

Troubleshooting Transactional Replication

Chris Skorlinski

<http://blogs.msdn.com/CHRISSK>
<http://blogs.msdn.com/REPLTALK>



March 6, 2010

Agenda

How Transactional Replication Works

- LogReader Agent Reader and Writer Threads
- Distribution Agent Reader and Writer Threads



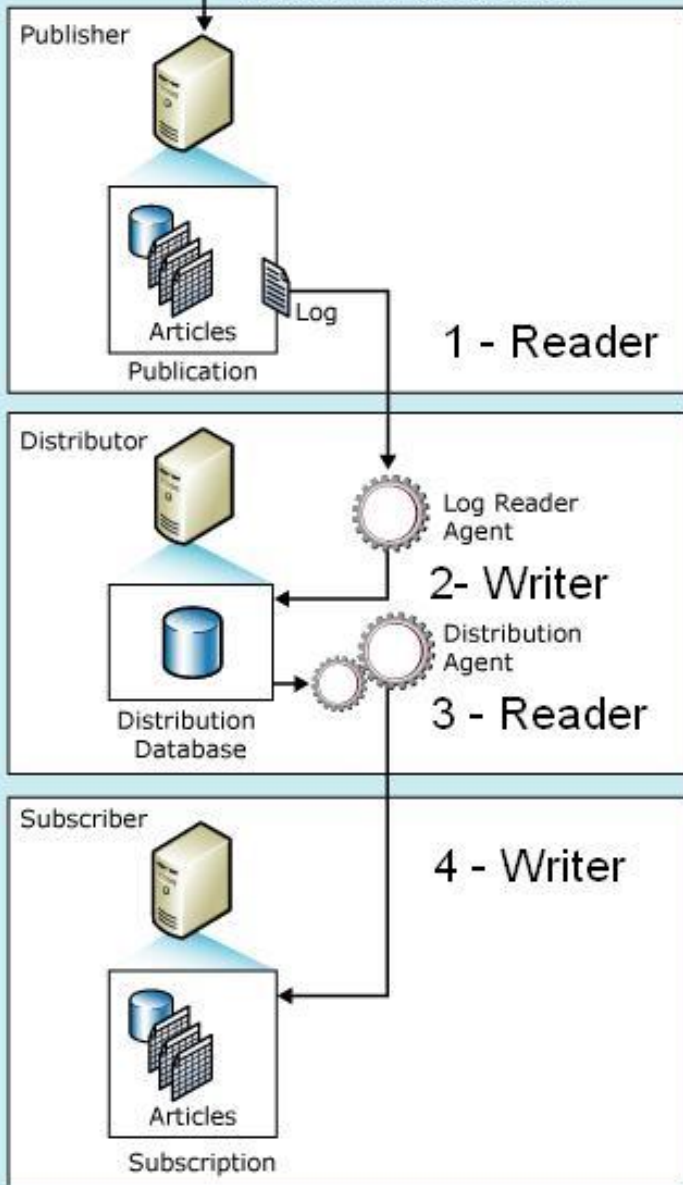
Tools to Monitor Replication

- Replication Monitor and Tracer Tokens
- Profile Trace and Perfmon
- Agent Stats



Key Take Aways for Transactional Replication

Transactional replication typically begins with the generation and application of the snapshot.

