

# Statement Calculation & Posting

---

## Performance Improvements



**Martin Kleindl**

Product Director

[Martin.Kleindl@LSRetail.com](mailto:Martin.Kleindl@LSRetail.com)

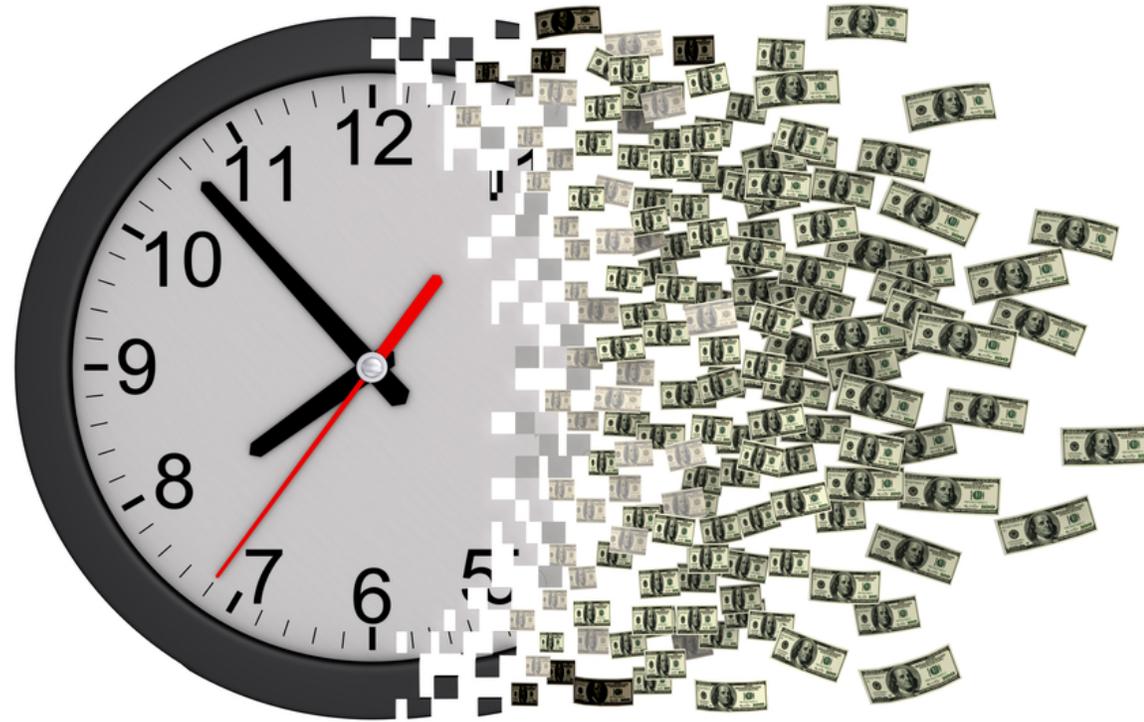


**LS Retail**

We make business easy

---

[www.LSRetail.com](http://www.LSRetail.com)



**Time = Money**



# Statement calculation & posting



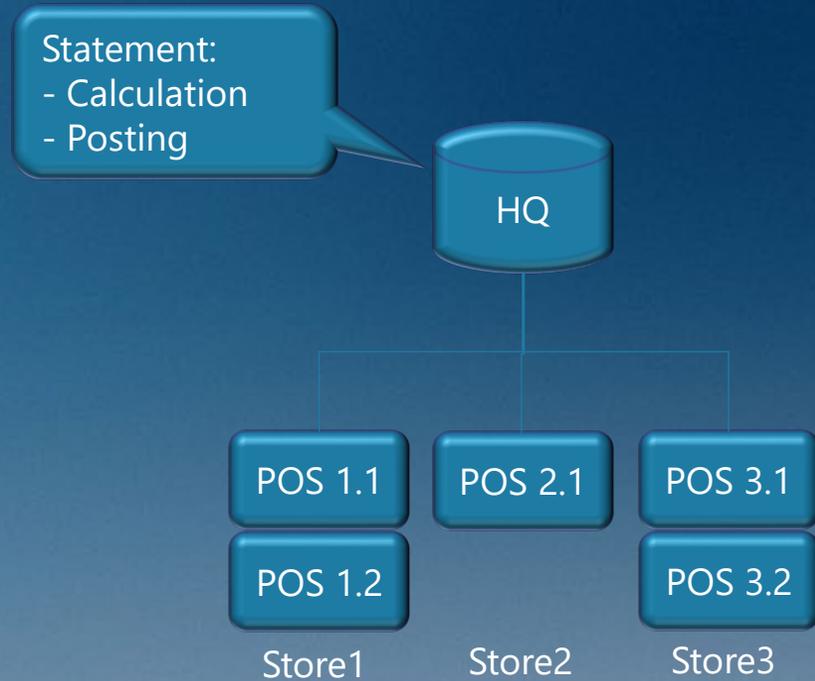
# Statement calculation & posting

Usually, this process runs smooth.

