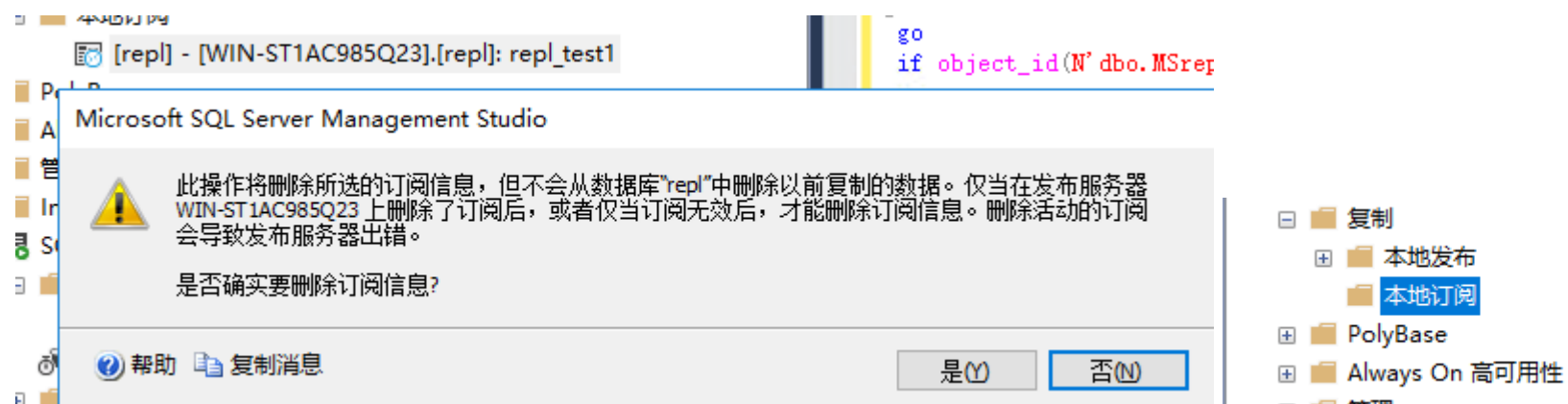


解决: 在配置订阅的时候, 选择发布服务器、订阅服务器的时候, 一定要用主机名, 否则会出现这个情况; 如下图:



(3) 找不到存储的过程 sp_MSins_tablename (误删订阅服务器上的订阅)

我在订阅服务器，不小心手动删掉了订阅，但主库的订阅居然没有被删掉。



我们再看看主库，依然有这个



错误信息：

发布服务器到分发服务器的历史记录

分发服务器到订阅服务器的历史记录

未分发的命令

查看(Y):

最后 100 个同步操作

分发代理的会话(S):

| | 状态 | 开始时间 | 结束时间 | 持续时间 | 错误消息 |
|--|------|-------------------|-------------------|----------|-------------------------------|
| | 正在运行 | 2015/1/4 14:55:03 | - | 00:01:59 | |
| | 错误 | 2015/1/4 14:50:38 | 2015/1/4 14:51:53 | 00:01:15 | 找不到存储过程 'dbo.sp_MSupd_dboSy.. |
| | 错误 | 2015/1/4 12:14:35 | 2015/1/4 14:49:44 | 02:35:09 | 找不到存储过程 'dbo.sp_MSupd_dboSy.. |
| | 已完成 | 2015/1/4 12:06:09 | 2015/1/4 12:13:21 | 00:07:12 | |

所选会话中的操作(I):

| 操作消息 | 操作时间 |
|-------------------------------|-------------------|
| 找不到存储过程 'dbo.sp_MSupd_dboSy.. | 2015/1/4 14:51:53 |
| 执行一批命令时出错。正在重试单个命令。 | 2015/1/4 14:51:53 |
| 初始化 | 2015/1/4 14:50:38 |

rep

上次刷新: 2020/10/13 19:11:33

发布服务器到分发服务器的历史记录

分发服务器到订阅服务器的历史记录

未分发的命令

查看(Y):

