

# Troubleshooting Transactional Replication

Chris Skorlinski

<http://blogs.msdn.com/CHRIS SK>  
<http://blogs.msdn.com/REPLTALK>



March 6, 2010

# Agenda

## How Transactional Replication Works

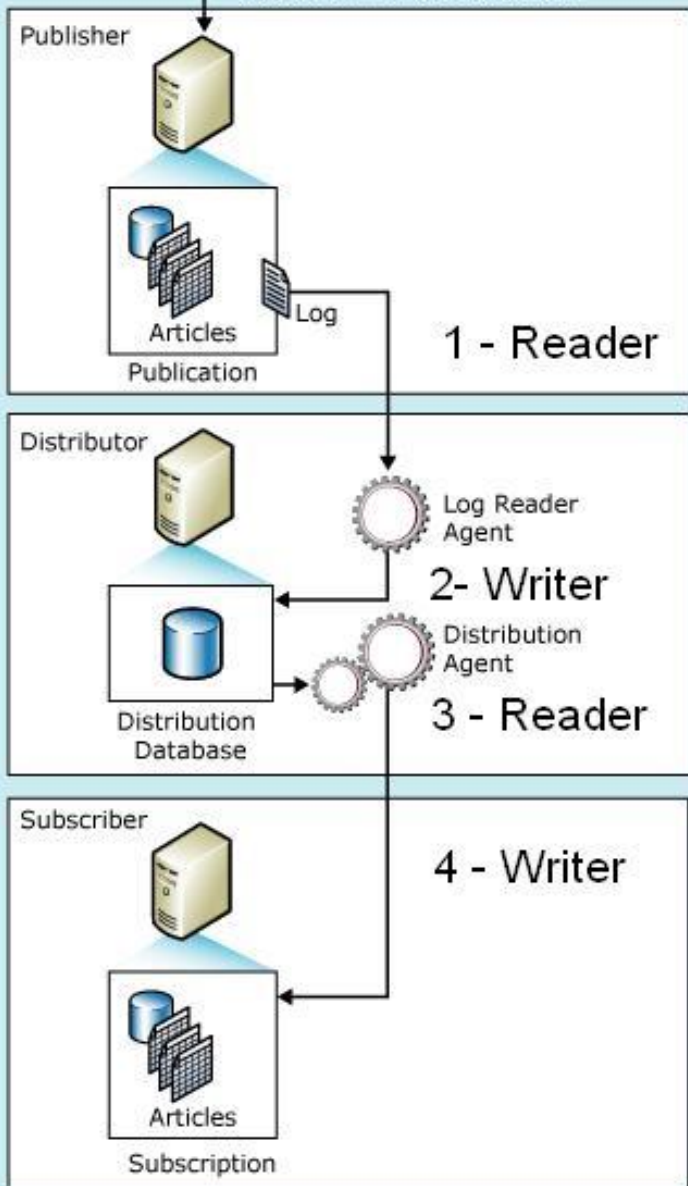
- LogReader Agent Reader and Writer Threads
- Distribution Agent Reader and Writer Threads

## 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.

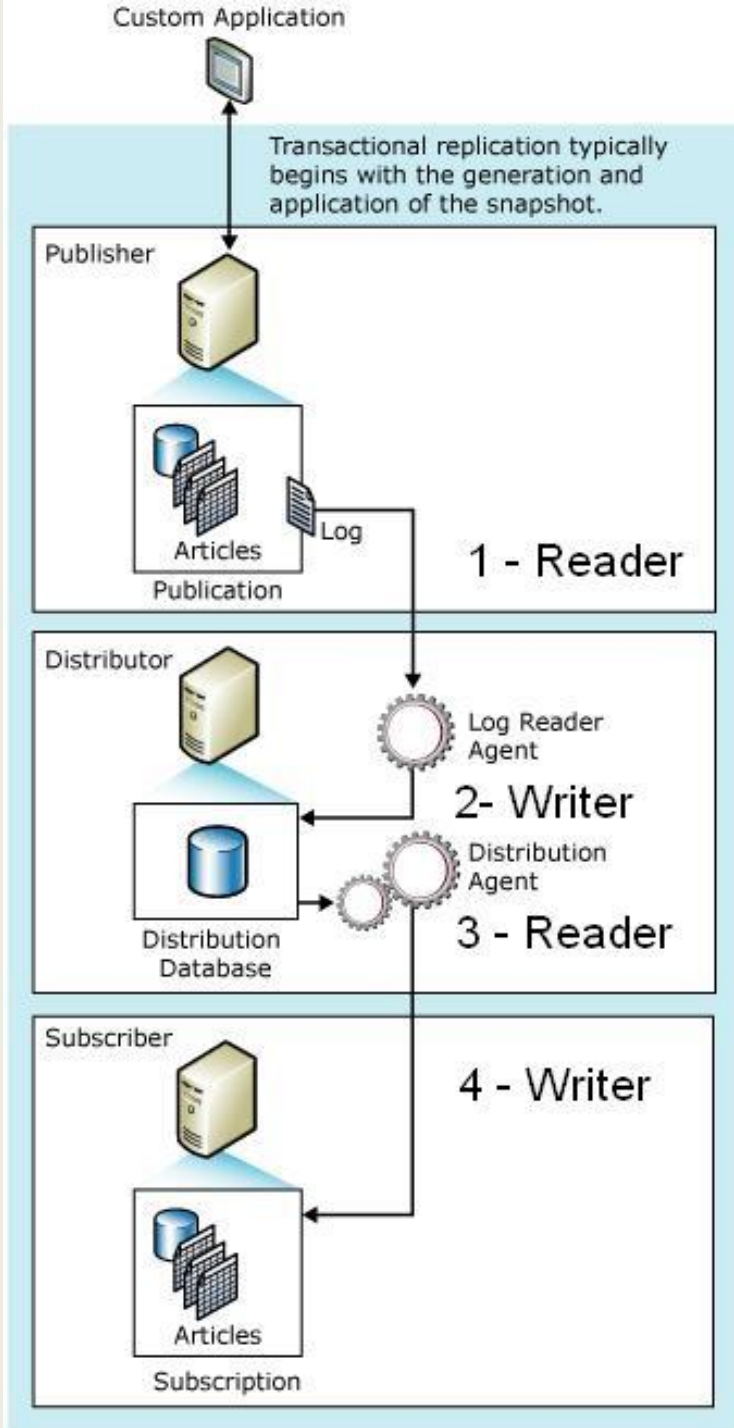


### Log Reader Agent Internals (logread.exe) – Sequence of Steps

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, SQLPerfmon, ReadTrace.tblBatches, perfmon.msc
- Unexpected high sub volume?
- Is Server /Log Reader restarted? 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, SQLNexus, 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

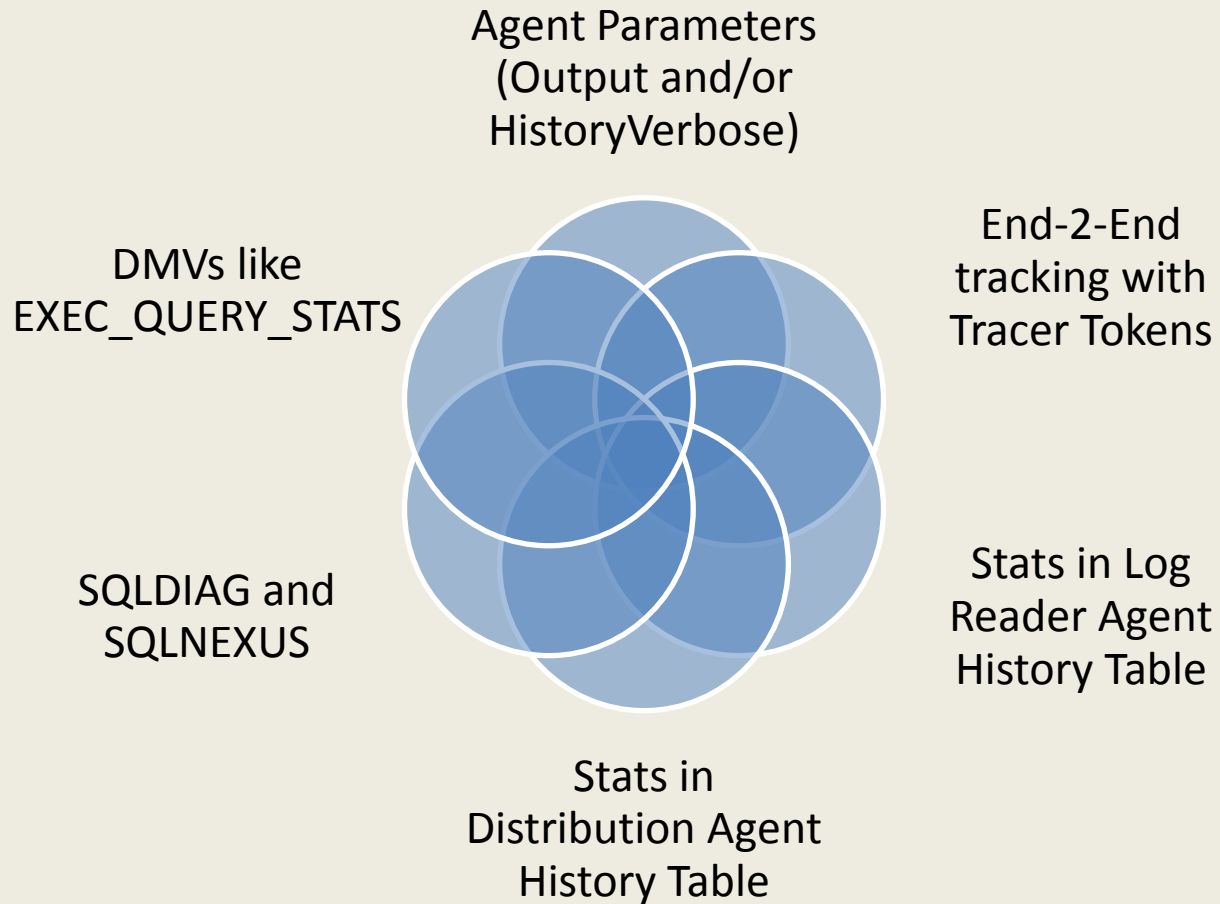
```
<stats state="1" work="14798" idle="2035">  
  <reader fetch="14798" wait="193"/>  
  <writer write="12373" wait="9888"/>  
  <sinclaststats elapsedtime="424" work="415" cmds="296900"  
cmdsperssec="713.000000">  
    <reader fetch="415" wait="7"/>  
    <writer write="377" wait="212"/>  
  </sinclaststats>  
</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, 3<sup>rd</sup> 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/CHRISSEK>