## Problems may arise when

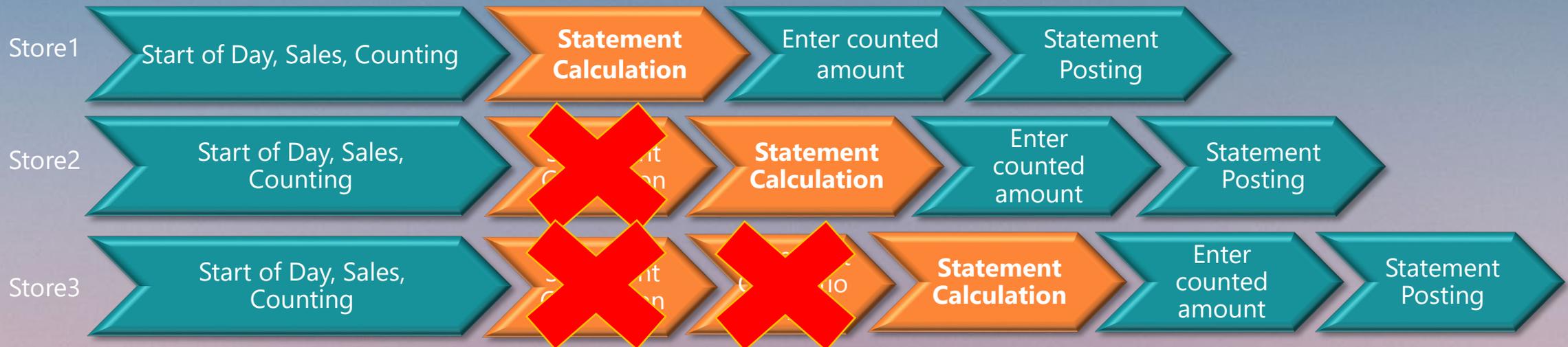
- Old/undersized hardware is used
- SQL Server not setup or maintained properly
- Retailers have a lot of transactions per day
- Retailers have a lot of stores that calculate and post Statements in parallel