最后 100 个同步操作

分发代理的会话(S):

| | 状态 | 开始时间 | 结束时间 | 持续时间 | 错误消息 |
|---|----|---------------------|---------------------|----------|---------------------|
| ✖ | 错误 | 2020/10/13 19:11:33 | 2020/10/13 19:16:39 | 00:05:06 | 对象名 'dbo.test1' 无效。 |

所选会话中的操作(T):

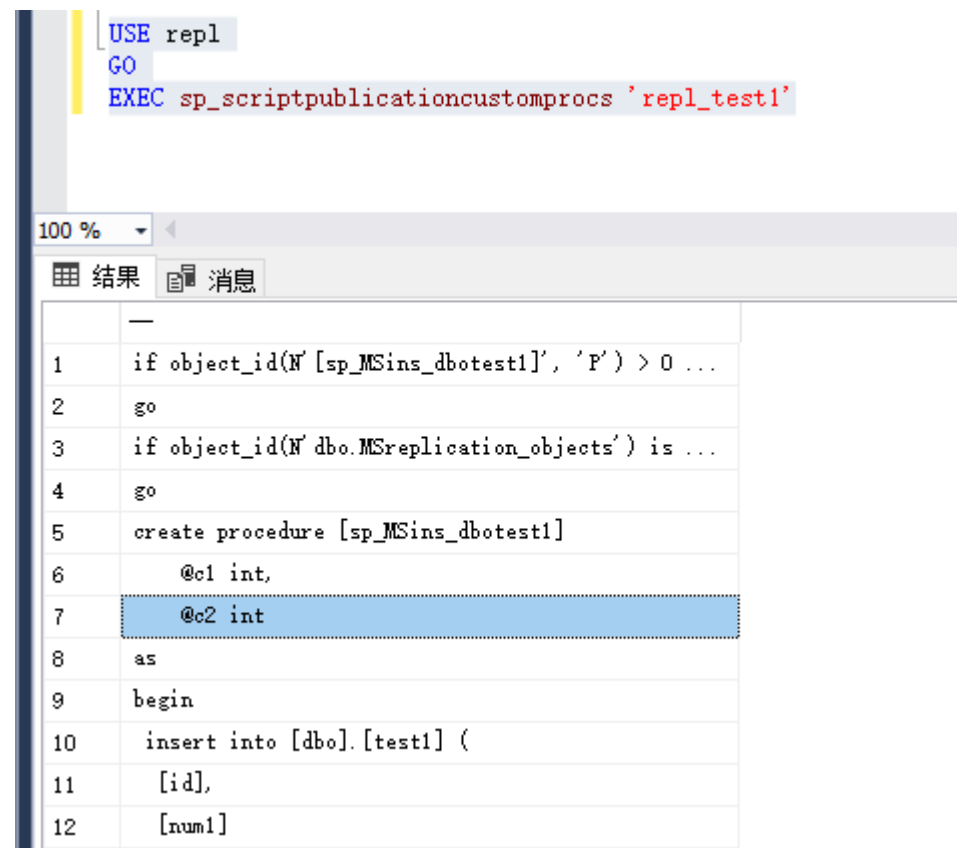
| 操作消息 | 操作时间 |
|---|---------------------|
| 对象名 'dbo.test1' 无效。 | 2020/10/13 19:16:39 |
| 执行一批命令时出错。正在重试单个命令。 | 2020/10/13 19:16:39 |
| 已在 1453 毫秒内将快照从 "unc\WIN-ST1AC985Q23_REPL_REPL_TEST1\20201013172432\" 子文件夹中传... | 2020/10/13 19:11:38 |
| 正在从 "[unc\WIN-ST1AC985Q23_REPL_REPL_TEST1\20201013172432\" 子文件夹中传... | 2020/10/13 19:11:38 |

