

CICSplex From Scratch

A User Experience Moving CICS Applications to CICSplex

David Stephens

Longpela Expertise / CPT Global

Share Fort Worth, February 2020



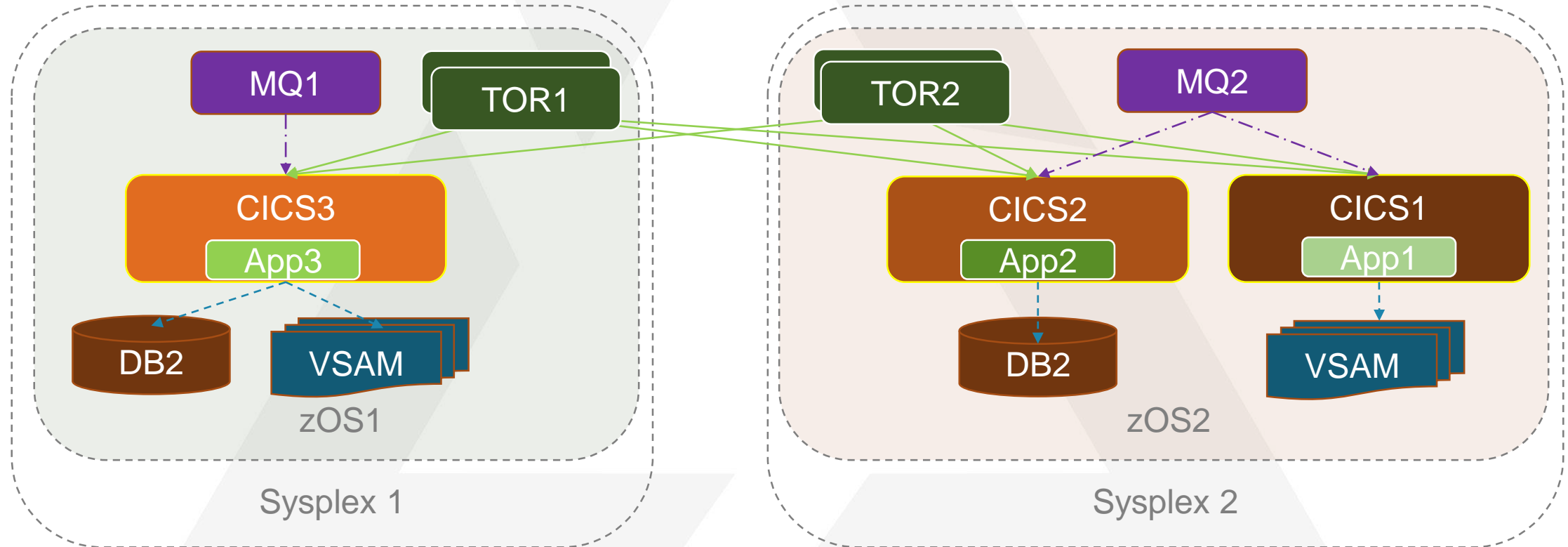


INTRODUCTION: MOVING APPLICATIONS TO CICSPLEX

The Goal: Before

Three Applications:

Three CICS Regions, Two z/OS Systems, Two Parallel Sysplexes



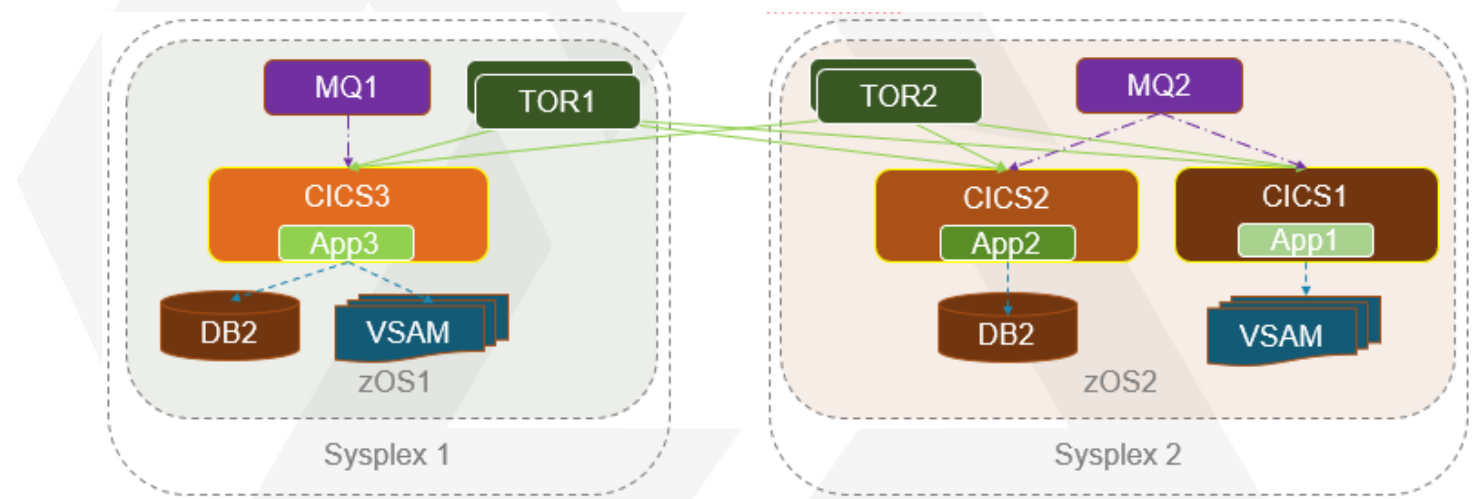
The Goal: Before

Three Applications:

- CICS/COBOL
- Primary Traffic: MQ
- 3270 via TOR
- Some Web / TCP/IP
- App1: CICS/VSAM
- App2 & App3: CICS/DB2, bit of VSAM

Single Threaded:

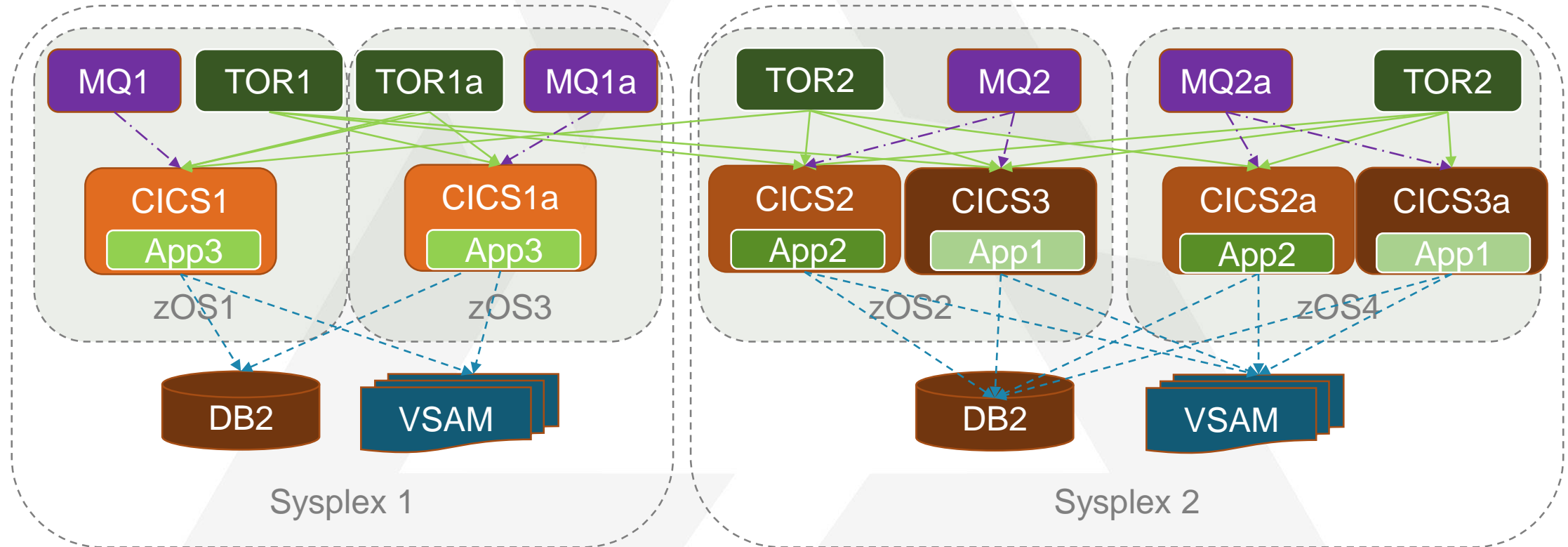
- One route in, one route out
- Lots of single points of failure



The Goal: After

Three Applications:

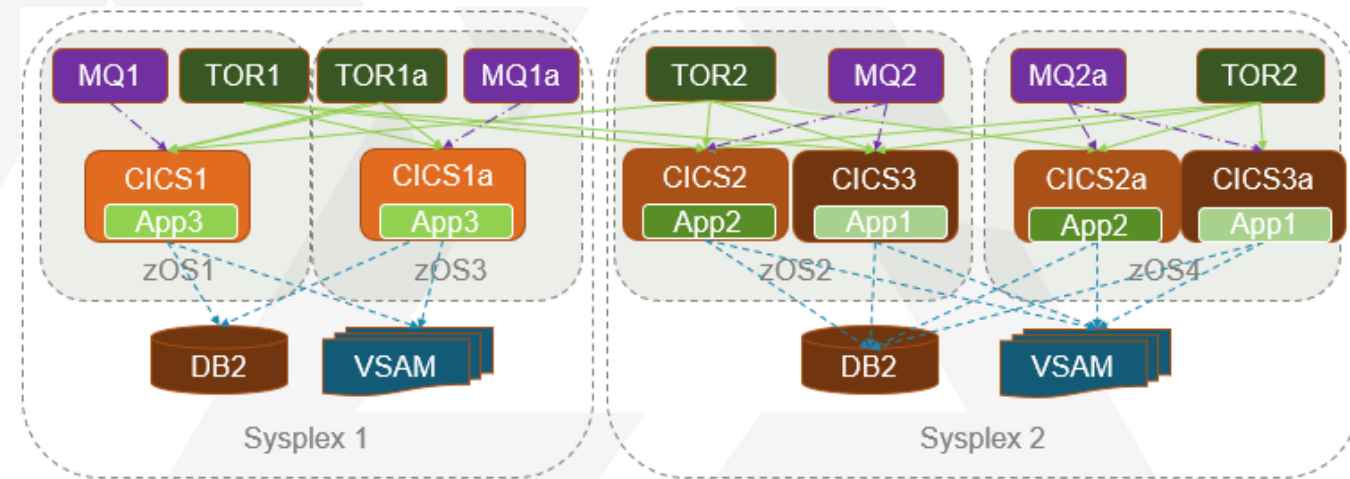
Six CICS Regions, Four z/OS Systems, Two Parallel Sysplexes



The Goal: After

Active-Active:

- Two 'copies' of each application
- Redundant MQ, TOR, AOR
- CICS regions share workloads



New Technology:

- CICSplex / CPSM
- MQ Queue Sharing (QSG)
- DB2 Data Sharing (DSG)
- VSAM RLS
- CICS Global ENQ
- CICS Shared temporary storage
- CICS Coupling Facility Data Tables (CFDT)
- CICS Named Counters

The Project: Systems and Applications

Project required both systems **and** application changes

Not a lot of information or user experiences on the application side.

Issues found implementing MQ queue sharing, VSAM RLS, CICS and more.

Today: Covering CICS: Systems and Application Issues:

- Four Systems Problems / Issues
- Six Application Problems / Issues
- Application Performance

(Spoiler alert: we broke stuff)

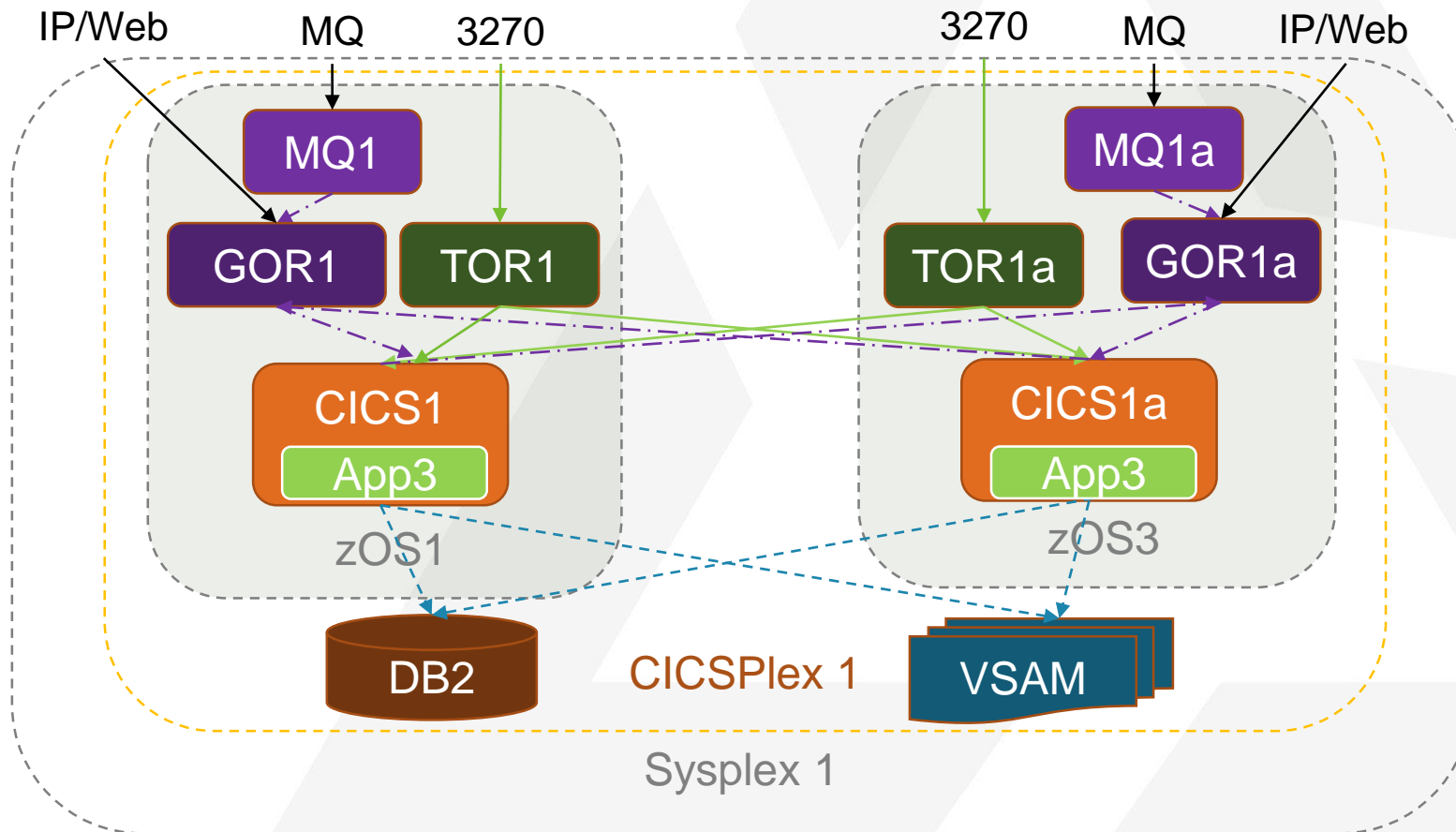


CHAPTER 1: CICS/CPSM SYSTEMS ISSUES

(Not all of them – only some of the more interesting ones)

Problem 1: The CICSplex Design

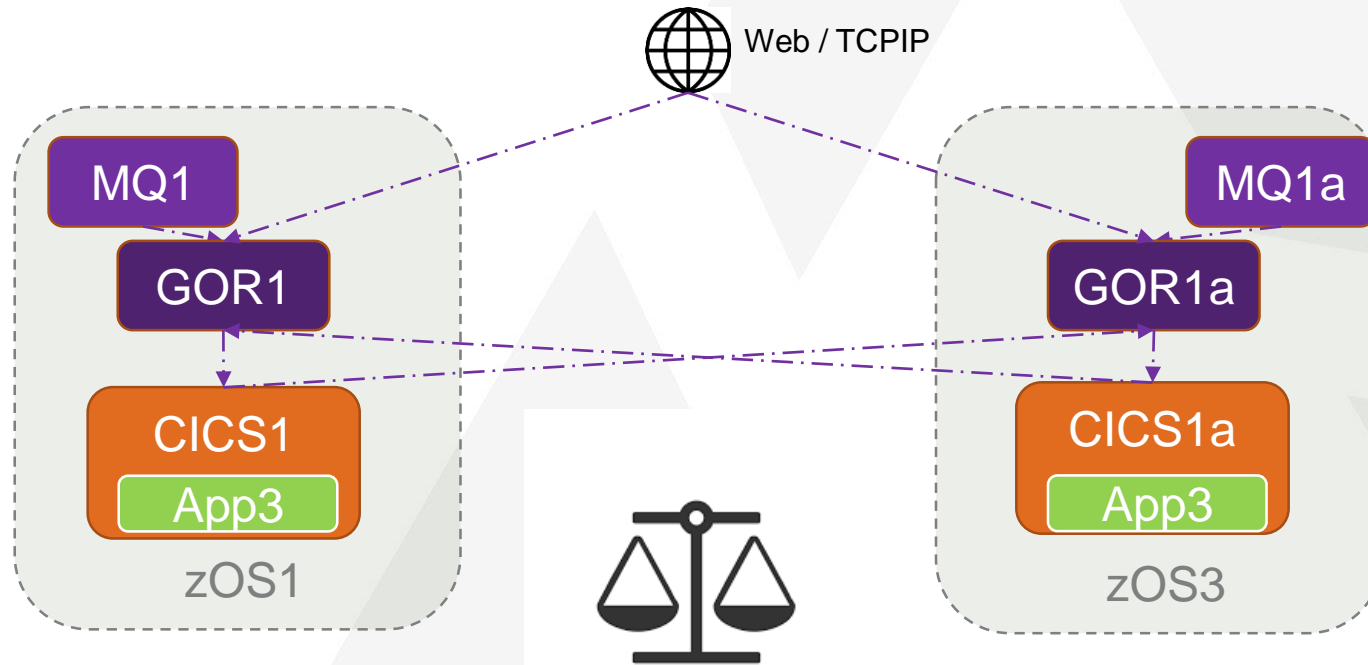
One CICSplex per Sysplex



- Existing TORs
- QOR for MQ triggering
- WOR for incoming IP/Web
- Combine QOR/WOR = “Gateway Owning Region”, or GOR

Problem 1: The CICSplex Design

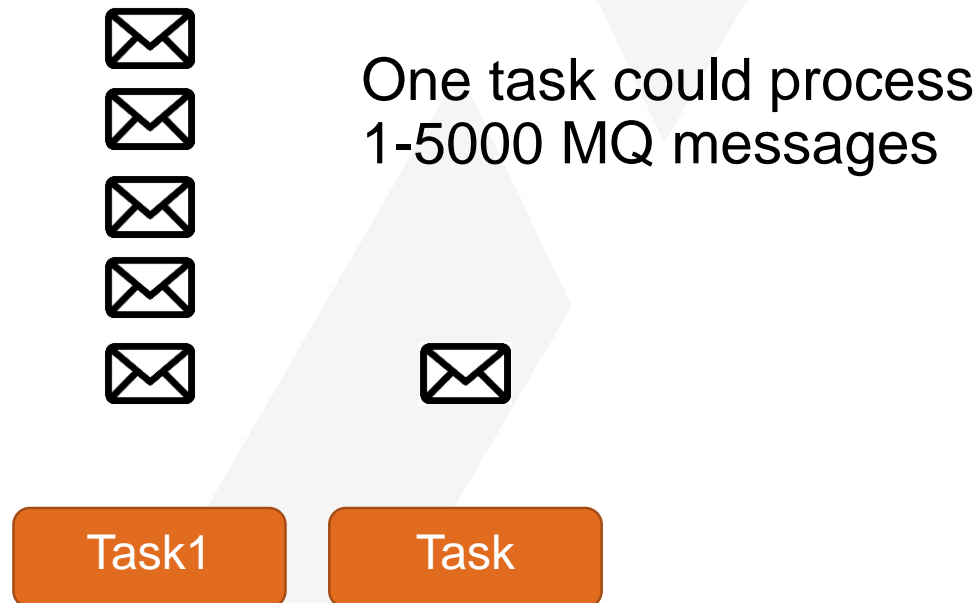
Client Asked: Why QOR/WOR (GOR)? Better Workload Balancing



- CPSM can prefer AOR with less workload
- Chose Link Neutral Queue (LNQUEUE) routing

Problem 1: The CICSplex Design

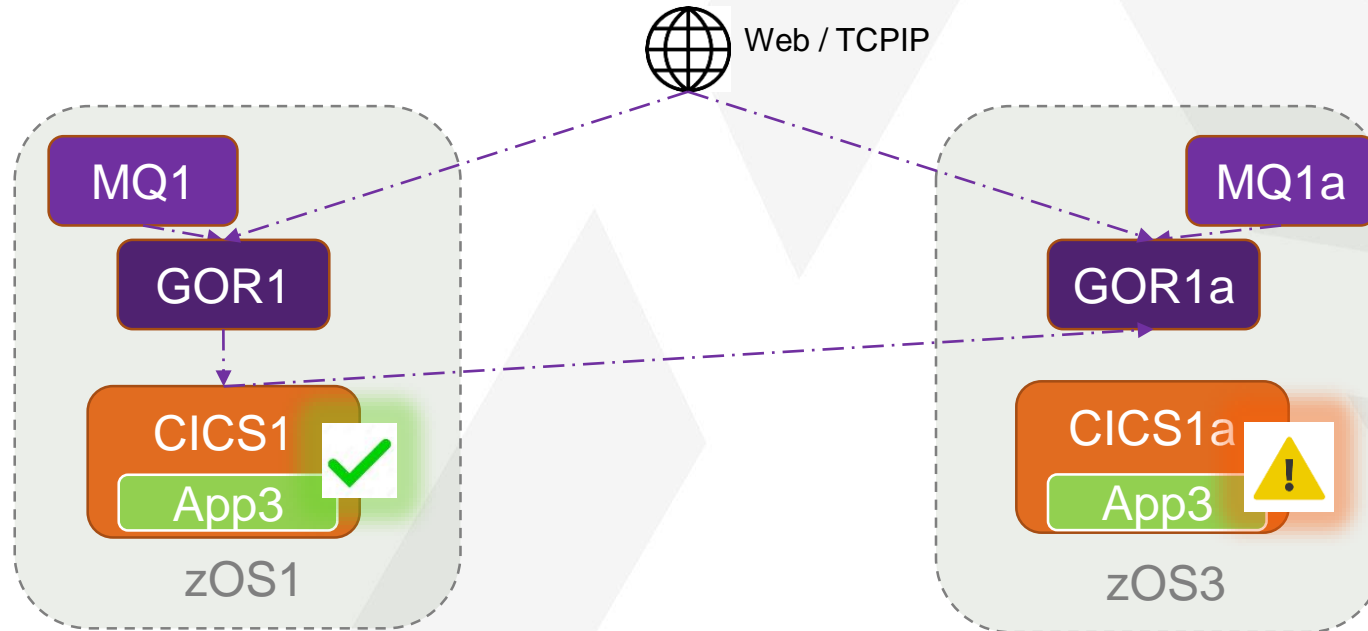
Why LNQUEUE? Long-running MQ processing



- Core processing performed by long running transactions processing multiple MQ messages
- Response time goals not suitable for these types of transactions
- GOAL routing algorithm good for transactions with response time goals from terminals (DTRPGM)
- QUEUE or LNQUEUE better for our workloads
- *(More on this shortly)*

Problem 1: The CICSplex Design

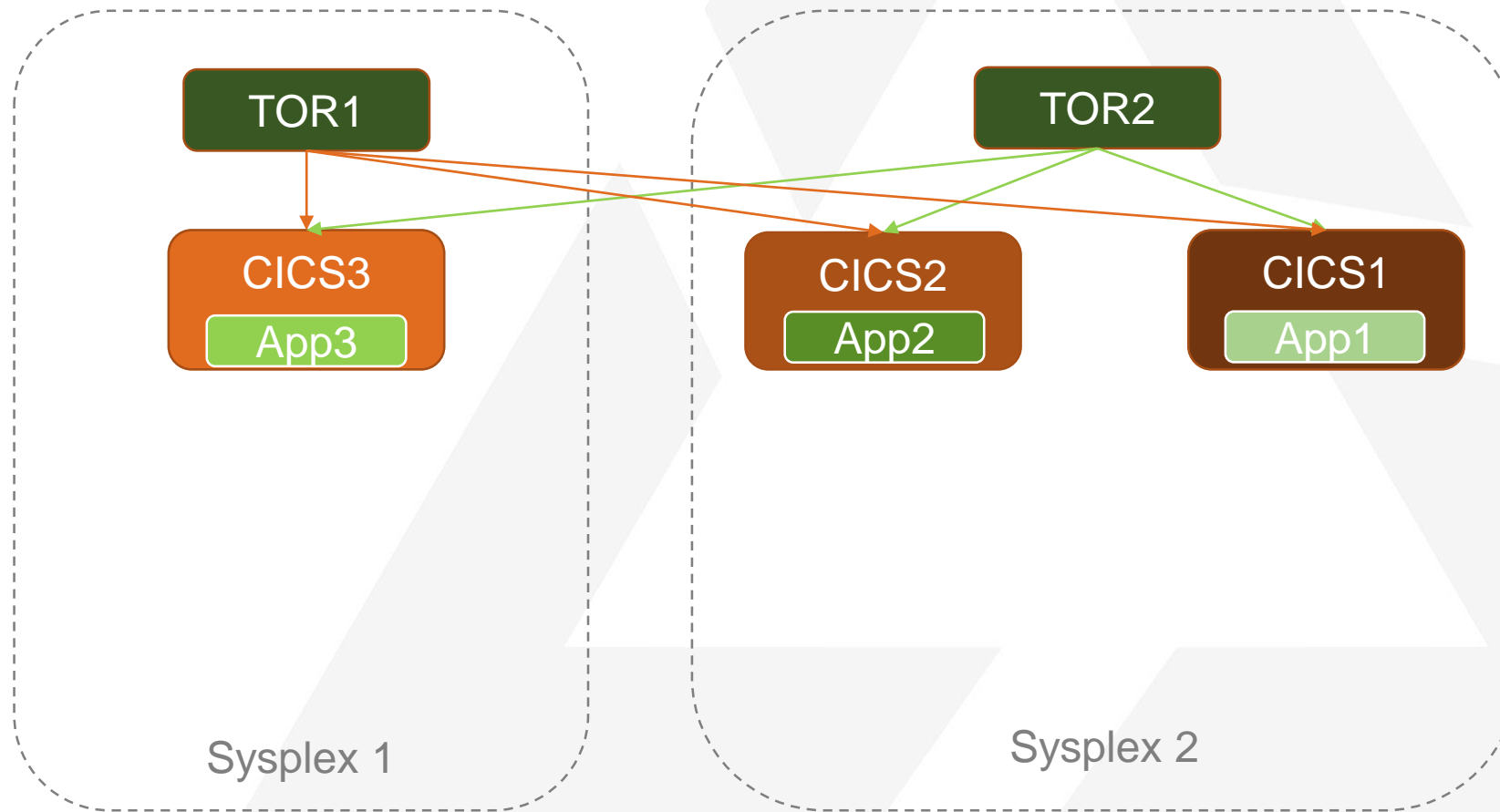
Client Asked: Why QOR/WOR (GOR)? Better Resilience



- If one AOR is 'ill' (maxtasks, SOS, etc), CPSM can prefer the 'healthy' one
- *(More on this shortly)*

Problem 1: Single CICSplex and TORs

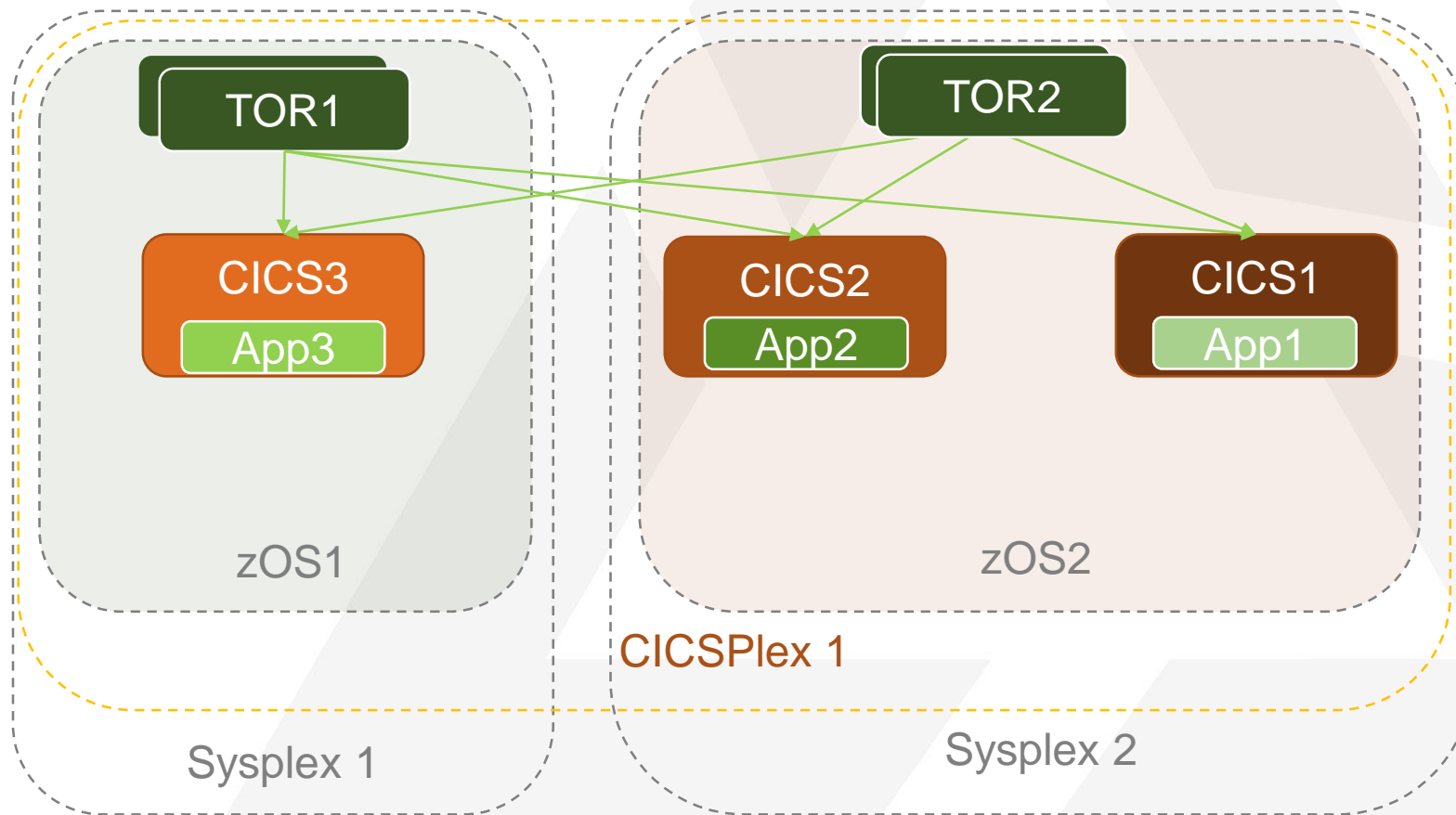
This design didn't work: TORs in different sysplexes connected to CICS AORs



- TORs routed 3270 transactions 'across' sysplexes
- Cannot route TOR transactions from one CICSplex to another
- Client did not want to force users to change TORs

Solution 1: One CICSplex, Two Sysplexes

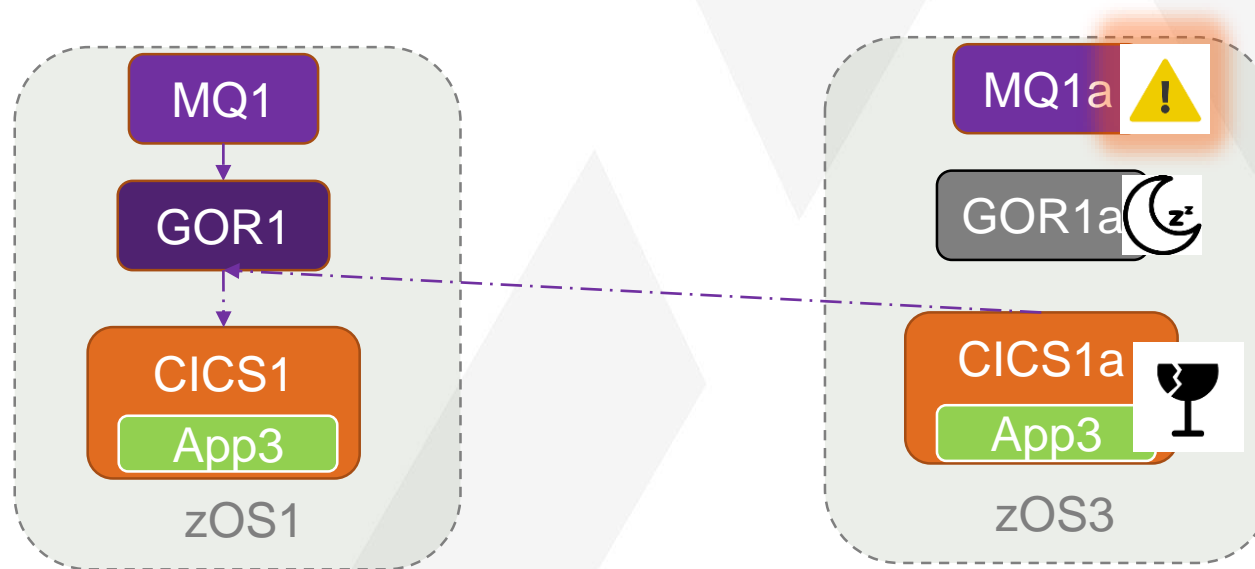
Allows CPSM to route all 3270 traffic across sysplexes



- Only needed for 3270 traffic.
- No cross-Sysplex routing for MQ/IP.

Problem 2: No Resilience on MQ Failure

CPSM continues to route to CICS AOR with no local MQ queue manager

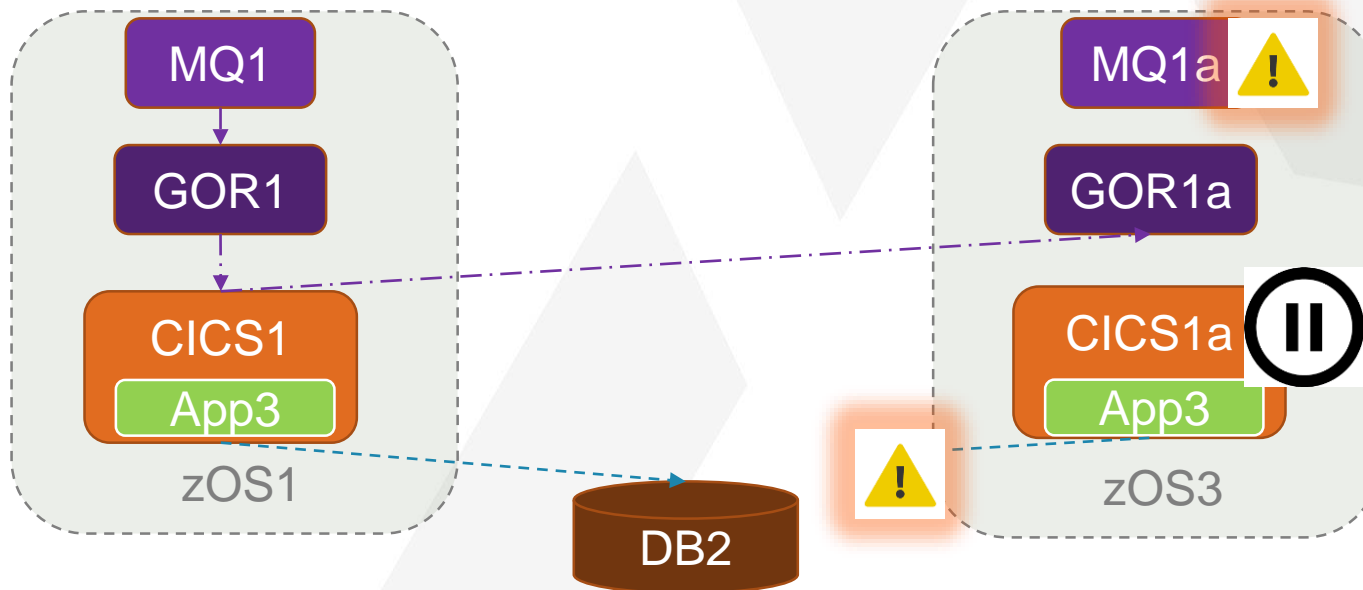


- Suppose one MQ queue manager fails
- GOR1a stops triggering
- GOR1 triggers, and continues to route traffic to CICS1 and CICS1a (*CPSM does not detect failure*)
- CICS1a transactions fail (*no local MQ*)
- But: CICS1a transactions end normally (*application handles abend*)
- More traffic routed to CICS1a ([stormdrain effect](#))

(We actually tested this by running a workload in both CICS AORs, and then cancelling one MQ queue manager)

Solution 2: Automation

Automation isolates 'ill' CICS

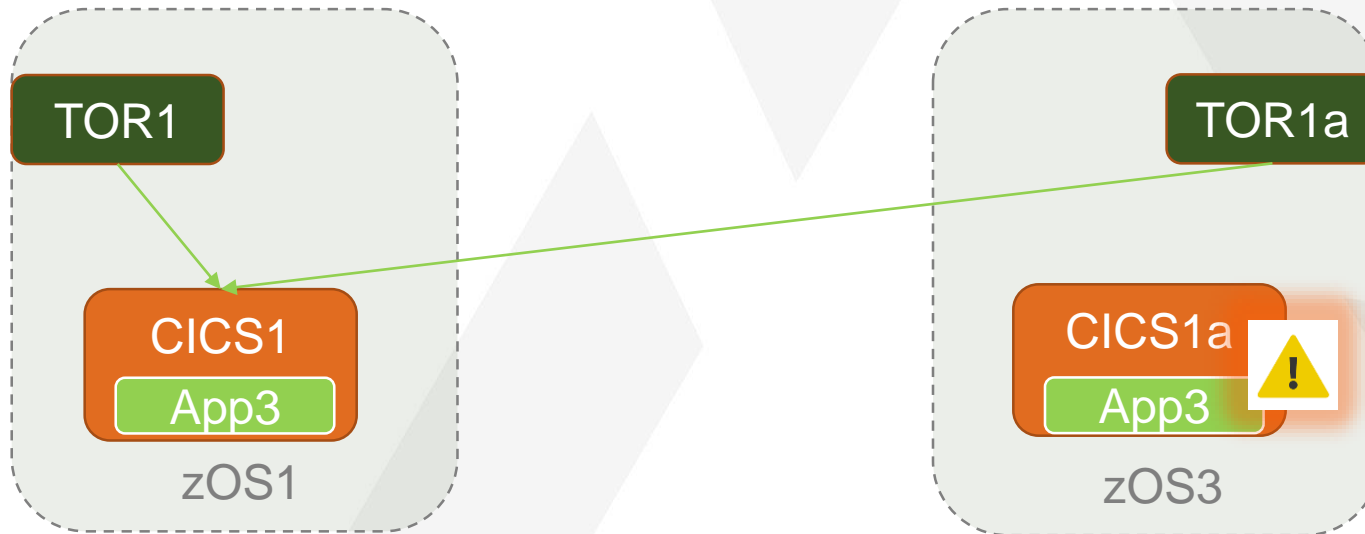


- Automation detects CICS messages on DB2/MQ connection failure
- Executes REXX script to quiesce CICS1a
- When automation detects connections up and running, executes another REXX to 're-activate' CICS1a

```
DFHMQ0334 I 09/05/2019 08:51:35 CICSAP3A Successful disconnection from queue
manager MQM0 release 0900 group MQG0
DFHDB2025I CICSAP3A 09/05/2019 08:24:45 CICSAP3A The CICS-DB2 attachment has
disconnected from DB2 subsystem D2M0 group DBG0
```


Problem 3: No Resilience for Non-Terminal Abends

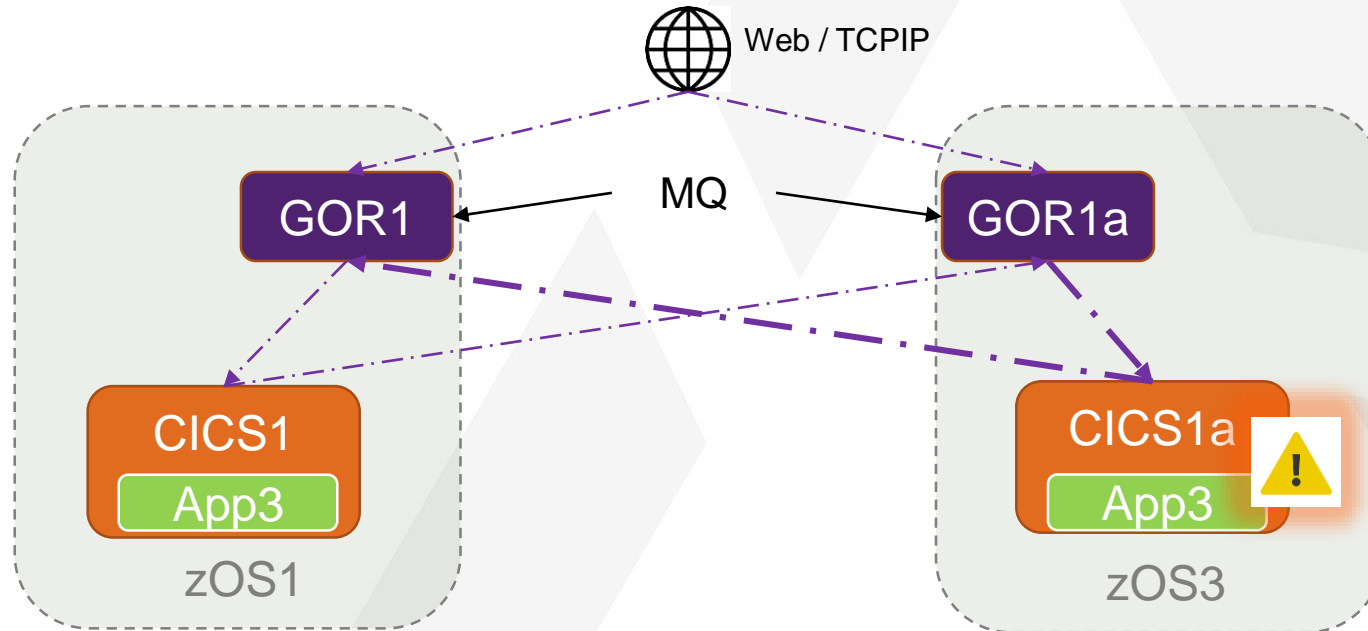
CPSM isolates a CICS AOR with lots of abends for terminal transactions



- Suppose many transactions begin to abend in CICS1a
- CPSM will route 3270 traffic to healthy CICS – Abend Probability
- This is great

Problem 3: No Resilience for Non-Terminal Abends

CPSM continues to route to 'ill' CICS AOR for non-terminal traffic



- Suppose transactions begin to abend in CICS1a
- CPSM will still route non-terminal traffic (e.g. MQ, IP) to both CICS1 and CICS1a
- Stormdrain effect again – more traffic to 'ill' CICS1a
- Remember: our critical workload is MQ

From IBM Documentation:

Abend probabilities are taken into account only if you are using the dynamic routing exit DTRPGM. Abend probabilities are not considered if you are using the distributed routing exit DSRTPGM.

Solution 3: (Still Working On It)

Things we're looking at:

- Change routing from LNQUEUE to QUEUE: GOR/TOR will prefer local AORs. Reduce Stormdrain effect.
- CICS Workload Manager Health API (CICS TS 5.4+)
- Automation – stronger rules to check for CICS abends/failures. Rules if supporting regions (shared TSQ server, CFDT server etc) fail.
- Educate Ops so they can quickly 'isolate' an ill CICS region

But It's Not All Bad News

We tested a lot of failure scenarios (I love breaking stuff). Our CICSplex solution worked well for most of them:

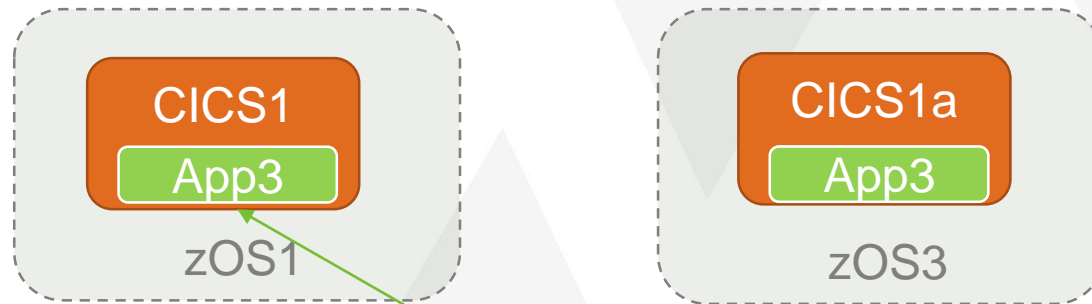
- Maxtasks
- Short on Storage
- CICS failure
- CPSM CMAS failure

Non-Functional Tests Attempted:

- Abend a MAS while processing
- Abend a CMAS (and maintenance point CMAS)
- Abend MQ queue manager
- Shutdown CICS/DB2 link
- Starve one LPAR of CPU
- Transaction on one CICS abends continuously
- Maxtasks
- Short on storage
- CICS startup without DB2, MQ, shared TSQ server, named counter server, CFDT server
- Shared temporary storage queues removed

Problem 4: Starting a Transaction From Batch

Batch job starts a transaction in a CICS region



- Using SDS CAFC
- CICS VTAM Applid specified in Job
- Now want resilience in case one CICS region is down.

```

//STEP1      EXEC  PGM=AFCP2016
//SYSPRINT  DD    SYSOUT=*
//CAFCTRAC  DD    SYSOUT=*
//CAFCTOS   DD    SYSOUT=*
//SYSIN     DD    *
CICS1,STRT,TXN1
  
```

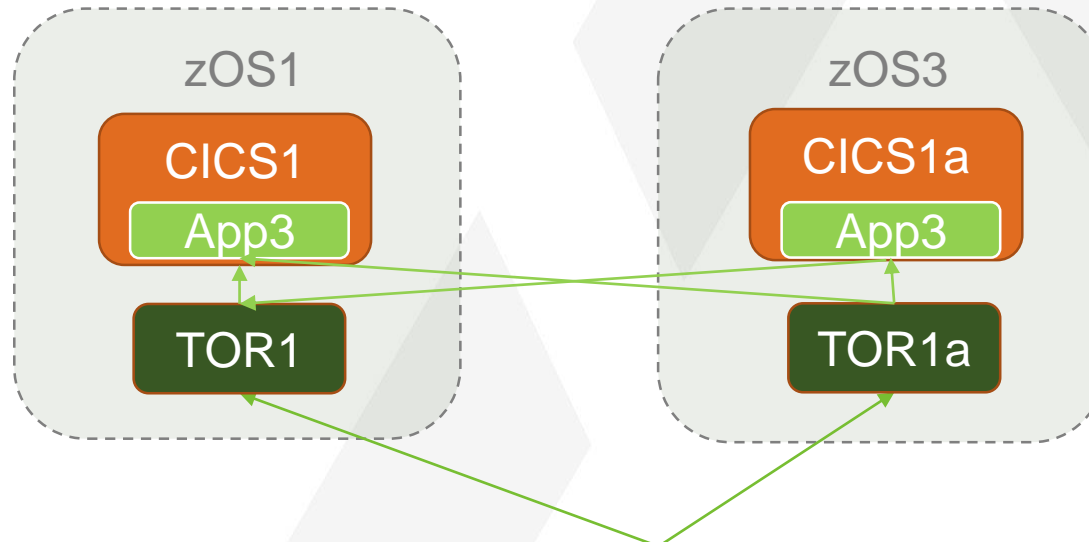
Command (Start Transaction) → `//CAFCTRAC DD SYSOUT=*`

CICS Applid → `CICS1,STRT,TXN1`

Transaction ID → `TXN1`

Solution 4: Route Through TOR

Batch job specifies Generic VTAM resource for TORs



- Use VTAM Generic Resources (already needed for 3270 users) pointing to TORs
- Specify this resource on CAFC jobs
- Connects to any available TOR: if one unavailable, other can be used.
- TOR routes to 'best' AOR. If one unavailable, other can be used.
- Uses VTAM – job can run on any z/OS in the VTAM network

```

//STEP1      EXEC  PGM=AFCP2016
//SYSPRINT  DD    SYSOUT=*
//CAFCTRAC  DD    SYSOUT=*
//CAFCTOS   DD    SYSOUT=*
//SYSIN     DD    *
PCICPLX,STRT,TXN1
    
```

Command (Start Transaction) → //CAFCTOS

Generic VTAM Applid → PCICPLX,STRT,TXN1

Transaction ID → TXN1

Lots of Other Issues

As you'd expect, this is only the beginning of the issues we faced.

- Batch job starting a transaction on every CICS region
 - WLM settings
 - Function shipping / DPL
 - Transaction timeout values
 - CPSM settings
 - Retained locks held by a failing CICS region
 - Managing transaction/program definitions across a CICSplex
 - Web/TCPIP workload issues
 - CICS monitor and CICSplex
 - Application uses same transaction in different Parallel sysplexes
 - Automation
 - First time using named counters and servers
 - First time using shared temporary storage queues and servers
 - First time using coupling facility data tables and servers
 - First time using CICS Explorer
 - Educating operations/application on CICSplex
 - RTA issues
 - Dynamic routing of EXCI requests
 - Dynamic routing of DB2 DSNACICS stored procedure
 - Coupling Facility structure recovery for shared TSQs, CFDTs and named counters
- But we can't cover them all today...

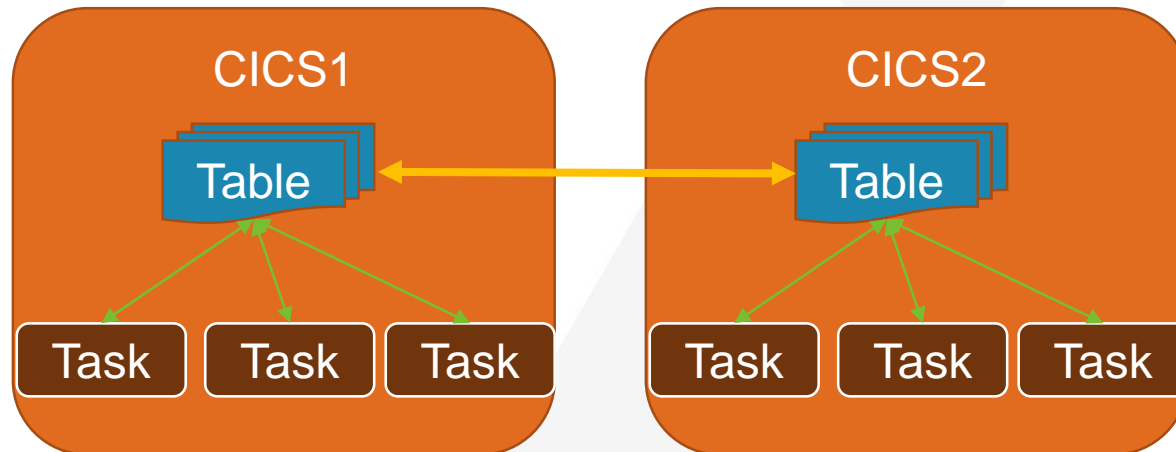


CHAPTER 2: APPLICATION ISSUES

(Again, only some of the more interesting ones)

Problem 1: In-Memory Tables

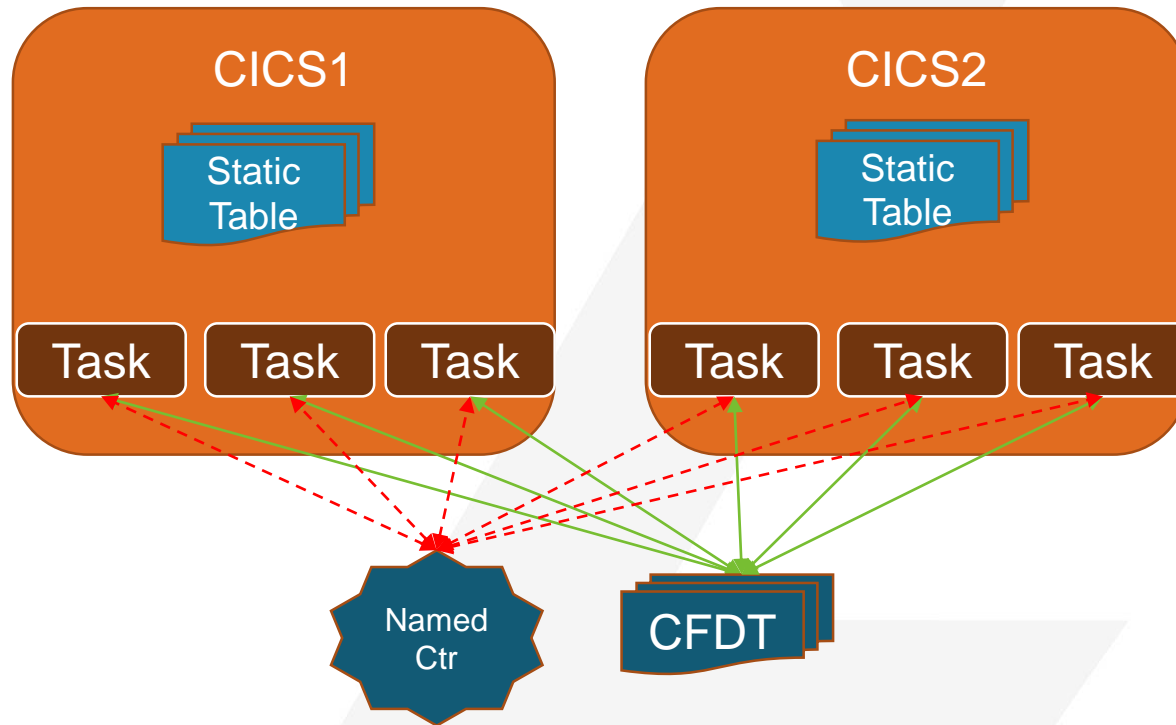
One application used tables in shared memory heavily



- One application used tables in shared memory, addressed through CICS CWA
- Tables loaded at CICS startup/application startup time into shared memory
- Tables read and updated by many different tasks
- With two CICS region, need to synchronize these tables

Solution 1: CFDT / Named Counter

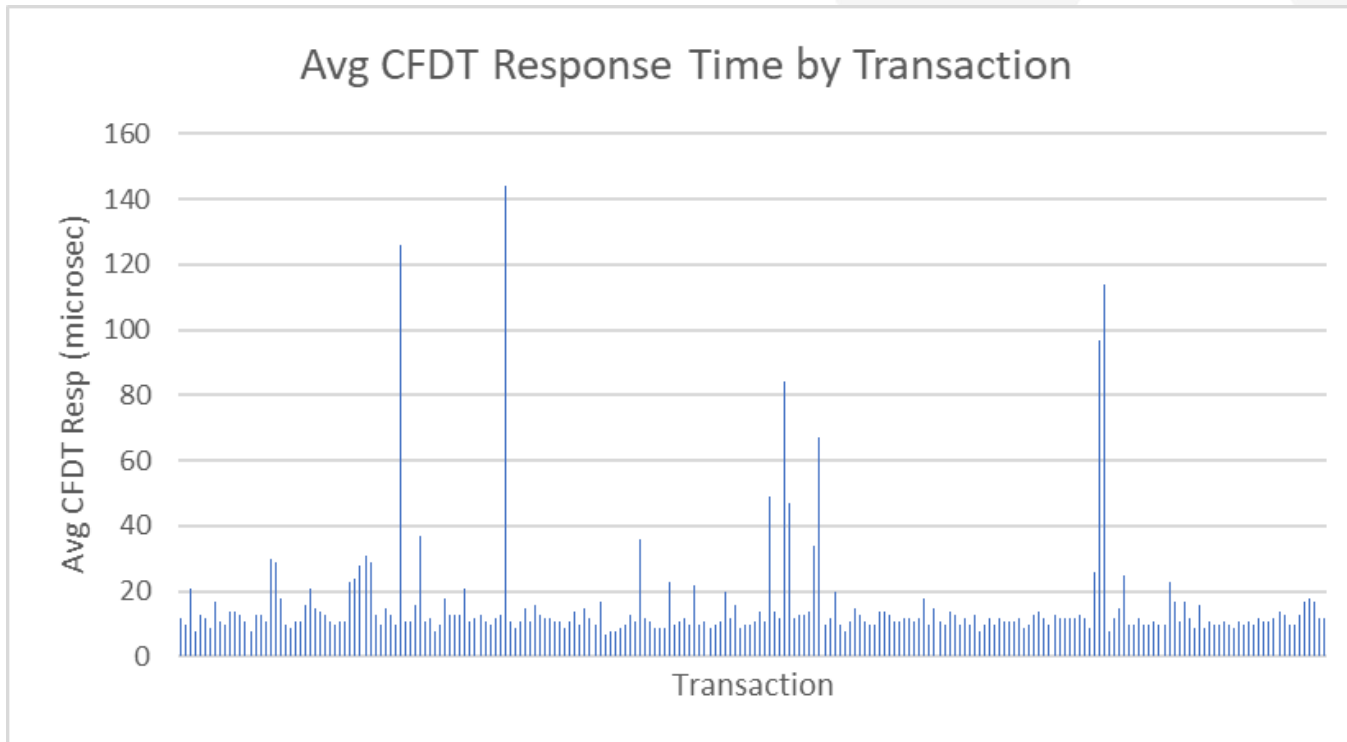
Used CFDT and Named Counter as the 'gold' copy



- Tables with static data (not updated): no change.
- Tables with dynamic data:
 - Counter fields replaced with Named Counters
 - Other fields replaced with Coupling Facility Data Tables (CFDT)

Solution 1: Why Not DB2?

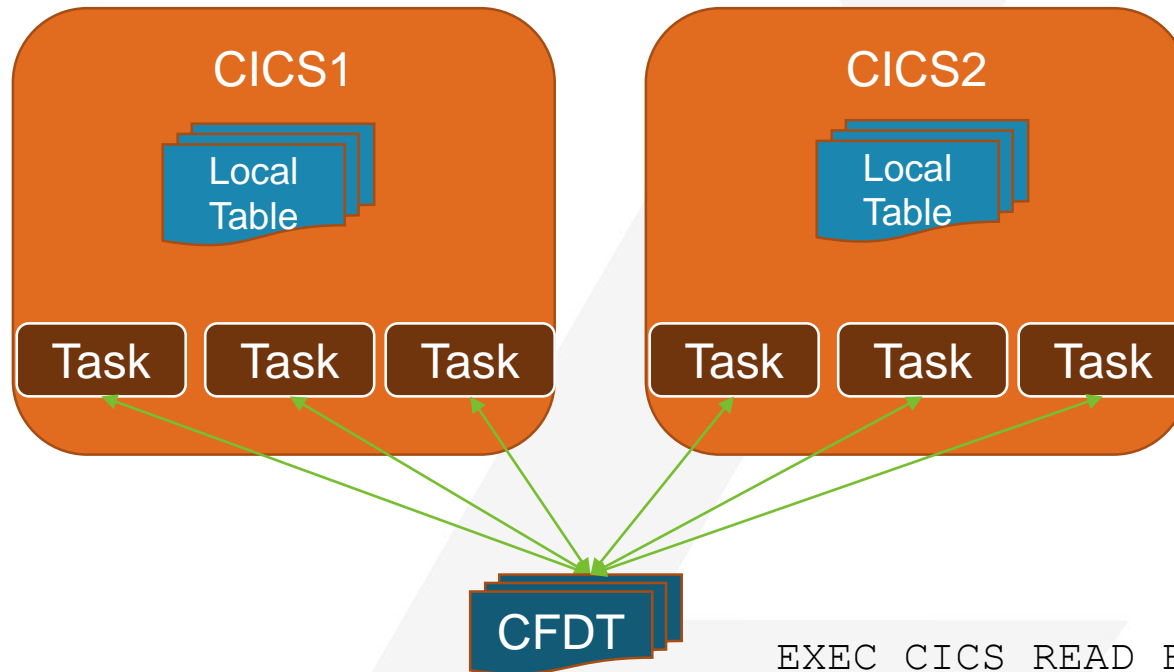
Why did we choose CFDT and Named Counters over DB2?



- Very heavily used: mainly read
- Application almost exclusively VSAM (little DB2)
- Decided that named counters and CFDTs were faster

Solution 1: Using CFDTs

Some issues using Coupling Facility Data Tables

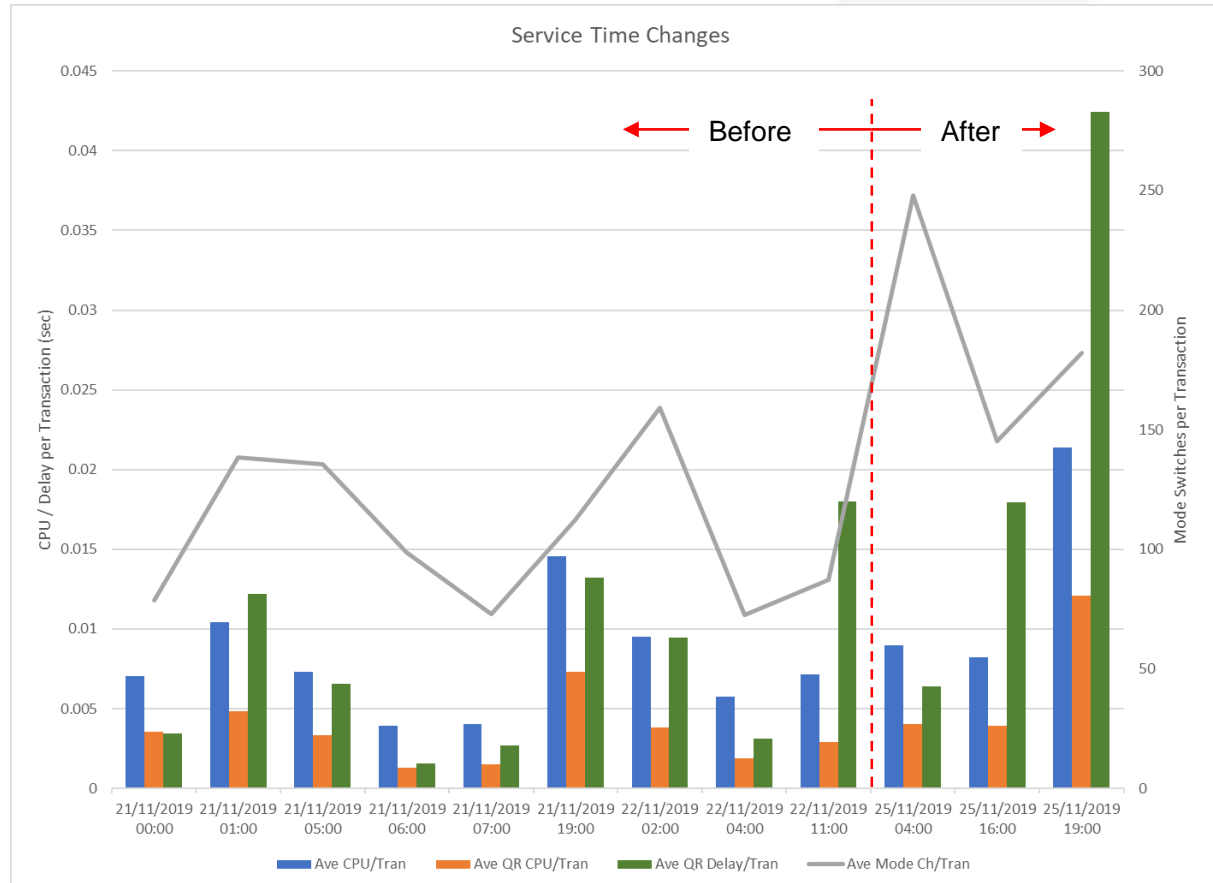


```
EXEC CICS READ FILE (CFDT1) UPDATE INTO (LOCAL-TBL)
      (change table)
EXEC CICS REWRITE FILE (CFDT1)
```

- Each CICS has a 'local' copy of table (*CFDT was 'gold' copy*).
- Applications would 'refresh' local copy, and then process as before (*minimized code changes*)
- Non-recoverable CFDT (*performance*)
- Serialization through READ UPDATE
- A 'new' lock: watch out for 'deadly embrace.'

Hiccup: CFDT Access Not Threadsafe


'Lost' threadsafe when using CFDTs



- CFDT access with CICS TS 5.3 not threadsafe.
- Missed when reviewing IBM documentation
- Missed during load testing (transactions were not threadsafe then)

Hiccup: CFDT Access Not Threadsafe

Performance still acceptable

Service Heat Map 26 November 2019 

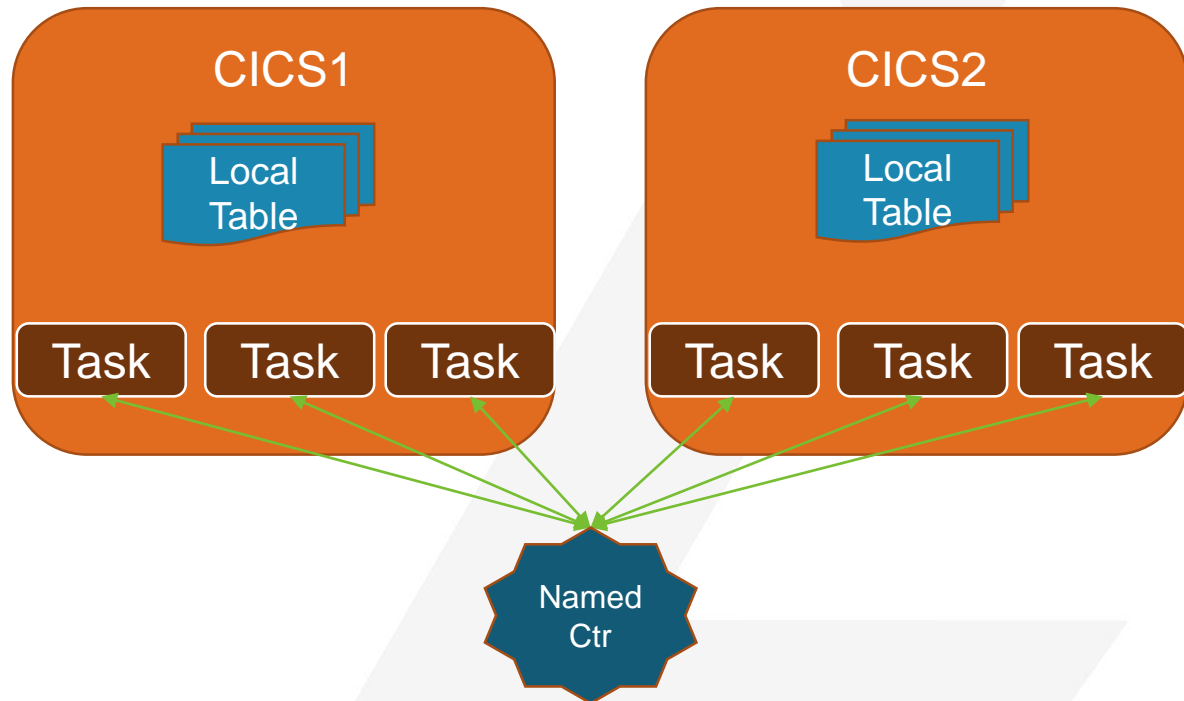
Transaction	Hour	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
97	912	1949	242	239	346	310	300	896	1104	969	1562	826	778	559	370	1278	696	198	523	81	200	27	51		
79	1219	416	160	183	233	204	234	587	521	956	1161	790	462	393	309	972	366	250	176	59	77	103	36		
74	937	398	200	253	252	123	213	704	530	640	736	568	392	309	240	793	321	183	250	108	148	83	44		
74	1116	426	213	251	333	158	231	740	702	1007	1115	706	498	396	400	813	419	188	212	83	153	91	59		
66	924	406	234	195	260	175	233	492	657	898	842	552	334	320	273	582	376	160	258	57	124	67	31		
97	1953	1333	728	417	629	400	549	1426	1475	2310	2882	1822	1275	847	663	2931	2275	795	508	303	135	329	58		
61	415	194	116	137	168	118	156	320	354	433	496	374	267	240	145	511	222	92	169	59	77	28	43		
20	61	123	29	53	21	21	22	40	31	40	27	26	25	21	24	21	21	21	0	0	0	487	20		
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	677	0	0	0	0		
6	1354	32	35	79	127	87	147	325	418	655	735	259	224	230	151	634	331	114	123	62	53	2	26		
37	556	287	214	267	265	136	143	646	521	440	527	825	502	263	123	474	126	163	191	46	51	29	30		
34	1307	1695	310	152	300	198	312	917	1185	1907	2407	1232	660	384	315	312	4310	511	346	194	52	209	20		
34	1249	1608	336	144	298	189	247	807	863	1065	1322	848	508	320	235	1320	841	272	179	155	55	266	23		
14	2199	50	83	73	70	42	70	162	242	275	570	334	177	110	277	366	257	68	4	23	7	3	3		
24	204	134	154	231	237	123	124	412	427	639	1140	763	405	274	241	1384	763	272	263	113	253	10	14		
10	63	57	67	67	74	65	56	228	183	279	452	337	245	120	105	600	379	121	53	39	53	3	6		
10	105	64	75	88	130	42	48	108	147	361	540	308	133	101	107	570	365	119	101	50	143	5	3		

- Monthly reporting shows amount of time in each hour transactions processing incoming MQ ‘work’
 - Black = working all the hour (no idle time = at maximum)
 - Red = working most of the time (almost at maximum room)
- Monitoring showed performance still OK (green – idle time each hour)
- **Solution:** remove threadsafe for ‘worst hit’ transactions, wait for CICS TS 5.5

See Longpela article [A Different Set of Goalposts for Application MQ Performance](#) for more details on how we monitored these MQ-based long running transactions

Solution 1: Using Named Counters

Some issues using named counters



```
EXEC CICS GET COUNTER (CTR1) VALUE (FLD1)
```

- Used single-word counters: signed
- Some in-memory tables being replaced used an unsigned fullword.
- Confirmed that maximum possible number was far less than single-word signed maximum.
- Applications must handle LENGERR
- Before counters, applications obtained number, then incremented.
- GET COUNTER increments, and then gets number.
- Applications had to change processing to handle this.

Problem 2: Building CFDT and Named Counter

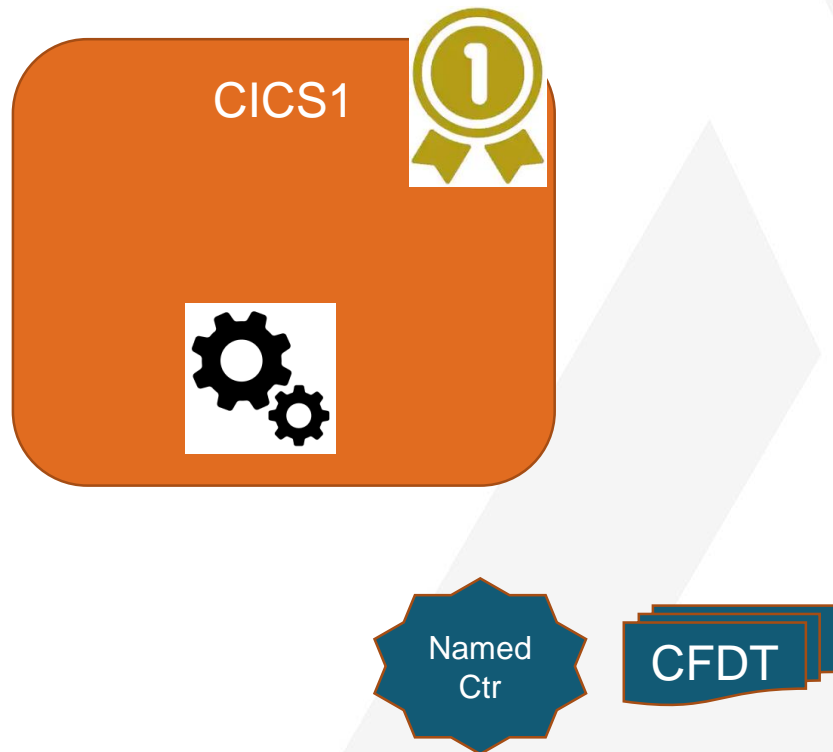
When do we rebuild CFDT and Named Counter?



- CFDT and named counters are non-recoverable. Need to be rebuilt from time to time
- In-memory tables were rebuilt during application start. With two CICS regions, which does it?
- Separated starts:
 - Big – CFDT and counters rebuilt
 - Small – use existing CFDT and counters.
- On coupling facility error: restart all application copies to rebuild structures

Solution 2: Building CFDT and Named Counter

How do we know if we are the first CICS region?



CICS Startup:

1. ENQ
2. EXEC CICS CPSM to see if any other CICS region is already up
3. For each CICS region, EXEC CICS LINK PGM() SYSID() to see if application is up
4. If find a CICS region with an application that is up, small start.
5. Otherwise, big start
6. DEQ once startup completed

In Memory Tables: Identifying

How do we know if an application uses an in-memory table?

Suspicious Commands:

```
EXEC CICS GETMAIN SHARED  
EXEC CICS ADDRESS CWA  
EXEC CICS LOAD
```

Two approaches to look for suspicious EXEC CICS commands

- DFHEISUP – CICS supplied utility to scan load modules
- Source Scanning

Output correlated with CICS end of day data to exclude programs no longer used.

Problem 3: Identifying Affinities

In-memory tables are an example of an affinity: change required for multiple CICS

Affinity	Solution
ENQ	If needed across multiple CICS regions, need global ENQ
INQUIRE	If need to inquire on status across multiple CICS regions, use EXEC CPSM
SET	If need to change status of a resource across multiple CICS regions, use EXEC CPSM
Temporary Storage	If shared across CICS regions, need global temp stg
Extrapartition Transient Data	If writing to a dataset, must be different dataset for each CICS
Intrapartition Transient Data	Cannot be shared across CICS: use MQ
ADDRESS	If passing storage addresses, may be an issue across CICS regions
START TRAN	If transaction must be started in every CICS region, must have multiple START TRANS
RETRIEVE WAIT	If used with START and RETRIEVE WAIT must execute in same CICS region
CANCEL	If cancelling a queued transaction, may need to be performed in every CICS region
DELAY	If waiting for a period of time, may need to be done in multiple CICS regions
Transaction Classes	If used for serialization, convert to global ENQ

Similar approach used to identify:

- DFHEISUP
- Source Scanning: looking for suspicious activity
- CICS end of day statistics – check if resources/programs are actually being used

Problem 4: EXEC CPSM

Applications need to use EXEC CPSM API

GCCS / 0017 EXEC CPSM for Beginners v1.3

9 APPENDIX: SAMPLE REXX BATCH PROGRAM

The following sample JCL executes CPSM in batch. This example performs an open/close of a file. This JCL executes a REXX program, which in turn calls CPSM.

9.1 JCL

```
//*  
//TSO EXEC PGM=IKJEFT1B  
//STEPLIB DD DISP=SHR,DSN=XIC01.CPSM.SEYUL  
// DD DISP=SHR,DSN=XIC01.CPSM.SEYUL  
//*  
//SYSTSPT DD SYSOUT=*  
//SYSEXEC DD DISP=SHR,DSN=XXXXX.XXXXX (REX  
//*  
//SYSTSIN DD *  
%CFSMFILE UBNYPLEX UGSPAOR G01ACTP1 C  
%CFSMFILE UBNYPLEX UGSPAOR G01ACTP1 O
```

Explanation of SYSTSIN parameters:

CICSPlex	=	UBNYPLEX
Scope Name	=	UGSPAOR
File name	=	G01ACTP1
Actions:		O - Open C - Close

Sysplex Project
EXEC CPSM for Beginners
(Version 1.3 January 2019)

Sysplex Project
PT Analysis of CPSM Command Security
April 2018

Prepared by:
CPT Global
Enabling Digital Business
175 Varck St
New York, NY 10014
www.CPTGlobal.com

Contact:
David Stephens dstephens@cptglobal.com
917.833.5884
Simon Foley sfoley@cptglobal.com
Brett Norris bnorris@cptglobal.com
Luke Tuijthof l.tuijthof@cptglobal.com

Contact:
David Stephens dstephens@cptglobal.com
917.833.5884
Brett Norris bnorris@cptglobal.com
Luke Tuijthof l.tuijthof@cptglobal.com

- Applications needed to communicate with their ‘copies.’ Examples:
 - Closing files
 - Checking if application active in another CICS region
 - Starting transaction in all CICS regions
- EXEC CPSM commands new to applications (and systems, and security, and ops)
- IBM documentation poor
- Provided our own ‘EXEC CPSM for Beginners’ doc, examples and training
- SCLM (CA Endeavor) changes needed

Problem 5: CICSplex Was New

Client unfamiliar with CICSplex

cpt

Sysplex Project
So You Want Sysplex for Your Application?
(Sep 2018)

Prepared by:
CPT Global
Enabling Digital Business
175 Varco St
New York, NY 10014
www.CPTGlobal.com

Contacts:
David Stephens dstephens@cptglobal.com
917.813.5884
Brent Harris bharris@cptglobal.com
Lukia Tuckert ltuckert@cptglobal.com

SDS

Introducing CPSM

Much of the CICSplex features are provided by an IBM product called CICSplex Systems Manager (CPSM)

CPSM is provided free with CICS, and works closely with it.

CPSM Runs in Every CICS Region

CPSM agent code runs also runs inside every CICS region in the CICSplex.

```

    Display Filter View Print Options Search Help
    CICS OUTPUT DISPLAY PCIC002E ST065775 DS10 2 LINE 168 COLUMN 21- 168
    COMMAND INPUT *****
    *EYU530 CICS002E Initializing CICS Web environment.
    *DFW108B CICS002E CICS Web environment initialization is complete.
    *JMS108B CICS002E Java processing attribute is STARTED.
    *SP1044B CICS002E Initiating connection to CICSplex SM.
    *YVX0801T CICS002E CPM Initialization program starting
    *YVX0802 CICS002E CPM Version 3.0.1 CMS startup in progress
    *YVX0810 CICS002E CPM Kernel loaded from EYU530
    *YVX0802 CICS002E CPM Phase 2 initialization complete
    *YVX0811 CICS002E CPM Start Up Parameters
    *YVX0812 CICS002E NAME(PCIC002E) Name of CICS Region.
    *YVX0813 CICS002E CPM END ***** CICSplex to share Region control
  
```

This code 'connects' to the CMAS on the local z/OS system when CICS starts up. Once connected, CPSM can provide functionality for that CICS region.

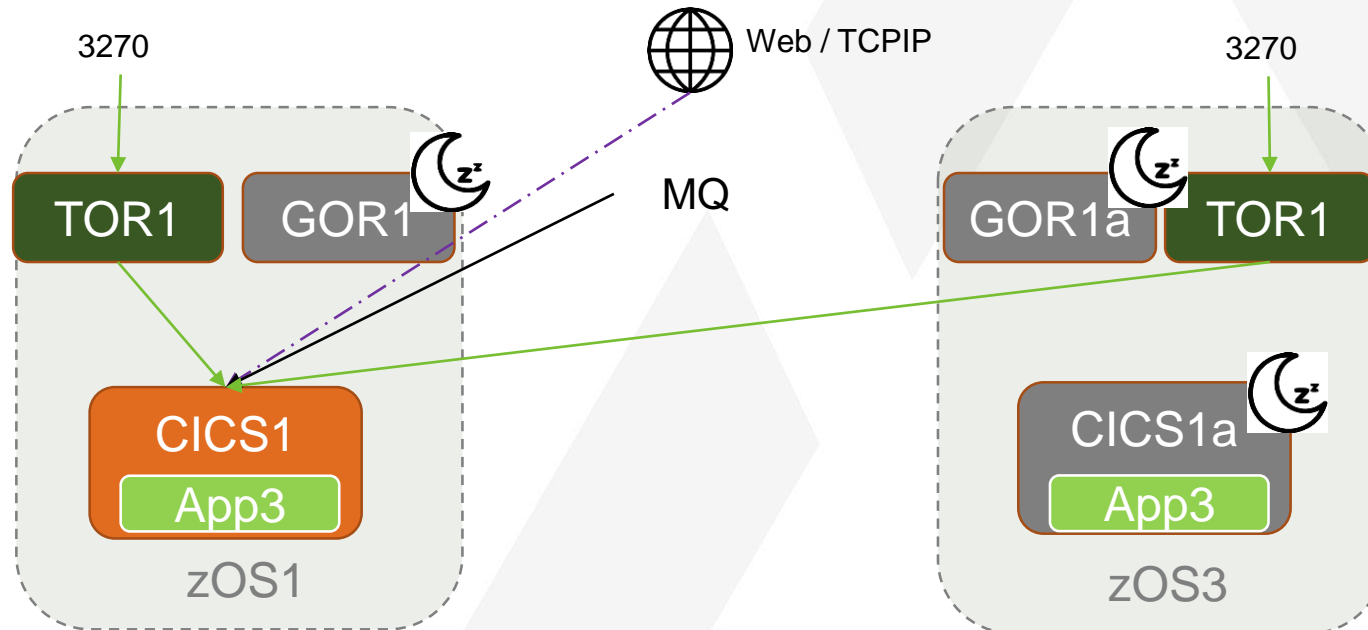
CICS can only connect to one CMAS that runs on the same z/OS system.

19 **cpt**

- CICSplex was new to client: systems, applications, operations, security, management.
- Again, IBM documentation was poor.
- Created our own documents, webinars and other information

Problem 6: Migration to Production

Client concerned about failures. So did a gentle migration. First, preparation



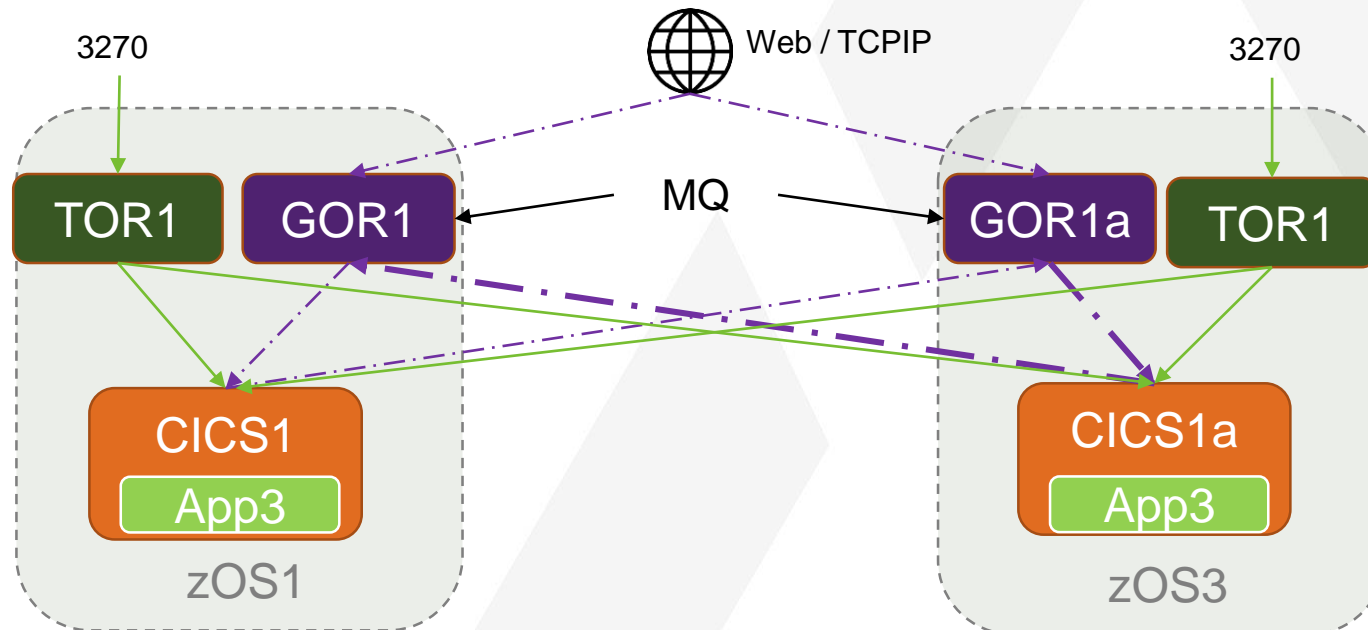
No work performed by CICS1a or GORs

Before Changing:

- Application changes completed.
- CPSM definitions completed
- 3270 TOR transaction definitions statically routed to one AOR *DYNAMIC (NO) ROUTABLE (NO) REMOTESYSTEM (aor)*
- IP/Web direct to one AOR.
- MQ queues trigger to one AOR
- MQ channels and queues shared
- VSAM files converted to RLS
- Batch jobs using EXEC CPSM for files
- All transactions defined in GORs as *DYNAMIC (YES) ROUTABLE (YES)*

Solution 6: Migration to Production

Migrated to production slowly: transaction-by-transaction



To Change:

- MQ triggering moved from AORs to GORs (and dynamically routed to either AOR) queue-by-queue. (*GORs dynamically route to AOR*)
- 3270 transaction definitions changed one-by-one in TOR from to DYNAMIC (YES) ROUTEABLE (YES)
- IP/Web traffic modified one-by-one at remote end to use GOR (*GORs dynamically route to AOR*)

Lots of Other Issues

As you'd expect, this is only the beginning of the issues we faced.

- Hard-coded CICS region names
 - Hard-coded z/OS system names
 - Sharing a CICS region with another (non-Sysplex) application
 - Printers defined in TORs
 - External applications calling programs/transactions
 - External applications accessing files, transient data queues
 - Foreign application programs running in region
 - Web/IP traffic issues
 - PLT programs using EXEC CPSM APIs
 - Shared TSQs – no data deletion
 - Shared TSQs – removing unused TSQs
 - Program NEWCOPY PHASEIN across CICS regions
 - Terminal affinities required for some CICS transactions
 - Increased maxtask delays from application MQ triggering and transaction class usage
 - Lockout problems with locks on CFDTs
- But we can't cover them all today...

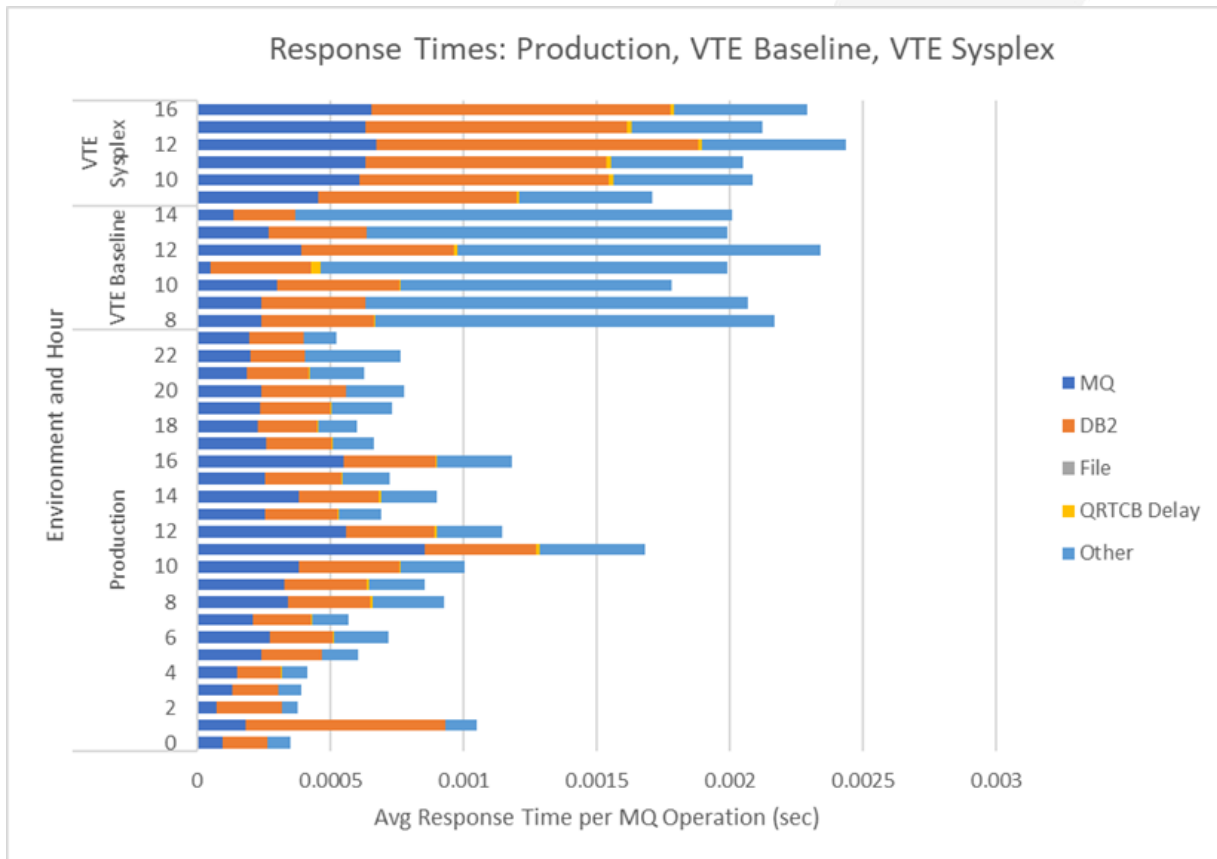


CHAPTER 3: PERFORMANCE

(A bit boring)

Performance: Load Testing

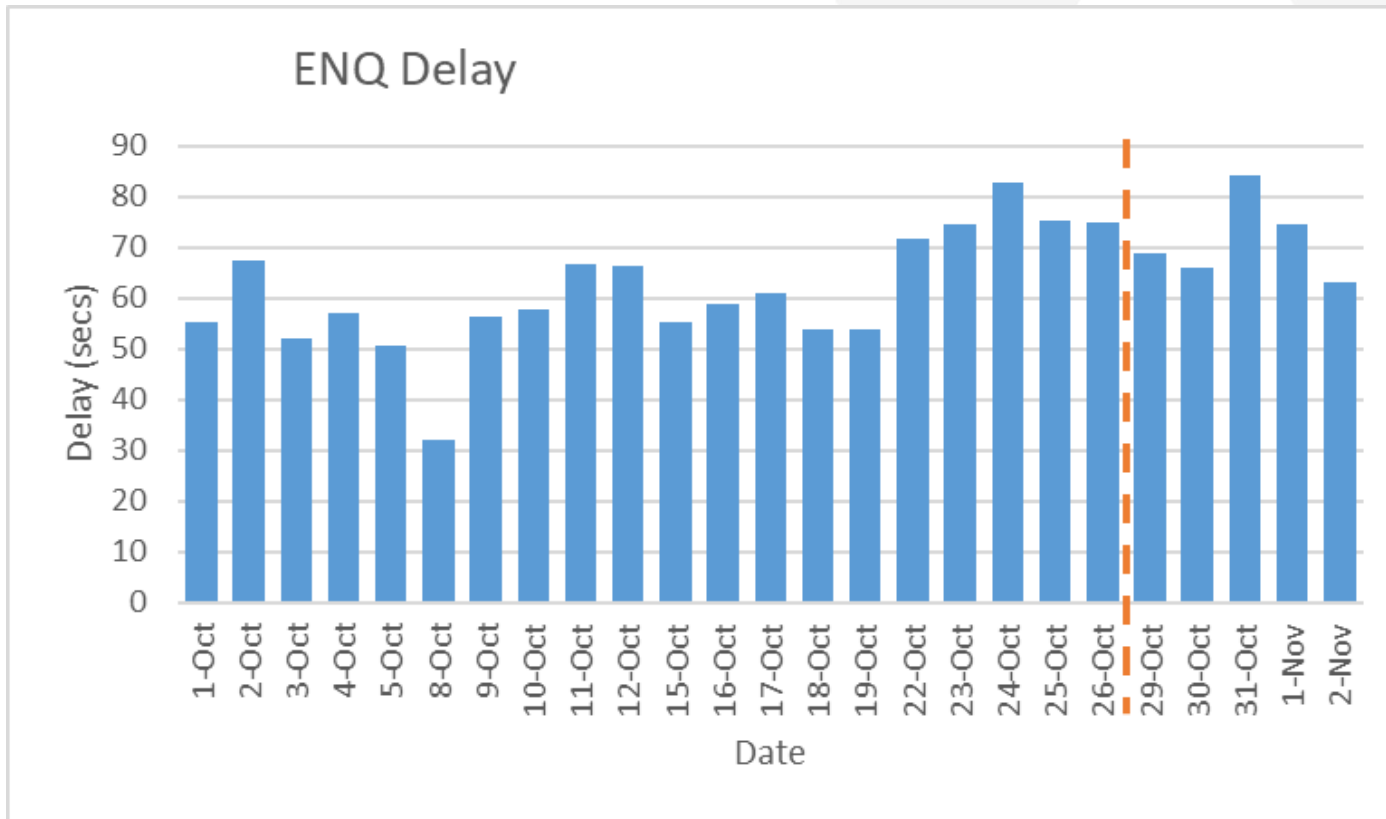
Client performed a load test before moving to production



- Looked at SMF 110 transaction statistics to compare:
- Production
- Test environment (VTE) before Sysplex changes
- Test environment (VTE) after Sysplex changes

Performance: ENQ

Closely monitored production changes. First change: ENQs.

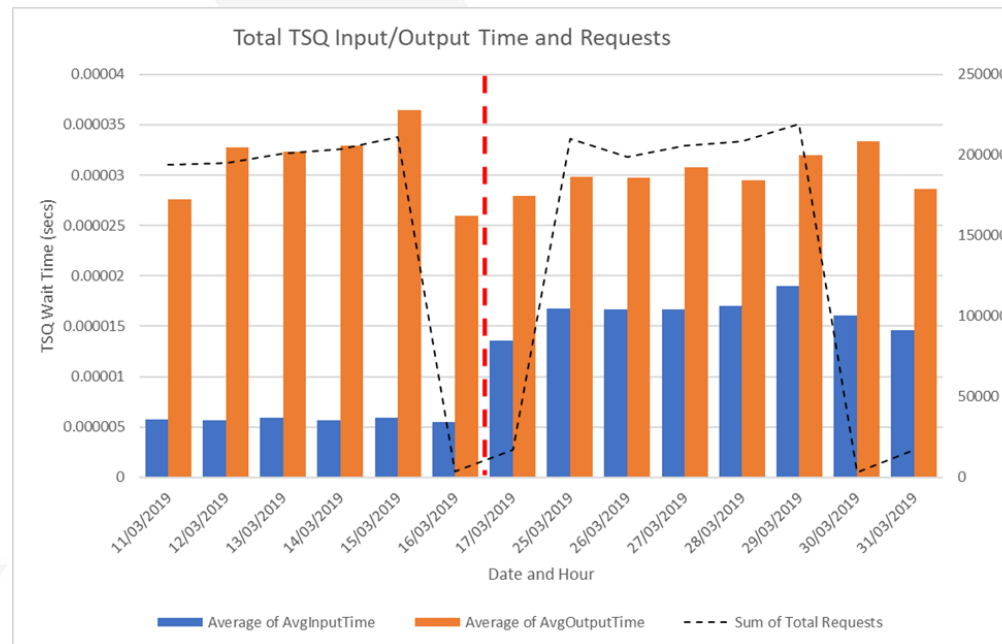
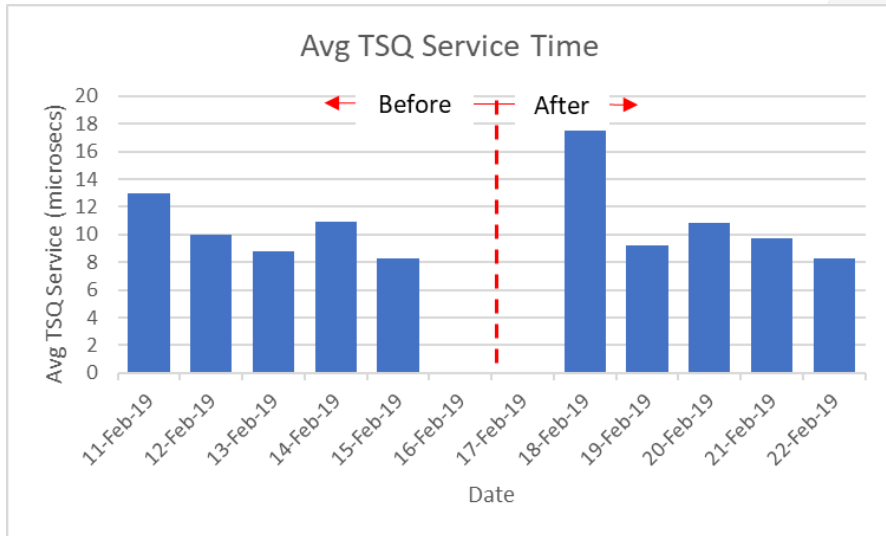


- Changed from local CICS ENQ to global ENQs
- Looked at SMF 110 (subtype 1) transaction statistics for ENQ wait times
- No measurable change detected

Performance: Temporary Storage

Changed from local TSQs to shared TSQs

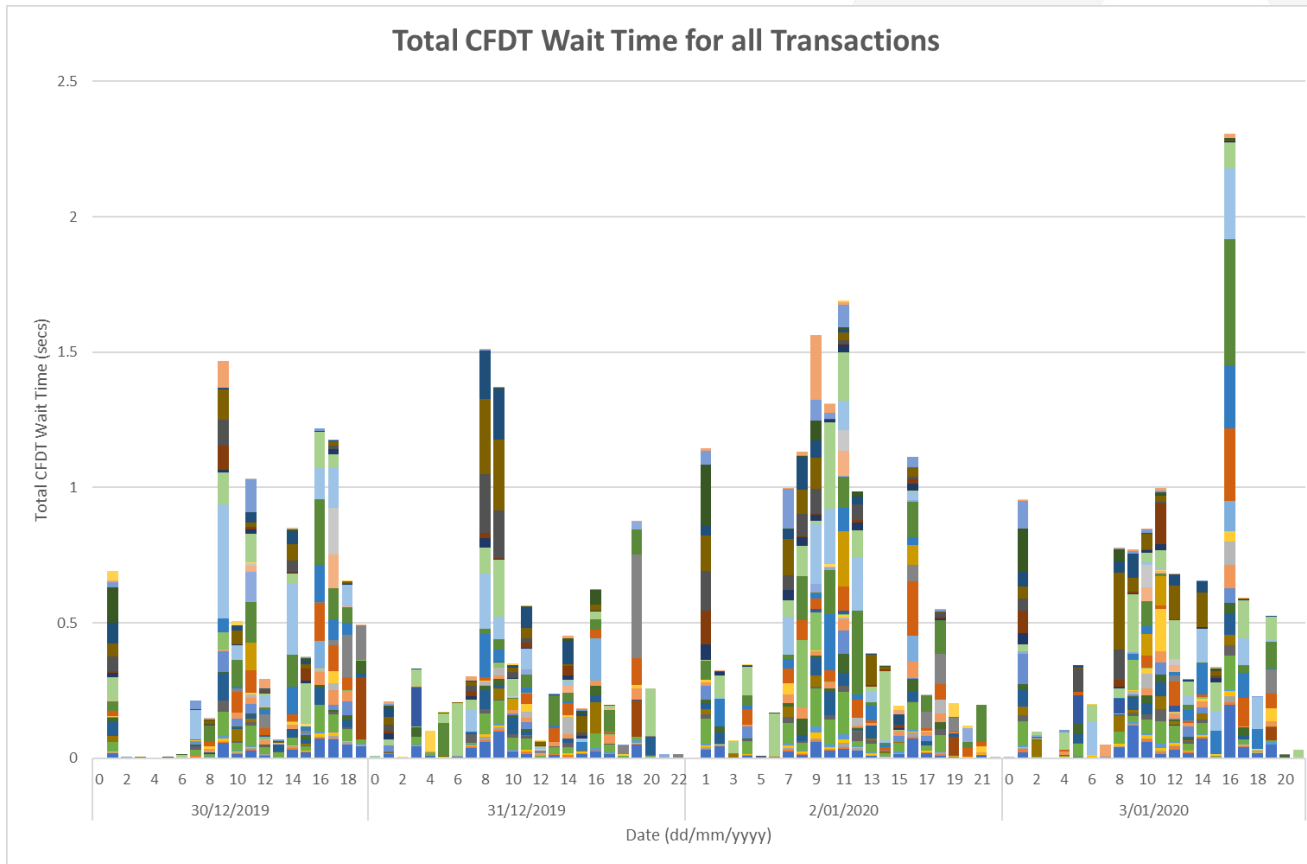
- Looked at SMF 110 transaction statistics for TSQ delays
- No measurable change detected



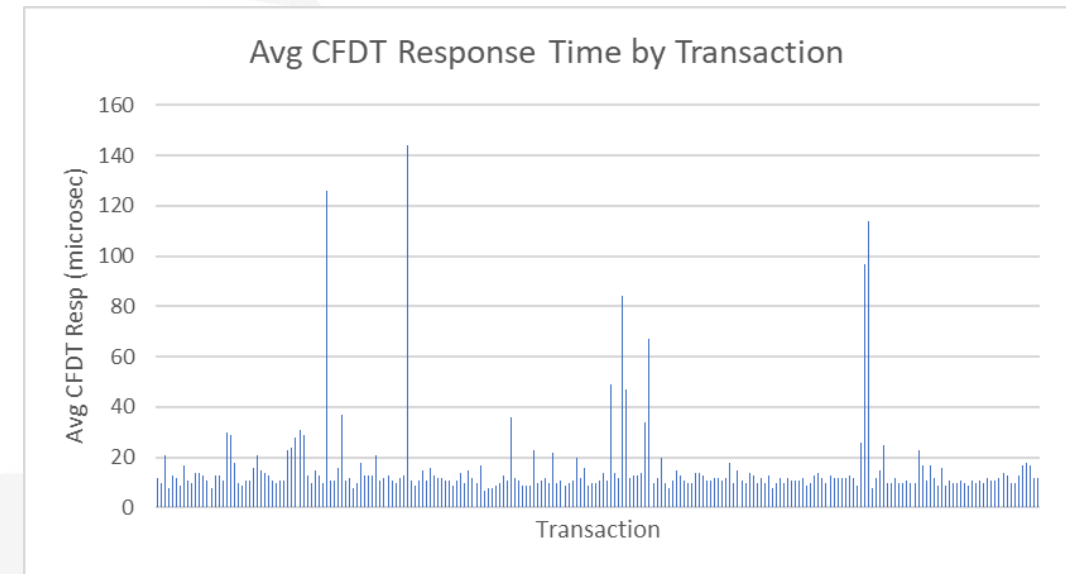
- No big users of TSQs
- *Graphs on this page show average temp stg service times for auxiliary temp storage and shared.*

Performance: Coupling Facility Data Tables

We expected a performance hit from moving to CFDTs for in-memory tables

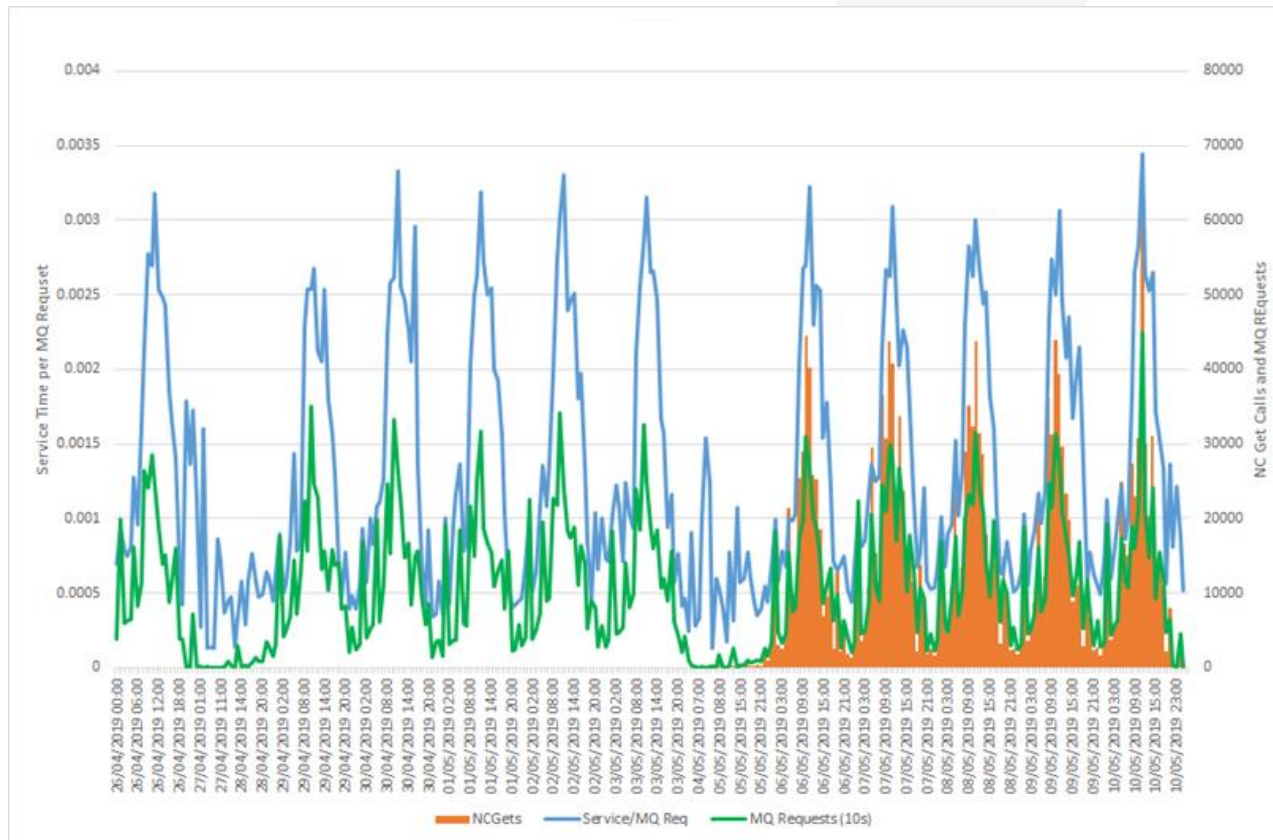


- Mostly reads from CFDT
- CFDT accesses fast
- No measurable impact of using CFDT on transaction service times



Performance: Named Counters

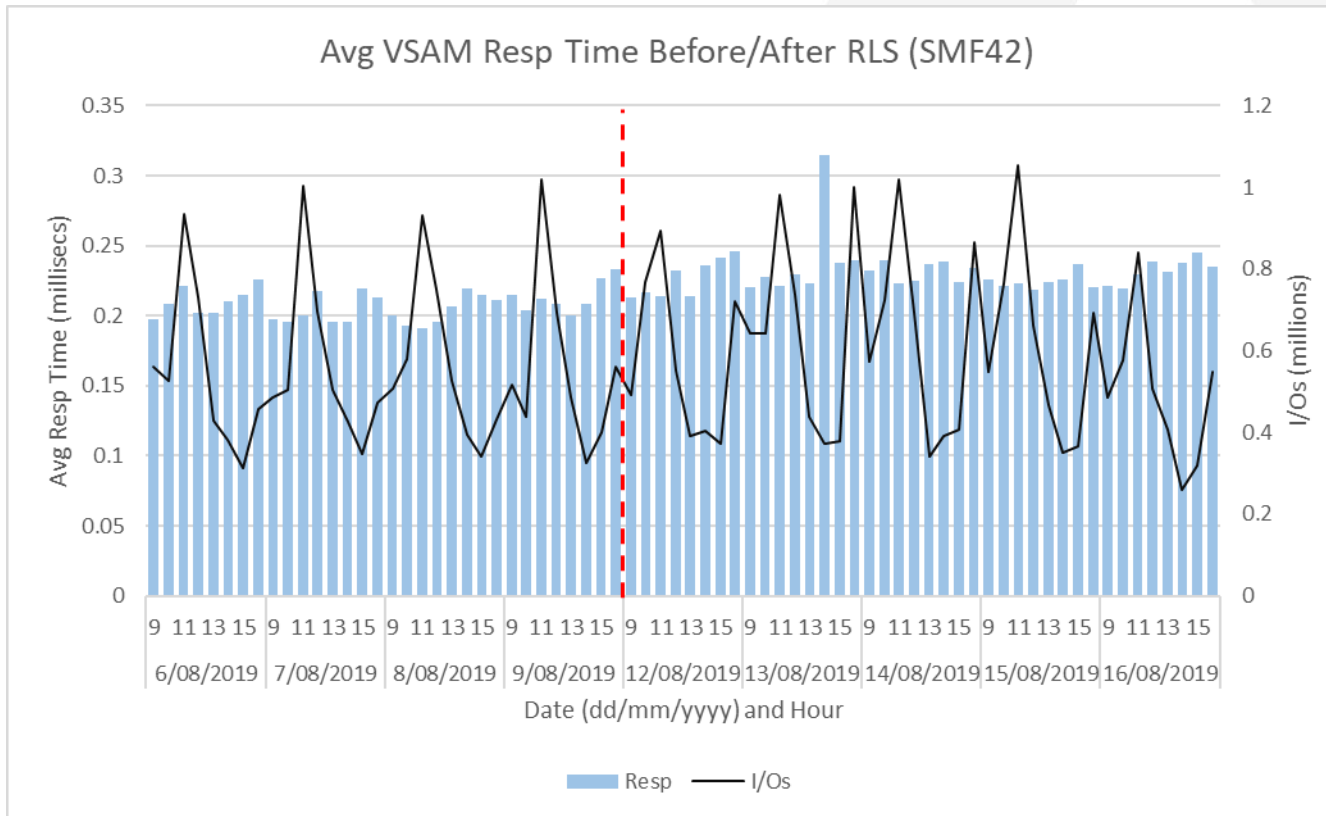
No field measuring named counter delay in SMF110 records



- Looked at overall transaction response times.
- Long running transaction processing MQ: normalized by MQ operation.
- No real difference measured

Performance: VSAM

Converted VSAM from local (LSR) to RLS

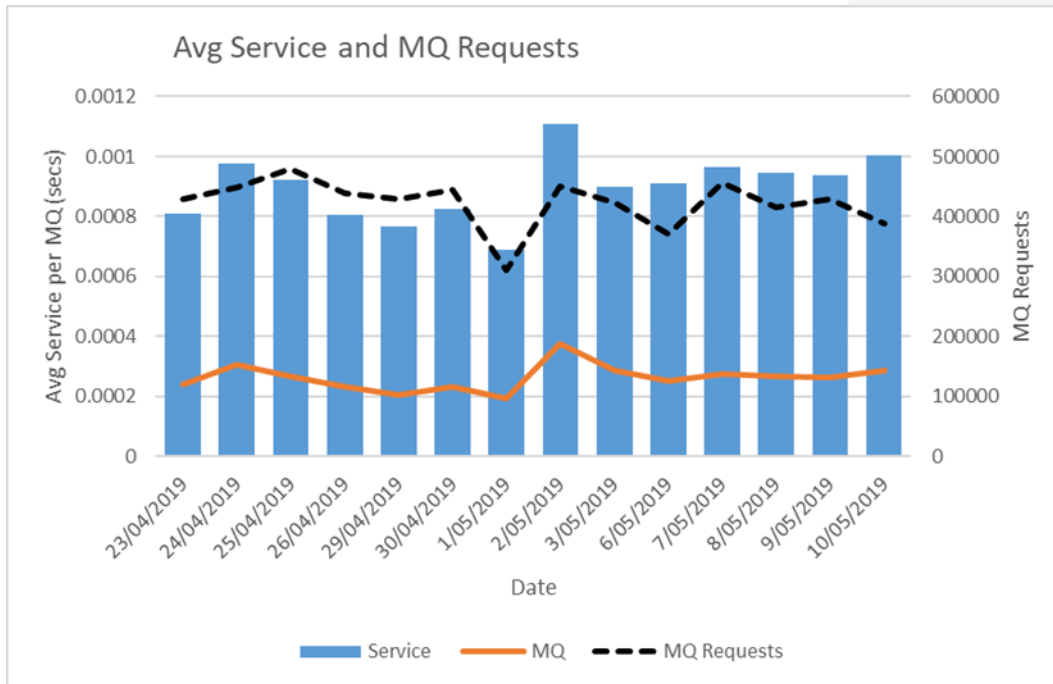


- Looked at SMF 110 transaction statistics for file delays
- Looked at CA SYSVIEW statistics for file-level stats.
- Looked at SMF Type 42 (subtype 6) records for file response times.
- Overall: small increase

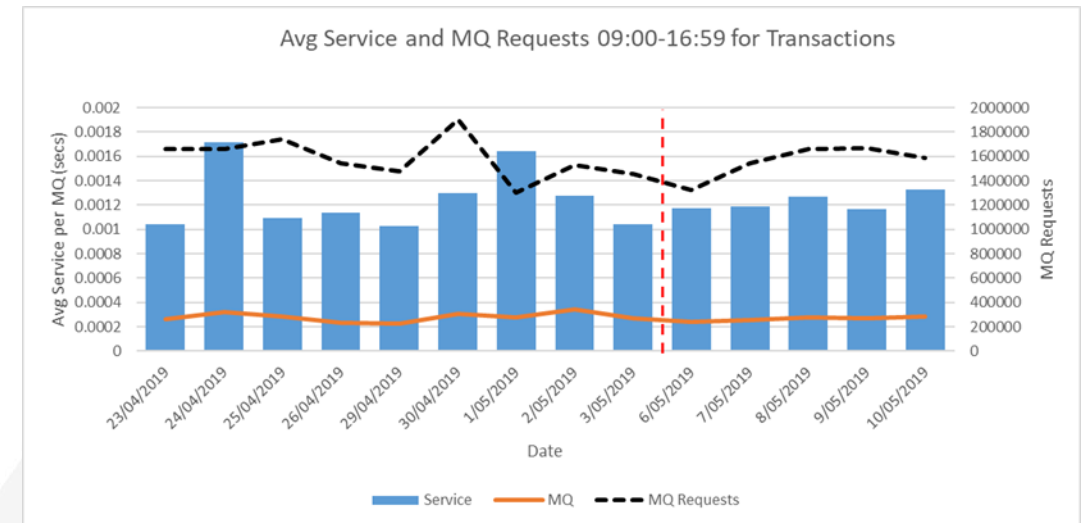
Covered in more detail in our articles at www.longpelaexpertise.com.au/ezone

Performance: MQ

Converted from local to shared MQ queues

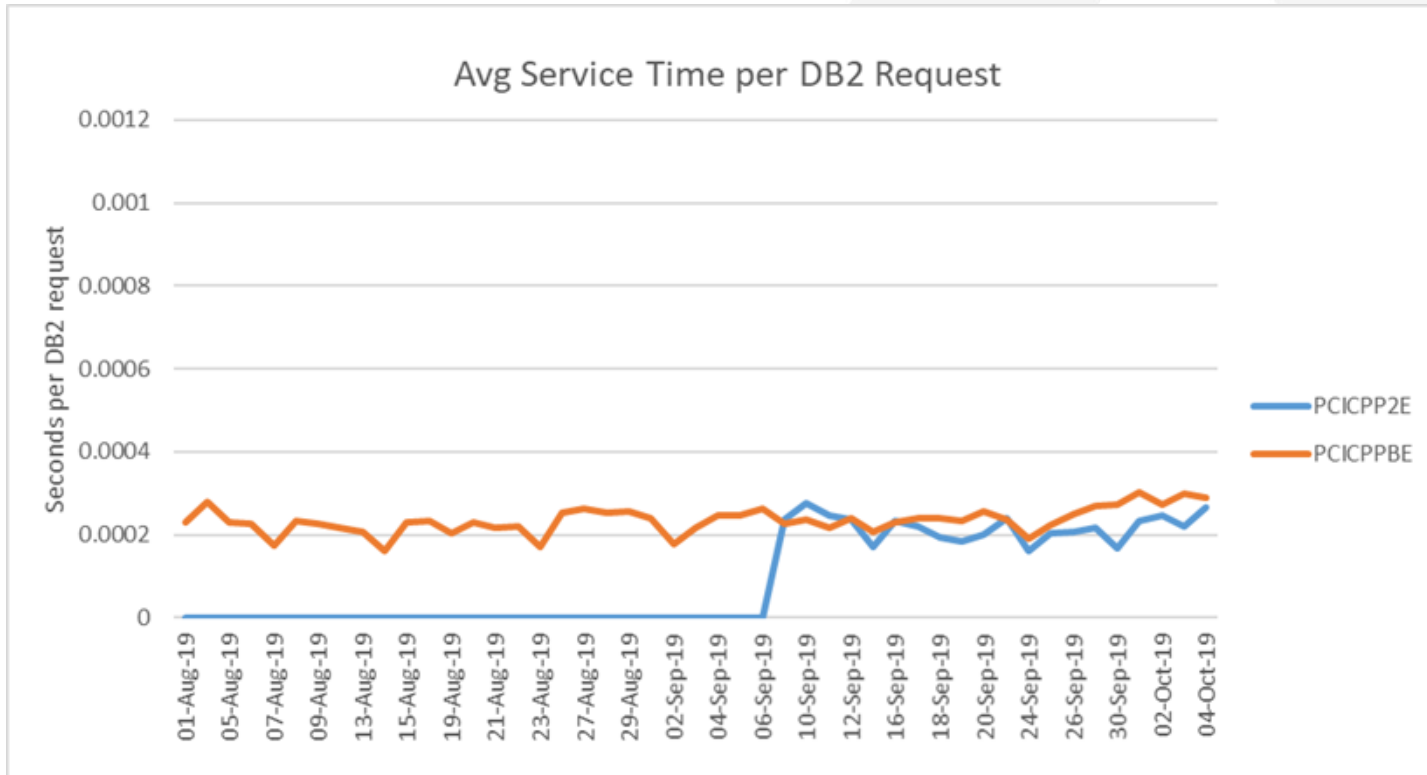


- Almost all persistent MQ queues
- Little performance change from local to shared queues
- *Graphs on this page show average 'net' MQ delays (excluding MQ Getwaits)*



Performance: DB2

DB2 data sharing group was already in place



- Looked at SMF 110 transaction statistics for DB2 delays for both CICS regions before and after
- Little change



WRAP UP

Summary

- Moving to CICSplex is both a system and application project
- Needed lots of coordination / discussion between systems and applications
- Performance was a concern: lots of performance monitoring done
- Load and 'break' testing performed to validate changes and resilience
- Changes implemented slowly to reduce errors
- A few surprises for CPT/Longpela and client
- Project still underway, but successful so far

More about issues we found (MQ, VSAM etc.) in February and May 2020 articles at www.longpelaexpertise.com.au/ezone

Who is David Stephens?

- z/OS systems programmer since 1989
- IBM software developer/L3 change team 2001-2010
- z/OS mainframe systems consultant since 2010

- CPT technical lead on CICSplex project since 2018

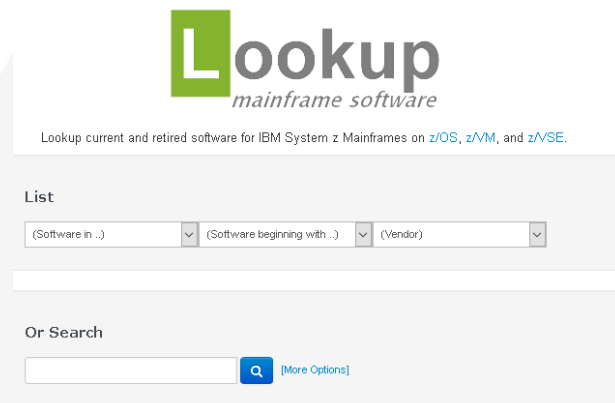
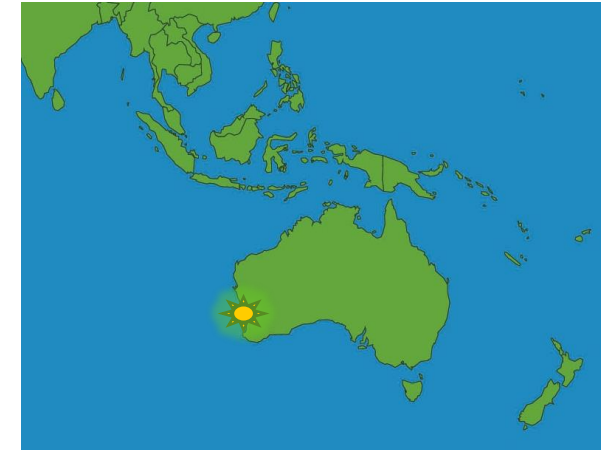
- Loves cricket, travel, red wine
- Lives in Perth, Western Australia
- Works worldwide



Contact at dzs@longpelaexpertise.com.au

About Longpela Expertise

- Small z/OS consulting firm started in 1996
- Based in Perth, Western Australia. Work worldwide
- Systems consultants: z/OS, CICS, IMS and more



Behind
www.lookupmainframesoftware.com
and the book “*What On Earth is a Mainframe*”

<http://www.longpelaexpertise.com.au>

About CPT Global

CPT Global Limited is a highly regarded, specialized technical consulting firm with a reputation build over engagements across the globe. CPT has been engaged by 80% of the world's largest banks. We have delivered outcomes and engagements for clients across 27 countries.

CPT is an established business with a high-profile customer base, including a number of Fortune 500 companies. In our home market Australia, CPT has strong and stable position in the professional services segment of the Australian information technology and digital services market.

CPT has experience working for federal and state government, banking and finance, insurance, telecommunications, retail and manufacturing sectors. We partner with world leading technology partners to bring unique value to our clients.

11 OFFICES
5 CONTINENTS
170+ CONSULTANTS

Hundreds of millions of dollars in measurable IT savings to Fortune 500 and blue-chip companies



<http://www.cptglobal.com>

Your feedback is important!

Submit a session evaluation for each session you attend:

SHARE mobile app -or- www.share.org/evaluation



QR Code for this session