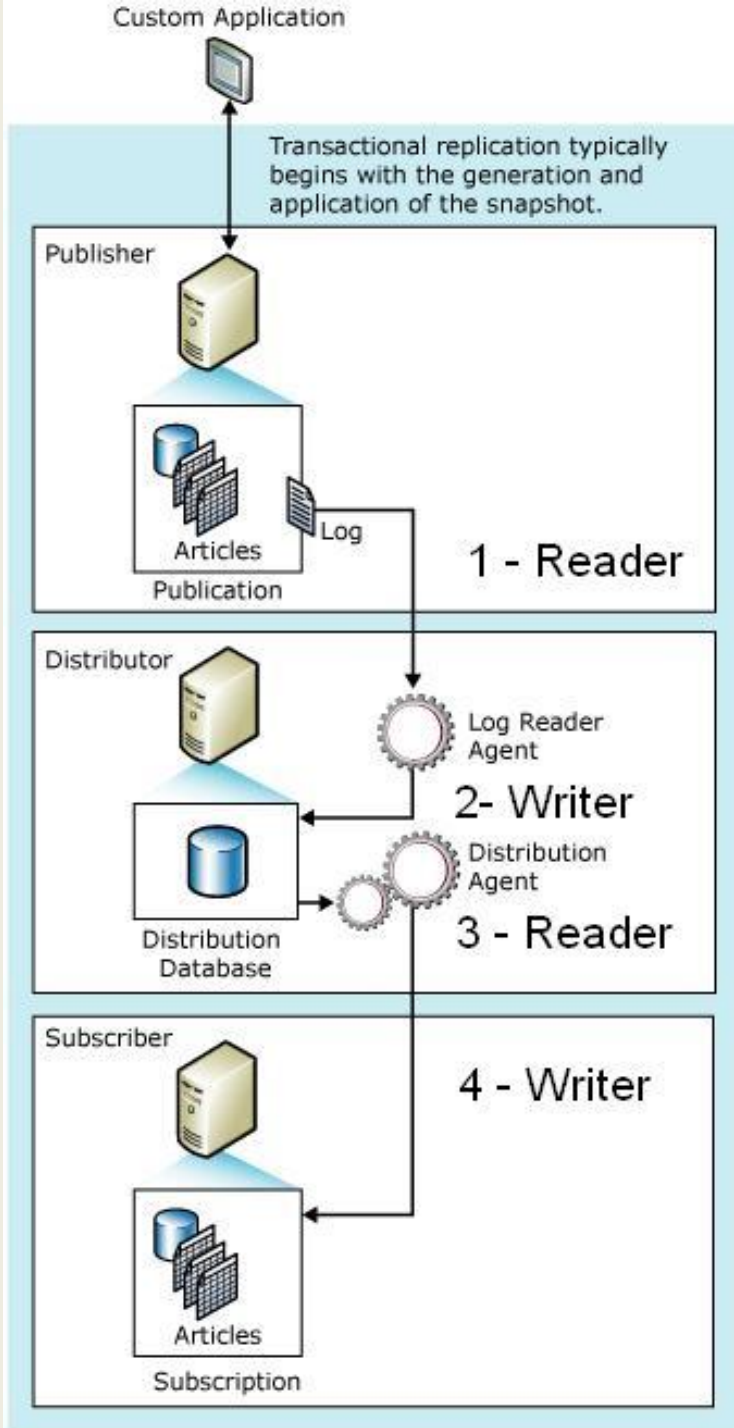


Log Reader Agent Internals (logread.exe) – Sequence of Steps

Calls **sp_MSadd_LogReader_History** to write to MSLogReader_History – “Starting Agent”
sp_MShelp_logreader_agentid – Obtain log reader agent specific information for that publication
sp_MShelp_profile – Obtains profile information for the log reader
MSadd_logreader_history to write MSlogreader_history – “Initializing”
sp_MSget_last_transaction – determine where the log reader agent left off reading the log.
 Read the transaction log – **sp_replcmds** (will cover in detail in next slide)
 Processes the returned commands from the **sp_replcmds** in batches of 27 commands by calling **sp_MSadd_repl_commands_27hp** once for each batch of 27 commands
 Marks this transaction as committed in distribution database by using **sp_repldone** procedure
 Adjusts the identity range if necessary and if you are using Automatic Identity Range Management y calling **sp_MSPub_adjust_identity**
 Calls **sp_MSget_last_transaction** to check the last transaction read and stored in **MSreplication_transactions** table
 When all transactions are read, LogRead.exe calls **sp_MSAdd_logreader_history** and writes message to MSLogReader_history “1 transactions with 9 commands were delivered”

Distribution Agent Internal - Sequence of Steps

master.db.sp_msget_jobstate – get the status of the job (if it is already started)
sp_Msadd_distribution_history – MSDistribution_history – Starting agent
sp_MSSubscription_Status – whether subscription has expired or the snapshot is ready
sp_server_info – determines the collation
sp_mshelp_subscriber_info – retrieve subscriber information
sp_mshelp_subscription_agentid – determine the name of the distribution agent
sp_Msadd_distribution_history – Initializing message – Msrepl_distribution_history
sp_Msadd_distribution_history – Connecting to Subscriber - Msrepl_distribution_history
so_datatype_info – to determine the data type mapping necessary to create the tracking table necessary for the Distribution agent
sp_MScheck_subscribe on subscription database – verifies that SQL Server Agent account is in sysadmin and db_owner role in subscription database
sp_mscreate_sub_tables on subscriber in subscription database – creates MSSubscription_agents and MSreplication_subscriptions tables
Sp_MSinit_Subscription_agent – updates the Subscription agent information on subscription database
 Retrieves transaction_timestamp and subscription_guid to determine what Distribution agent has already replicated to the Subscriber. **Transaction_timestamp** correlates to **xact_seqno** column in **MSreplication_transactions** table in distribution database. All values large than the xact_seqno will be replicated
 If we are doing initial sync, Distribution Agent calls **sp_MSupdatelastsyncinfo** which updates **MSreplication_subscriptions** and **MSSubscription_agents** table
 Starts to retrieve all transactions and their corresponding commands from **MSreplication_transactions** and **MSreplication_commands** table where transaction_timestamp column in subscription database < xact_seqno column in **MSreplication_transactions** table. Applies the transaction using **sp_MS_get_repl_commands** procedure
 Issues dynamic SQL to update the **MSreplication_subscriptions** table with the last delivered transaction ID
sp_MSdistribution_history to write the **MSrepl_distribution_history** table with status message “19 transaction(S) with 19 command(s) were delivered”