解决：在发布数据库上运行

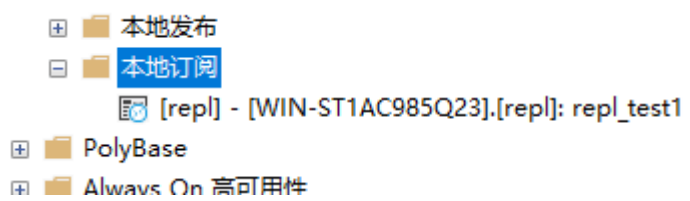
USE [发布数据库]

GO

EXEC sp_scriptpublicationcustomprocs '发布'



把结果集拿出来，放到有问题的订阅端上去执行；刷新一下订阅段的复制，发现订阅段的这个又出来了，如下图



再看主库复制监视器，我们可以发现，他其实是重新构建了一次，重新初始化了；

| | | | | | |
|---|------|---------------------|---------------------|----------|---------------------|
| 上次刷新: 2020/10/13 19:17:41 | | | | | |
| 发布服务器到分发服务器的历史记录 分发服务器到订阅服务器的历史记录 未分发的命令 | | | | | |
| 查看(Y): 最后 100 个同步操作 | | | | | |
| 分发代理的会话(S): | | | | | |
| | 状态 | 开始时间 | 结束时间 | 持续时间 | 错误消息 |
| | 正在运行 | 2020/10/13 19:17:39 | - | 00:05:03 | |
| | 错误 | 2020/10/13 19:11:33 | 2020/10/13 19:16:39 | 00:05:06 | 对象名 'dbo.test1' 无效。 |
| 所选会话中的操作(I): | | | | | |
| 操作消息 | | 操作时间 | | | |
| 无复制的事务。 | | 2020/10/13 19:22:42 | | | |
| 已在 1609 毫秒内将快照从 “unc\WIN-ST1AC985Q23_REPL_REPL_TEST1\20201013172432\” 子文件夹中传... | | 2020/10/13 19:17:41 | | | |
| 正在表 “[dbo].[test3]” 上创建主键索引 | | 2020/10/13 19:17:41 | | | |
| 正在表 “[dbo].[test2]” 上创建主键索引 | | 2020/10/13 19:17:41 | | | |
| 正在表 “[dbo].[test1]” 上创建主键索引 | | 2020/10/13 19:17:41 | | | |
| 已将数据大容量复制到表 “test3” 中 (12833 行) | | 2020/10/13 19:17:41 | | | |
| 正在将数据大容量复制到表 “test3” | | 2020/10/13 19:17:41 | | | |
| 已将数据大容量复制到表 “test3” 中 (12431 行) | | 2020/10/13 19:17:41 | | | |

(4) 无法禁用服务器 xxxx 上的发布与分发



一次只能有一个日志读取器或日志相关过程

无需理会，好像报这个错，它也已经禁用删除发布和分发了。

如果真的要管的话，把代理作业中的 事务日志代理 和分发代理作业停止一下，就好了。

(5) 服务器名和主机名不一致



```
IF SERVERPROPERTY ('SERVERNAME') <> @@SERVERNAME

Print '服务器名和数据库实例名不同！请修改操作后重启！'

IF SERVERPROPERTY ('SERVERNAME') <> @@SERVERNAME
BEGIN
DECLARE @server SYSNAME
SET @server=@@SERVERNAME
EXEC sp_dropserver @server=@server
SET @server=CAST (SERVERPROPERTY ('SERVERNAME') AS SYSNAME)
EXEC sp_addserver @server=@server,@local='LOCAL'
END

--重启服务器，重启后再执行验证，看2个值是否一样
IF SERVERPROPERTY ('SERVERNAME') , @@SERVERNAME
```

(6) sql server复制需要实际的服务器名称才能连接到服务器



- (1) 情况1：必须要用主机名
- (2) 情况2：如果要用链接服务器来发布订阅，必须链接服务器名和对方IP对应的主机名相同！

(7) 强行删除订阅发布

```
--或者暴力卸载、强制卸载
exec sp_removedbreplication db_name -- 强制卸载该数据库相关的分发发布
exec sp_droppublication publish_name -- 删除发布
```