# Statement calculation & posting



## Example:

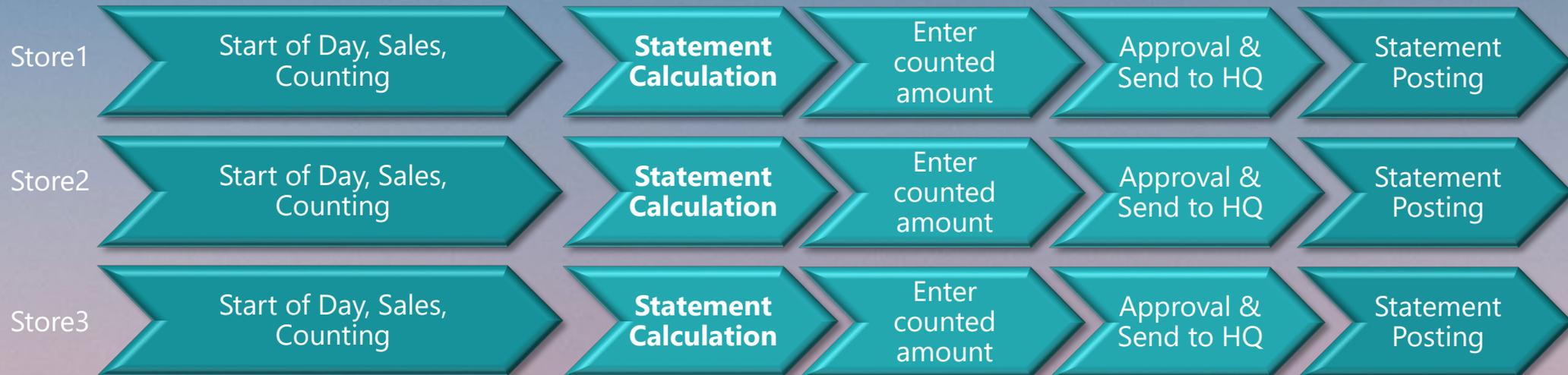
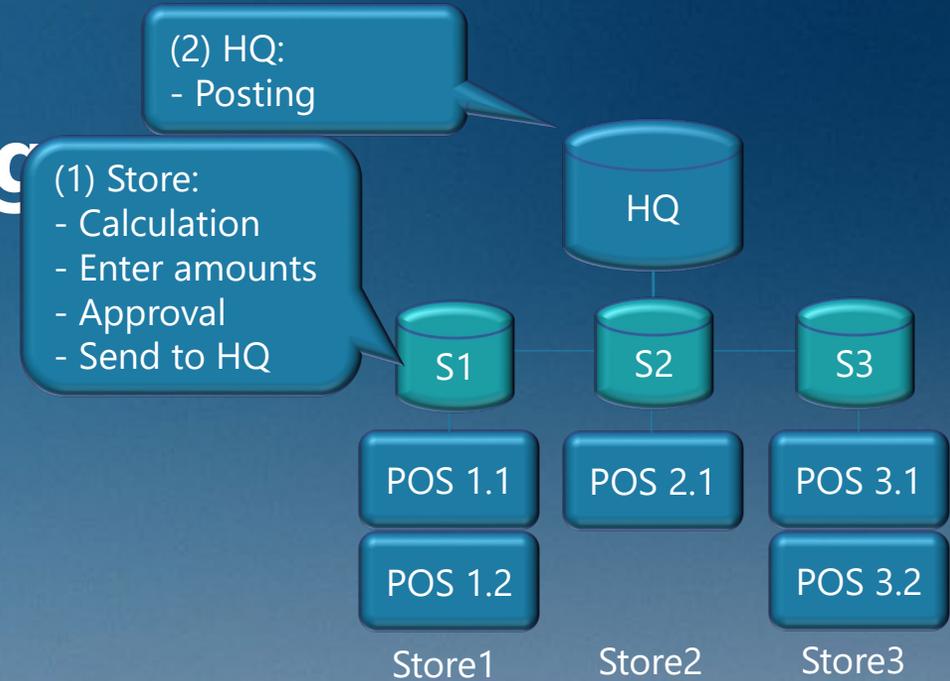
- 3 Stores
- Transactions have been replicated to the HQ database
- Statements are calculated in parallel in the HQ database by the Store users



# Statement calculation & posting

## Decentralize calculation:

- Introduce Store databases
- Replicate transactions to Store and HQ databases
- Calculate Statements in Store databases (in parallel)



# Statement calculation

---

## technical improvements

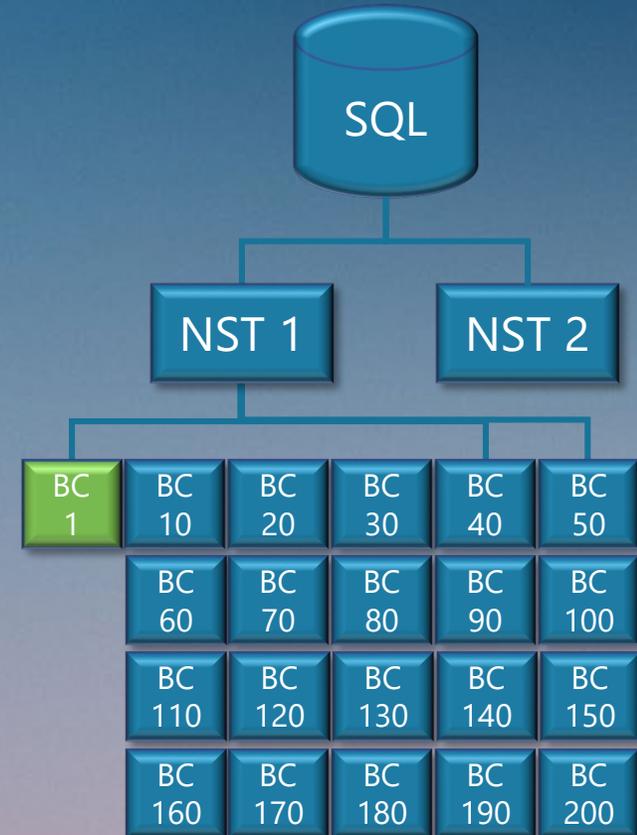
# Statement calculation

## Technical improvements

- Code optimization
- In-memory calculation (temp tables with flush back)
- Enable parallel calculation

## Original code

- 48 sec./Stmt.
- 1 parallel Stmt.