LogRead.exe

- 1) Reader
- 2) Writer

Distrib.exe

- 3) Reader
- 4) Writer

LogReader Latency

Slow Reader

- Agent History Stats
- SQLDiag, SQL Server Enterprise, ReadTrace.tblBatches, perfmon.msc
- Unexpected high sub volume?
- Is Server /Log. read in Restart? Large Non-Replicated transactions?
- How big is the Transaction Log?
- Ratio Replicated/Non-Replicated

Slow Writer

- Blocking at the Distribution Database
- How large is the Distribution Database?
- Reindex Replication System Tables
- How are the Cleanup Jobs running?
- We too wish drives were faster
- Sometimes it is the network

Demo Time

LOGREADER AGENT Statistics

***** STATISTICS SINCE AGENT STARTED

Execution time (ms): 3620955

Work time (ms): 19926

Distribute Repl Cmds Time(ms): 7997

Fetch time(ms): 15602

Repldone time(ms): 721

Write time(ms): 1201

Num Trans: 166 Num Trans/Sec: 8.330824

Num Cmds: 12808 Num Cmds/Sec: 642.778280

MSlogreader_history

```
<stats state="1" work="9" idle="295">  
  <reader fetch="8" wait="0"/>  
  <writer write="9" wait="0"/>  
  <sinclaststats elapsedtime="304" work="9" cmds="52596"  
cmdsperssec="5753.000000">  
    <reader fetch="8" wait="0"/>  
    <writer write="9" wait="0"/>  
  </sinclaststats>  
</stats>
```

STATE	Indication
1	Normal Log
2	Reader Thread has to WAIT for Writer
3	Writer Thread has to WAIT for Reader

Distribution Agent Latency

Slow Reader

- Agent History Stats
- SQLDiag, SQL Nexus, ReadTrace.tblBatches, performance counters, query plans? blocking?
- Unexpected replication volume?
- How big is the Distribution DB?

Slow Writer

- Blocking at Subscriber
- Subscriber Performance
- Subscriber Triggers?

Demo Time

Distribution Agent Statistics

***** STATISTICS SINCE AGENT STARTED *****

Total Run Time (ms) : 603349 Total Work Time : 20124
Total Num Trans : 38 Num Trans/Sec : 1.89
Total Num Cmds : 9192 Num Cmds/Sec : 456.77
Total Idle Time : **580185**

Writer Thread Stats

Total Number of Retries : 0
Time Spent on Exec : **8833**
Time Spent on Commits (ms): 15 Commits/Sec : 0.89
Time to Apply Cmds (ms) : 20124 Cmds/Sec : **456.77**
Time Cmd Queue Empty (ms) : **14006** Empty Q Waits > 10ms: 307
Total Time Request Blk(ms): 594191
P2P Work Time (ms) : 0 P2P Cmds Skipped : 0

Reader Thread Stats

Calls to Retrieve Cmds : 129
Time to Retrieve Cmds (ms): 16083 Cmds/Sec : **571.54**
Time Cmd Queue Full (ms) : **4555** Full Q Waits > 10ms : 41

Distribution Agent Statistics

***** STATISTICS SINCE AGENT STARTED *****

Total Run Time (ms) : 493135 Total Work Time : 305046
Total Num Trans : 17 Num Trans/Sec : 0.06
Total Num Cmds : 1238 Num Cmds/Sec : 4.06
Total Idle Time : **185049**

Writer Thread Stats

Total Number of Retries : 0
Time Spent on Exec : **303735**
Time Spent on Commits (ms): 16 Commits/Sec : 0.03
Time to Apply Cmds (ms) : 305046 Cmds/Sec : **4.06**
Time Cmd Queue Empty (ms) : **3523** Empty Q Waits > 10ms: 71
Total Time Request Blk(ms): 188572
P2P Work Time (ms) : 0 P2P Cmds Skipped : 0

Reader Thread Stats

Calls to Retrieve Cmds : 45
Time to Retrieve Cmds (ms): 3822 Cmds/Sec : **323.91**
Time Cmd Queue Full (ms) : **296010** Full Q Waits > 10ms : 9