(8) 无法作为数据库主题执行，因为主体dbo不存在

sql server删除订阅发布失败,无法作为数据库主题执行，因为主体"dbo"不存在、无法模拟这种类型的主体，或您没有所需要的权限。

[回到顶部](#)

【参考文档】

参考联机丛书，复制的配置与使用：<https://docs.microsoft.com/zh-cn/sql/relational-databases/replication/tutorial-preparing-the-server-for-replication?view=sql-server-2017>

大菠萝博客系列：<https://www.cnblogs.com/diablol/p/4097373.html>

1、概念与搭建

- Step1: [SQL SERVER复制介绍](#)
- Step2: [SQL Server 复制事务发布](#)
- Step3: [SQL Server 通过备份文件初始化复制](#)
- Step4: [SQL Server 跨网段（跨机房）复制](#)

[Step5: SQL Server 跨网段（跨机房）FTP复制](#)

[Step6: SQL Server 数据变更时间戳（timestamp）在复制中的运用](#)

[Step7: SQL Server 多实例下的复制](#)

总结参考使用：<http://blog.itpub.net/30126024/viewspace-2639648/>

2、优化与使用

[SQL Server复制情况下的高可用方案（一）镜像+复制](#)

[SQL Server提高事务复制效率优化（四）修改数据同步过程优化](#)

[SQL Server提高事务复制效率优化（三）订阅初始化优化](#)

[SQL Server提高事务复制效率优化（二）快照初始化优化](#)

[SQL Server提高事务复制效率优化（一）总体概述](#)

[distribution数据库过大问题](#)

[【sql server复制】不重新初始化快照的情况下新增表/存储过程/函数等](#)

3、常见故障与问题

<https://www.cnblogs.com/gered/p/9042763.html>

<https://www.cnblogs.com/gered/p/9831781.html>

[【SQL Server复制】数据库复制：修改表结构、新增表、新增存储过程会被复制到订阅服务器？](#)

4、深入解析与优化

[使用复制存储过程执行解决"事务复制中的表大量更新导致无法及时同步"的问题（转）](#)

[SQL Server 事务复制分发到订阅同步慢 SqlServer 复制中将大事务分成小事务分发的方法](#)

[【推荐优先查看】如何处理SQL Server事务复制中的大事务操作](#)

[SQLSERVER监控复制并使用数据库邮件功能发告警邮件](#)

[回到顶部](#)

分类: [sql server 高可用](#)

好文要顶

关注我

收藏该文



郭大侠1

粉丝 - 238 关注 - 74

+加关注

« 上一篇: [Typora Markdown文本编辑器工具](#)

» 下一篇: [如何查看windows server是否有web发布?](#)

3

推荐

0

反对

posted @ 2020-10-14 19:22 [郭大侠1](#) 阅读(1628) 评论(2) [编辑](#) [收藏](#) [举报](#)

[会员救园](#)

[刷新页面](#) [返回顶部](#)

登录后才能查看或发表评论, 立即 [登录](#) 或者 [逛逛](#) 博客园首页

[【推荐】阿里云金秋云创季: 云服务器新秀99元/年, 百款产品满减折上折](#)

[【推荐】秋风送爽, 又是一年双11: 一文解读天翼云全民上云节玩法攻略](#)

[【推荐】会员救园: 园子走出困境的唯一希望, 到年底有多少会员](#)

编辑推荐:

- [记一次 .NET 某工厂无人车调度系统 线程爆高分析](#)
- [C++ 小练习: 字符串分割的高性能实现](#)
- [一个重量级 HTTP api 的 304 优化分析与突发失效问题解决](#)
- [.NET 中的数组在内存中如何布局?](#)
- [数据库系列: 前缀索引和索引长度的取舍](#)

阅读排行:

- [C#/.NET/.NET Core优秀项目和框架2023年10月简报](#)
- [NativeBuffering,一种高性能、零内存分配的序列化解决方案\[性能测试篇\]](#)
- [HarmonyOS UI 开发](#)
- [同城售后系统退款业务重构心得](#)
- [实现一个极简的字节数组对象池](#)