

LSN DISTANCE MOVED EVERY HOUR (Measure of transaction logs size produced)

Last updated by | Sergiu Ernu | Mar 3, 2021 at 10:40 PM PST

Suppose these are the three current LSN's at these three hours.

PreciseTimeStamp	pg_current_wal_lsn	walFileName
2021-02-18 01:00:00	FDD/CF4120A0	0000000100000FDD000000CF
2021-02-18 02:00:00	FDD/E800AB88	0000000100000FDD000000E8
2021-02-18 03:00:00	FDE/F6BC1C88	0000000100000FDE000000F6

The number of wal files between 2nd and 1st hour is $E8 - CF = 19$

The number of wal files between 3rd and 2nd hour is $(FDE - FDD) * 255 + (F6 - E8) = 269$

We can clearly say that there was much more activity in the 2-3 hour compared to the 1-2 hour

We refer to this as the LSN Distance moved in an hour.

The below query calculates the LSN Distance moved every hour by a given server.

Over the hours and days if we see it is moving more distance every hour, then it means there is more write activity on the server.

Execute the highlighted portion of the kusto query to see the above calculations and comparison with an hour before.

The rest of the query is to show the LSNDistances for every hour, instead of every 5 mins.

All we need for this kusto query is the pg_current_wal_lsn logged every 5 mins or so.

```
//EastUS2
```

```
MonDmPgSqlXLogFileCount
```

```
| where LogicalServerName == "snipgsqldr01"
| extend index = indexof(pg_current_wal_lsn, '/', 0, -1, 1)
| extend offset = substring(pg_current_wal_lsn, index + 1)
| extend len = 8 - strlen(offset)
| extend walFileName = strcat('00000001', strrep('0', 8-index), substring(pg_current_wal_lsn,
0,index), '000000', strrep('0', len), substring(offset, 0, 2 - len))
| extend logicalXLogFile = strcat('0x', substring(walFileName, 8, 8))
| extend logicalXLogFileInt = toint(logicalXLogFile)
| extend xlogSegment = strcat('0x', substring(walFileName, 16, 8))
| extend xlogSegmentInt = toint(xlogSegment)
| extend walFileNumber = logicalXLogFileInt * 255 + xlogSegmentInt
| order by PreciseTimeStamp asc
| project LogicalServerName, PreciseTimeStamp, pg_current_wal_lsn, walFileName, offset,
logicalXLogFile, xlogSegment, logicalXLogFileInt, xlogSegmentInt, walFileNumber, prevWalFileNumber =
prev(walFileNumber, 12, 0), distance = walFileNumber - prev(walFileNumber, 12, 0)
| summarize maxWalFileNumberForHour = max(walFileNumber) by LogicalServerName, bin(PreciseTimeStamp,
1h)
| order by PreciseTimeStamp asc
| serialize PreciseTimeStamp, maxWalFileNumberForHour, prevMaxWalFileNumberForHour =
prev(maxWalFileNumberForHour, 1, 0), Distance = maxWalFileNumberForHour -
prev(maxWalFileNumberForHour, 1, 0)
| extend FinalDistance = iff(Distance > 10000, 0, Distance)
| project LogicalServerName, PreciseTimeStamp, FinalDistance
| render timechart
```

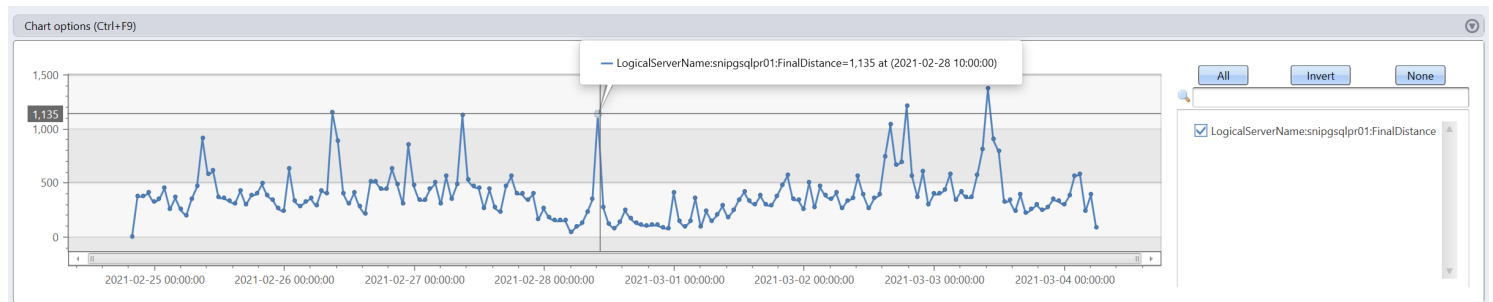
Note that you can do FinalDistance * 16MB to get the size of the transaction logs produced per hour.