MSdistribution_history

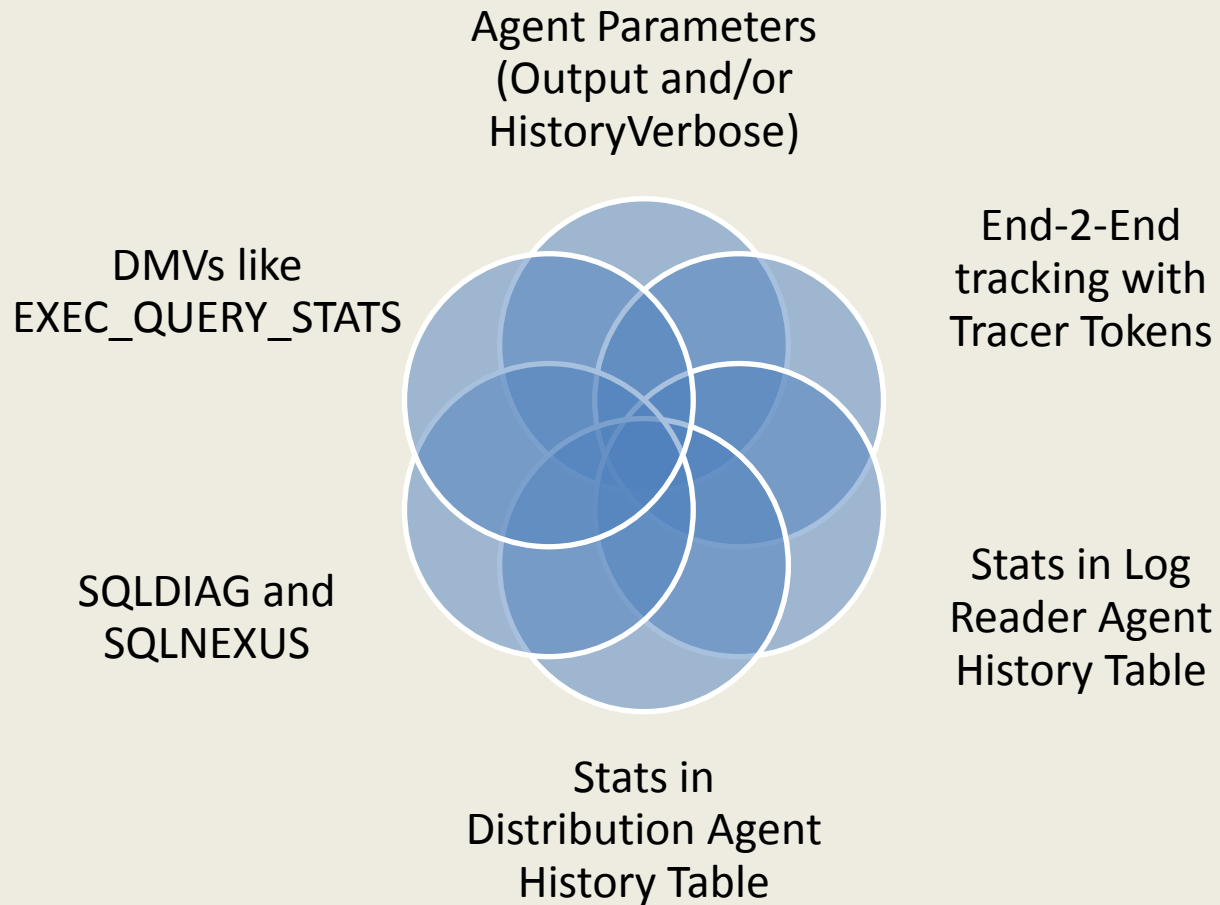
```
<stats state="1" work="14798" idle="2035">  
  <reader fetch="14798" wait="193"/>  
  <writer write="12373" wait="9888"/>  
  <sinclaststats elapsedtime="424" work="415" cmds="296900"  
cmdspersec="713.000000">  
    <reader fetch="415" wait="7"/>  
    <writer write="377" wait="212"/>  
  </sinclaststats>  
</stats>
```

READER Thread WAITED only 7 seconds for WRITER

WRITER Thread WAITED 212 seconds

Why is the WRITER waiting? Investigate performance of sp_MSget_repl_commands

Troubleshooting Tools



Transactional Replication

Key “Take Away”

Just 2 EXEs, few SPs and couple tracking table
Tune like you would for other application

Plan for when it Breaks
Script the Publication, Transaction Log Growth, Restore databases, Manually Merge Data

Many Tools and Techniques to analyze Replication Performance
DMVs, Profile Trace, Perfmon, Tracer Tokens, Agent Stats, 3rd Party Tools

The bigger the data the bigger the problems
Database Size, Data Volumes, Startup Time, Index Maintenance

BING

*"The more they over think the plumbing,
the easier it is to stop up the drain."*

<http://blogs.msdn.com/ReplTalk>

<http://blogs.msdn.com/CHRISSK>