# Statement calculation

## Test environment:

- Database server  
Azure Standard D16s v3 (16 vCPUs, 64 GiB memory)  
Premium SSD
- 2 VMs for NST
- Test Database  
Data: 200 Stores, 20.000 items  
Transactions with multiple sales/day
- SQL Server Analysis scripts  
(from LS Retail's SQL Server Audit toolbox)
- SQL Server tools  
SQL Perform (Index tuning, missing index detection)

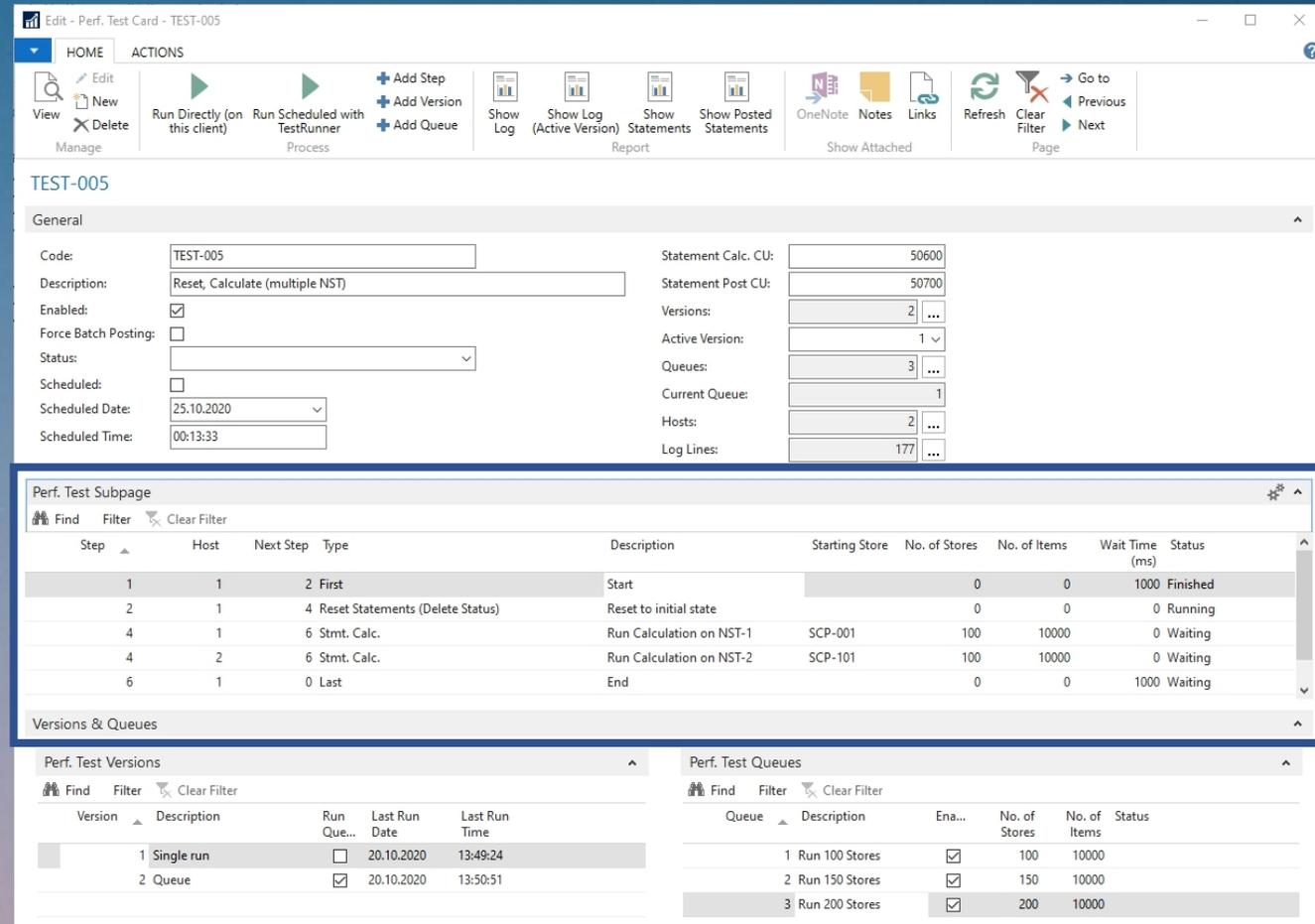
**LS Retail** Server Audit  
Toolbox

**SQL Perform**

# Statement calculation

## Test tool:

- Semi-automated test runs
- Multiple Service Tier support
- Test workflow
- Scale Tests  
Spin up background processes to test simultaneous calculation and posting (with STARTSESSION command)



**TEST-005**

**General**

Code: TEST-005  
 Description: Reset, Calculate (multiple NST)  
 Enabled:   
 Force Batch Posting:   
 Status:   
 Scheduled:   
 Scheduled Date: 25.10.2020  
 Scheduled Time: 00:13:33

Statement Calc. CU: 50600  
 Statement Post CU: 50700  
 Versions: 2  
 Active Version: 1  
 Queues: 3  
 Current Queue: 1  
 Hosts: 2  
 Log Lines: 177

**Perf. Test Subpage**

Step	Host	Next Step	Type	Description	Starting Store	No. of Stores	No. of Items	Wait Time (ms)	Status
1	1	2	First	Start		0	0	1000	Finished
2	1	4	Reset Statements (Delete Status)	Reset to initial state		0	0	0	Running
4	1	6	Strmt. Calc.	Run Calculation on NST-1	SCP-001	100	10000	0	Waiting
4	2	6	Strmt. Calc.	Run Calculation on NST-2	SCP-101	100	10000	0	Waiting
6	1	0	Last	End		0	0	1000	Waiting

**Perf. Test Versions**

Version	Description	Run Que...	Last Run Date	Last Run Time
1	Single run	<input type="checkbox"/>	20.10.2020	13:49:24
2	Queue	<input checked="" type="checkbox"/>	20.10.2020	13:50:51

**Perf. Test Queues**

Queue	Description	Ena...	No. of Stores	No. of Items	Status
1	Run 100 Stores	<input checked="" type="checkbox"/>	100	10000	
2	Run 150 Stores	<input checked="" type="checkbox"/>	150	10000	
3	Run 200 Stores	<input checked="" type="checkbox"/>	200	10000	

# Statement calculation

## Technical improvements

- Code optimization
- In-memory calculation (temp tables with flush back)
- Enable parallel calculation

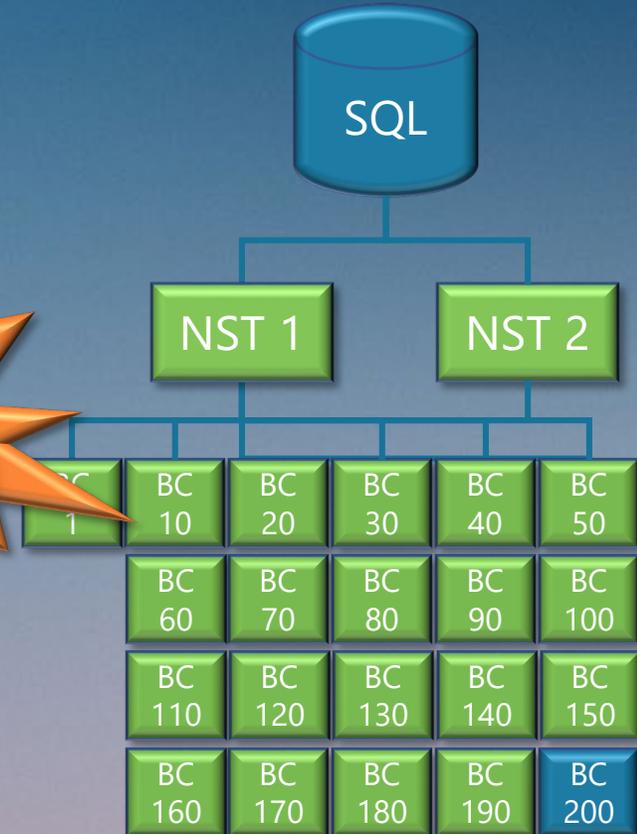
### Original code

- 48 sec./Stmt.
- 1 parallel Stmt.

### Improved code

- 14 sec./Stmt.
- 195 parallel Stmt.

2 NST reduce the calculation time by 14%



# Statement calculation

---

**new  
process**

# Statement calculation

## Incremental Statement Calculation

- Calculates and updates Statements during the day



Scheduled calculation and update of the Statements  
(for example every hour)

# Statement calculation

## Incremental Statement Calculation

- Calculates and updates Statements during the day
- At the end of the day, only the unprocessed transactions need to be calculated



“Only” adds Transactions created since last Statement calculation

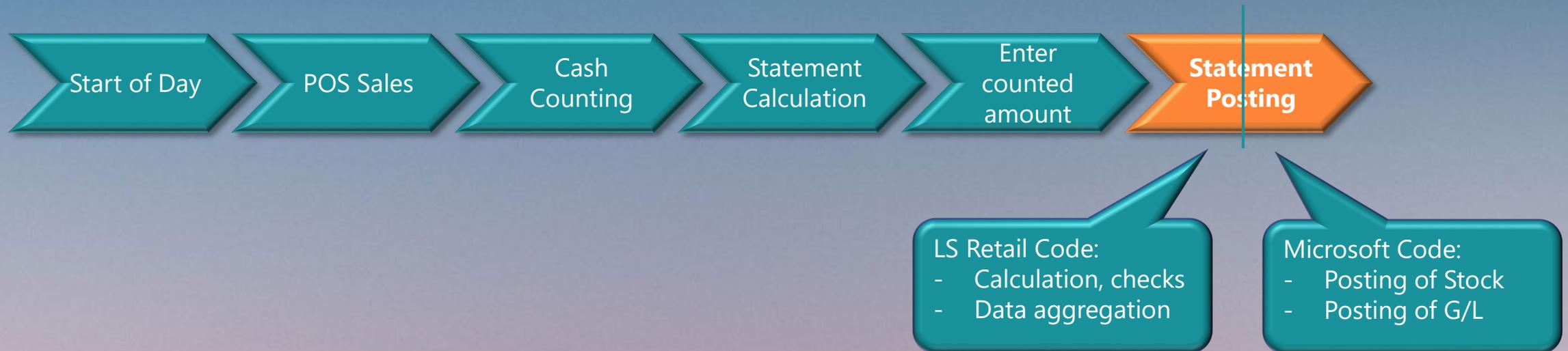
# Statement posting technical improvements

---

# Statement posting

## Technical improvements

- Code optimization
- In-memory calculation (temp tables with flush back)



# Statement posting

## Technical improvements

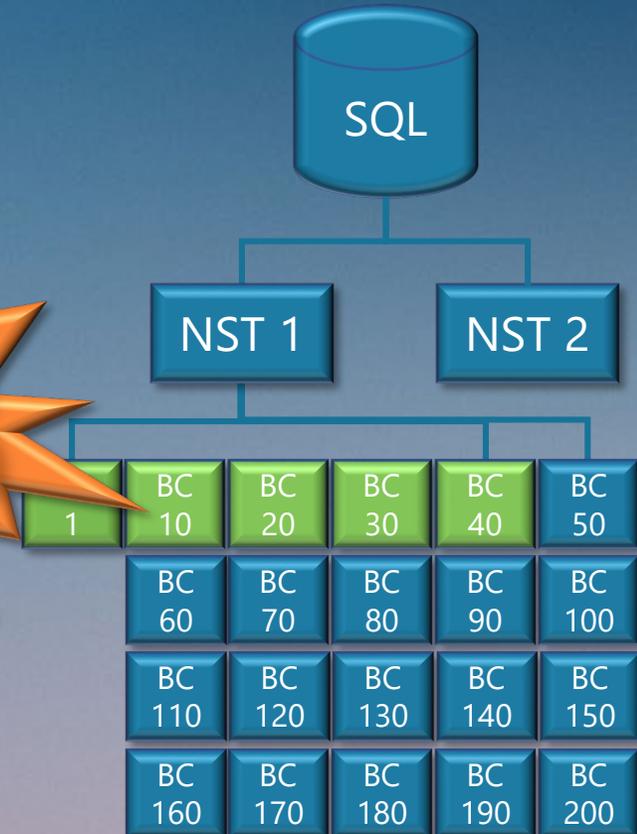
- Code optimization
- In-memory calculation (temp tables with flush back)

### Original code

- 80 sec./Stmt.
- 1 parallel Stmt.

### Improved code

- 50 sec./Stmt.
- 45 parallel Stmt.



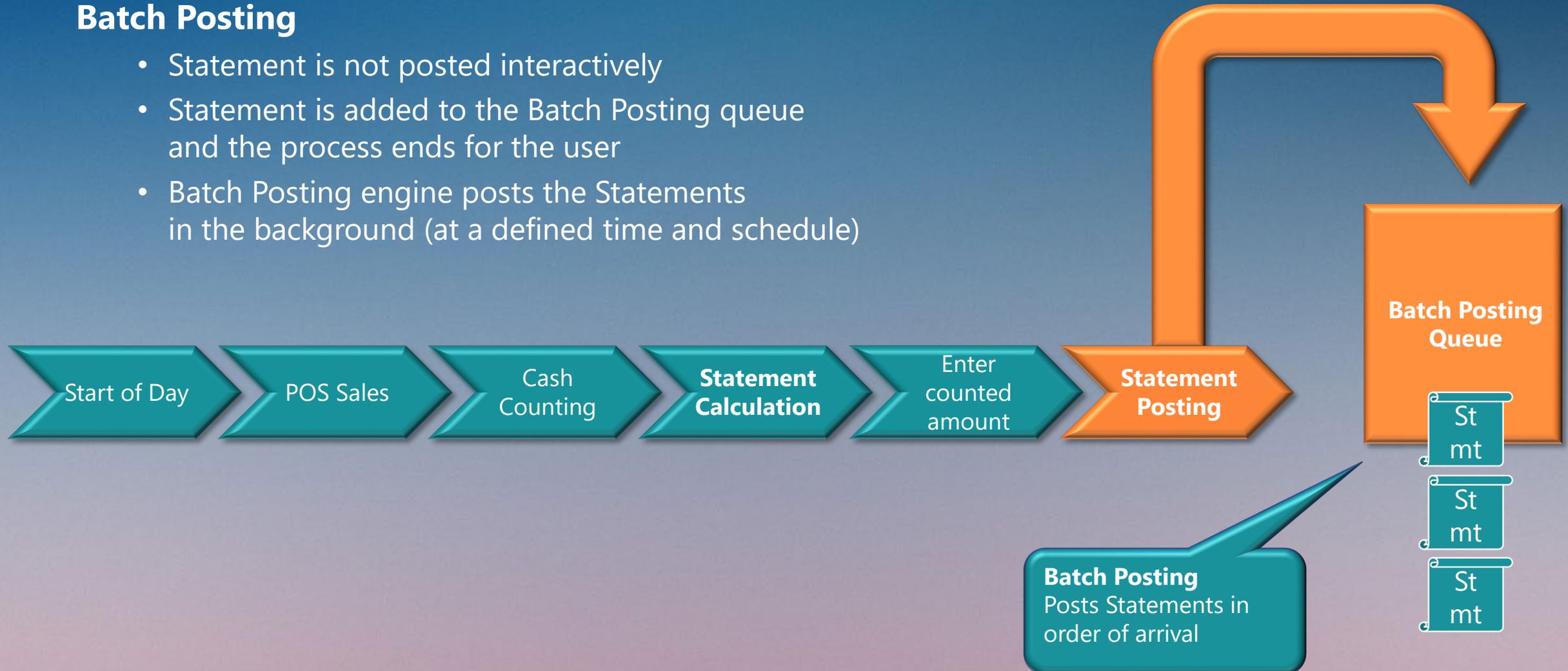
# Statement posting process improvements

---

# Statement posting

## Batch Posting

- Statement is not posted interactively
- Statement is added to the Batch Posting queue and the process ends for the user
- Batch Posting engine posts the Statements in the background (at a defined time and schedule)







# FamilyMart

- World's 2nd largest convenience store chain
- 40+ years of experience in convenience retailing
- Over 17,500 stores across various markets globally

## FamilyMart Malaysia

- 200+ stores, partially open 24 hours, high traffic
- In malls, at gas stations, separate stores
- Ecommerce mobile app, home delivery
- Convenience items, fresh food, hot snacks, coffee and bakery