The highlighted section produce the following output:

LogicalServerName	PreciseTimeStamp	pg_current_wal_lsn	walFileName	offset	logicalXLogFile	xlogSegment	logicalXLogFileInt	xlogSegmentInt	walFileNumber	prevWalFileNumber	distance
snipgsqlpr01	2021-02-25 12:03:25.1575816	10B8/BFC09E48	00000001000010B8000000BF	8FC09E48	0x000010B8	0x000000BF	4280	191	1091591	1091043	548
snipgsqlpr01	2021-02-25 12:08:36.3636253	10B8/E1172C48	00000001000010B8000000E1	E1172C48	0x000010B8	0x000000E1	4280	225	1091625	1091112	513
snipgsqlpr01	2021-02-25 12:13:47.1636479	10B9/18F694A8	00000001000010B900000018	18F694A8	0x000010B9	0x00000018	4281	24	1091679	1091157	522
snipgsqlpr01	2021-02-25 12:18:57.2447410	10B9/687B7AA0	00000001000010B900000068	687B7AA0	0x000010B9	0x00000068	4281	104	1091759	1091189	570
snipgsqlpr01	2021-02-25 12:24:21.6228467	10B9/7EBA0470	00000001000010B90000007E	7EBA0470	0x000010B9	0x0000007E	4281	126	1091781	1091229	552
snipgsqlpr01	2021-02-25 12:29:31.6565089	10B9/8F6C0C08	00000001000010B90000008F	8F6C0C08	0x000010B9	0x0000008F	4281	143	1091798	1091273	525
snipgsqlpr01	2021-02-25 12:34:42.0188656	10B9/A1E30E40	00000001000010B9000000A1	A1E30E40	0x000010B9	0x000000A1	4281	161	1091816	1091327	489
snipgsqlpr01	2021-02-25 12:39:52.0529727	10B9/B0779E58	00000001000010B9000000B0	B0779E58	0x000010B9	0x000000B0	4281	176	1091831	1091350	481
snipgsqlpr01	2021-02-25 12:45:02.4935346	10B9/C9A338D8	00000001000010B9000000C9	C9A338D8	0x000010B9	0x000000C9	4281	201	1091856	1091372	484
snipgsqlpr01	2021-02-25 12:50:12.5434066	10B9/DAD639C0	00000001000010B9000000DA	DAD639C0	0x000010B9	0x000000DA	4281	218	1091873	1091416	457
snipgsqlpr01	2021-02-25 12:55:22.9838492	10B9/EE61DA00	00000001000010B9000000EE	EE61DA00	0x000010B9	0x000000EE	4281	238	1091893	1091474	419
snipgsqlpr01	2021-02-25 13:00:33.3930428	10BA/1CE3DD8	00000001000010BA0000001C	1CE3DD8	0x000010BA	0x0000001C	4282	28	1091938	1091530	408
snipgsqlpr01	2021-02-25 13:05:43.4583069	10BA/32088400	00000001000010BA00000032	32088400	0x000010BA	0x00000032	4282	50	1091960	1091591	369
snipgsqlpr01	2021-02-25 13:10:53.8049726	10BA/41FF0628	00000001000010BA00000041	41FF0628	0x000010BA	0x00000041	4282	65	1091975	1091625	350
snipgsqlpr01	2021-02-25 13:16:04.0110299	10BA/62CB6628	00000001000010BA00000062	62CB6628	0x000010BA	0x00000062	4282	98	1092008	1091679	329

Done (00:04.072): 2052 records

The full Kusto query will have the following output:



As suggested to get the size in MB of the transactions logs produced per each hour we can modify the last 2 lines of the query:

```

MonDmPgSqlXLogFileCount
| where LogicalServerName == "snipgsqlpr01"
| extend index = indexof(pg_current_wal_lsn, '/', 0, -1, 1)
| extend offset = substring(pg_current_wal_lsn, index + 1)
| extend len = 8 - strlen(offset)
| extend walFileName = strcat('00000001', strrep('0', 8-index), substring(pg_current_wal_lsn, 0, index))
| extend logicalXLogFile = strcat('0x', substring(walFileName, 8, 8))
| extend logicalXLogFileInt = toint(logicalXLogFile)
| extend xlogSegment = strcat('0x', substring(walFileName, 16, 8))
| extend xlogSegmentInt = toint(xlogSegment)
| extend walFileNumber = logicalXLogFileInt * 255 + xlogSegmentInt
| order by PreciseTimeStamp asc
| project LogicalServerName, PreciseTimeStamp, pg_current_wal_lsn, walFileName, offset, logicalXLogFile
| summarize maxWalFileNumberForHour = max(walFileNumber) by LogicalServerName, bin(PreciseTimeStamp, 1h)
| order by PreciseTimeStamp asc
| serialize PreciseTimeStamp, maxWalFileNumberForHour, prevMaxWalFileNumberForHour = prev(maxWalFileNumberForHour)
| extend FinalDistance = iff(Distance > 10000, 0, Distance)
| project PreciseTimeStamp, FinalDistance, FinalDistanceMB=FinalDistance*16
| render timechart

```

We can see the size in MB of the transaction logs for each hour:

