POL00028911

Callendar Square / Falkirk

Date: 08/02/2019 10:12:00

1. Management Summary

In Jason Coyne's Supplementary Report dated 1st February 2019, he identifies in Table 1 (after Para 3.21 on Page 12) a number of groups of bugs he has identified within Horizon.

Fujitsu has been asked to provide an analysis on the way in which these were handled.

This document describes what he refers to as "Callendar Square / Falkirk bug" which he comments on in para 3.34 to 3.42.

2. Key Information

Info	Response
Source of incident: whether this was customer or internal alerting	The actual Callendar Square issue was reported by the SPMR.
Relevant Peak number used as "master" for 4LS / development to work on	Callendar Square Peaks PC0126042 PC0126376 Other Peaks related to the issue PC0056922 PC0086212 PC0075892 PC0193012 PC0103864 PC0083101
Release Peak number / release ID	
Timescale (report to fix)	**Specific Callendar Square Peaks** **Peak PC0126042** raised 15/09/2005 Problem diagnosed as relating to an earlier issue (E-0509140700) with replicating data between counters on the previous day Passed to MSU to raise a BIMS which would result in POL raising an Error Notice to correct the SPMR's accounts which was done on 16/09/2005. **Peak PC0126376** raised 21/09/2005 Further occurrence of the same issue at the same branch, but this time the SPMR didn't attempt to repeat the missing transactions so no Error Notice required. On 22/09/2005, the Riposte events relating to the failure are identified and the issue is related to "Timeout waiting for lock" events and passed to development. They eventually respond in January 2016 saying the problem will be fixed in S90 (later in 2006) and the Peak is then closed. Given the instructions in the related KELs, it is not clear whether or not SMC had detected the issue and contacted the Branch and advised a restart as indicated in the KEL. **Related Peaks** **PC0056922, PC0075892, PC0083101 and PC0193012** were all closed within about a week

POL00028911

	of being opened **PC0103864** was closed about 2 months after being opened. This refers to KEL CObeng2025L **PC0086212** was closed about 2 years after it was opened as part of a management purge of unresolved Peaks.
KEL (s) identified. Why more than one (if there is)	JSimpkins338Q JBallantyne5245K CObeng2025L
Answers to any contentious points raised by Coyne	See section 3 below
Analysis of Peaks quoted: Which of them truly refer to same issue?	They are all related to different incidents of the same fundamental error message from Riposte.
Analysis of Peaks quoted: Is any of the attached evidence pertinent to conclusion, if so, name it because it will need to be disclosed.	Everything relevant is in the Peaks
Analysis of Peaks quoted: Any QFP decisions on whether to fix	See section 3 below
How we dealt with the problem to get to the root cause	When first spotted in 2000, an avoidance action was identified and this was identified in the KEL. The advice was for SMC to monitor the associated events and then alert the branch. It isn't clear how effective this was.
Analysis of Peaks quoted. Which of them truly refer to same issue	They all relate to the same Riposte error. It isn't clear why this re-occurred in 2010 after the Riposte fix in 2006.
What action we agreed to take to fix the problem	To monitor the issue via SMC and advise a reboot when it occurs. Finally in 2006 a fix was issued.
Scope / impact of issue: Subsequent analysis of scope of incident after root cause determined.	The root cause of all these was a bug in Riposte that had the effect of preventing a counter from writing messages – either those being replicated to it or those generated on that counter. This was not always immediately obvious to the user of the counter. This could result in them thinking that some transactions which had been entered, were missing, and so they attempted to re-enter the transactions on another counter. When the offending counter was re-started, both versions of the transaction become visible and this could cause errors in the accounts. Attempting to balance the branch when a counter was in this state could also result in errors.
Scope / impact of issue: In particular how we identified branches involved and any details of what we passed to Service	The approach was to try and detect the symptoms of the issue (loads of events) and for the SMC monitoring staff to contact the branches recommending a restart of the counter. NB the counter was automatically restarted overnight, so the issue should go away the next day.
Any evidence of subsequent work to ensure that we had the information consolidated.	None.

3. Further Info

This looks like quite a long saga and until now I hadn't realised the full scope of it.

I was previously aware of the "Callendar Square" problem when it came up as part of the Misra case in 2010. At that time I produced a summary which has been quoted by Charles McLachlan in his WS. This was based on conversations with Anne Chambers (SSC) about the issue and looking at Peak PC0126376 (which was discussed in court). As far as the Misra case is concerned it was irrelevant for 3 reasons:

- It was due to issue with transfers between SUs. There was only a single SU in West Byfleet (so there were no Transfers)

POL00028911

- It was fixed in 2006 and the Misra data related to 2007 to 2008
- A check was made in the event logs from West Byfleet, and there was no flood of events (a symptom of the underlying Riposte issue)

However on re-reading Peak PC0126376, I can see it refers to 2 KELs (which I presumably didn't look at back in 2010), which were raised much earlier. This shows that the Riposte issue had been initially identified back in 2000. This is made clear in KEL JBallantyne5245K and the associated Peak PC0056922. This shows that there is a problem in Riposte such that if it loses a Thread which holds a critical lock, then Riposte grinds to a halt and the counter becomes unusable. The avoidance action is to restart the counter. The symptoms of the problem are a large number of events. The Peak advises that if the issue occurs more than once per month, then we would need to try and reproduce the issue. The KEL also refers to PC0083101.

Past experience shows that Escher wouldn't consider bugs if they are not reproducible.

The Peak was then closed and the KEL JBallantyne5245K produced. In particular the KEL advises SMC (who monitor events from counters), that if such events are seen to phone the branch and advise them to re-start the affected counter, and if they are balancing to abandon the balance until the reboot has happened as this prevents replication working correctly.

Meanwhile KEL JSimpkins338Q was raised in May 2002 and seems very similar, but does have a clearer description of the root cause namely the Riposte error "Timeout occurred waiting for lock (0xC1090003)" in an event.

JBallantyne5245K does refer to JSimpkins338Q, but it isn't clear why both KELs were needed. They are now both in the Deleted KELs collection as they refer to errors in Riposte which has not been used since 2010.

This KEL was not triggered by any referenced Peak, but it does identify Peak PC0086212 which was sent to development in January 2003. This in turn identifies Peak PC0075892 which I suspect is what triggered the KEL - it was raised on 2/5/2002 and the KEL was raised on 10/5/2002 which was when the Peak was closed (by me) with insufficient evidence. It also refers to Peak PC0103864.

KEL CObeng2025L was also raised, and refers to KEL JSimpkins338Q. Again I can't see the need for this 3rd KEL related to the issue.

Looking at the issue now and at the associated Peaks and the 2 KELs, there seem to have been quite frequent issues of "Timeout waiting for locks", and the approach taken was for SMC to monitor these events and advice the SMPR to reboot should these events occur, and nothing was done to actually get to the bottom of the issue. Escher seem to have fixed the problem in 2006, although there is a Peak (PC0193012 - referenced by JSimpkins338Q) which was raised in 2010 sometime after the Escher fix and again a reboot fixed the issue.

Responses to points raised in Mr Coyne's report:

1. 3:34: He says: "This leads to a discrepancy in the branch accounts since the double entry principle of accounting is not applied."

   This isn't strictly true. Double Entry Bookkeeping was always applied. However in the case of a Transfer, there are normally 2 separate double entry baskets:

POL00028911

- The Transfer Out from one Stock Unit
- The Transfer In to another Stock Unit

There is also an indicator of all outstanding Transfers. This should be updated when the Transfer In occurs to prevent a further Transfer In. However this update to the Transfer record was not visible to other counters, thus allowing the Transfer I to be repeated. Balancing assumes that if there are no outstanding Transfers, then all Transfers out and Transfers In in a Branch cancel out. However if (as happened in Callendar Square) there were 2 Transfers In for a single Transfer Out, then this affects the cash holding in the branch and results in a loss to the SPMR.

2. I agree with the observation in 3.36 that the underlying issue in Riposte had been present since at least 2000. Avoidance action had been identified in the KELs, but clearly had not been effective in Callendar Square.
3. We do have a spreadsheet obtained from SSC which does identify affected branches (ripostelockprobs.xlsx, below). I don't think this has been declared.

ripostelockprobs.xlsx

4. In 3.40 Mr Coyne comments on inconsistencies in the peak Closure reasons. I think this is covered by Steve Parker's statement, in that they are the best view of the person concerned and not necessarily fully consistent.
5. In 3.41 there is a discussion about whether or not the peak was routed to Escher for a fix. In general, Escher would not accept any defect unless it could be reproduced, and this was one we couldn't reliably reproduce. It is clear from the response in Peak PC0126376 that at some point the issue had been reported to Escher (presumably via some other Peak, not directly linked to this investigation) and that they had identified the issue and provided a fix in the version of Riposte we were due to issue in 2006 as part of S90.
6. I can't really dispute the comments in 3.42.