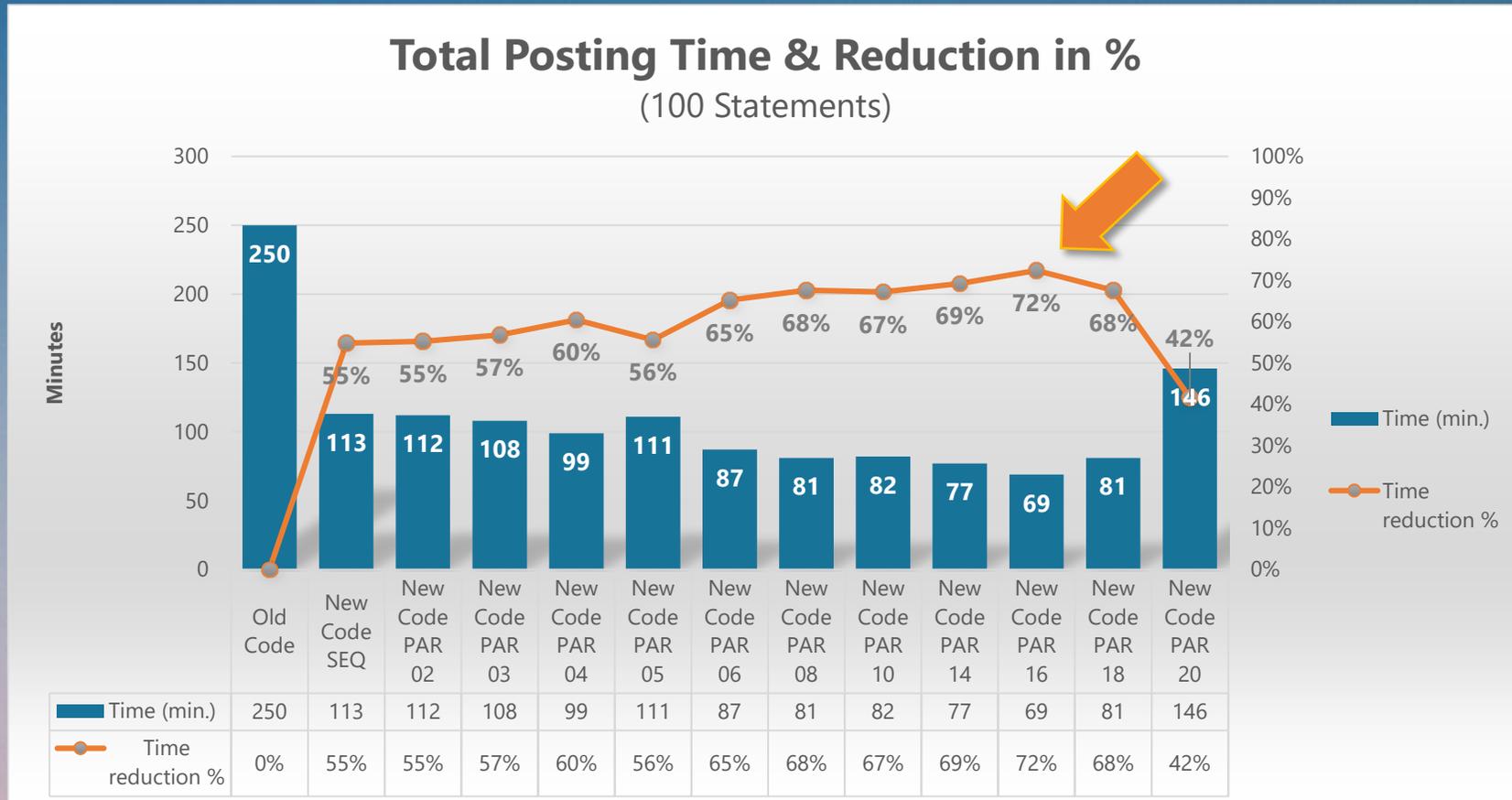
## Performance test with customer's data

- 2 Terrabyte database
- 200+ stores, 6000+ items
- Massive sales history and posted entries



# Statement calculation/posting Performance Tuning

Test results – FamilyMart database:



Original code:  
250 min. for  
100 statements

**55% reduction**  
with improved  
code

**Additional 17%  
reduction** with  
parallel posting

**Total reduction:  
72%**

# Summary

## Statement calculation

- **70% faster** due to code improvements
- New process: Incremental Statement Calc.

## Statement posting

- **55% faster** due to code improvement
- Improved process: Parallel Batch Posting  
(**72% time reduction** with parallel posting)

## Microsoft improvements

(as presented in April 2021)

- **40% faster posting** of background document batches



# How to get started ?

All the shown functionality is part of LS Central version 17.4 (released March 2021)

Running on AL based version?  
(version 15 onwards)

- Download LS Central 17.4 from LS Retail's Partner portal
- Test and Implement

Running on C/AL based version?  
(version 14 and earlier)

- Contact LS Retail Technical Support for C/AL based object-package (version 14.3)
- Test and Implement





# Solution Audit

LS Retail Consulting offers support on different environment reviews:

- Solution Architecture
  - Service Tier servers and other application servers
  - Data flow, web services and replication
- SQL Server audit  
(with LS Retail's performance audit toolbox)
  - Configuration, maintenance jobs
  - Performance issues

These reviews give indicators for

- Hardware sizing and bottlenecks
- Software configuration and bottlenecks
- Best practices guidelines





**Don't  
miss  
the train**



**Thank you**

**Thank you  
for attending**

[www.LSRetail.com](http://www.LSRetail.com)