Database Corruption Challenge #8 has Ended

With many participants and 16 who correctly solved Corruption Challenge week 8, the challenge has come to an end. The overall winner was Rob Farley, and there were several new participants this week. The scores page has been updated.

Some participants called this the toughest challenge yet which makes my job even harder to come up with something more challenging for next time. There were a couple interesting twists in this one. One was the way the corruption caused issues with DBCC Page. On SQL Server 2014 DBCC Page on the corrupt page with the output parameter of 3 didn't work, and the only output option that worked was option 2. However on SQL Server 2008R2 and 2012, none of the DBCC Page output options worked on the corrupt page, which forced some to open the database file with a hex editor to pull the data in from the corrupt page.

The one thing that I didn't judge anyone badly with was the capitalization of the person in the corrupt record. I accepted EMMA, E WILLIAMS, and Emma, E Williams, and Emma, E WILLIAMS all as correct answers, as to figure out any of those it took the same amount of work.

The solution by Rob Farley:

In Record1035, Emma E Williams' record had the problem. Her correct record is:

-9223372036854775435, -9223372036854775688, 10730, EMMA, E, WILLIAMS, Chief, (null), 20141030

I got the values out by just looking in page 249. The structure was easy.

There are some problems with the NCIX too, on both Record1035 (Jakob B Miller) and Record1000 (WA980), but they could just be rebuilt.

(eg, page 244 was corrupt here, where Jakob was messed up as "SMIIob")

000000001153A1F4: 6b6f6241 534d4954 48361902 00000000 00800400 kobASMITH6.....

000000001153A208: f0030019 001a0020 534d4949 6f62424d 494c4c45 ð..... SMIIobBMILLE

000000001153A21C: 52365b01 00000000 00800400 f0030019 001a001f R6[.....ð.....

And page 191 was corrupt here:

000000001153AF78: 19001b00 23005741 38383157 41323031 31303732
....#.WA881WA2011072

000000001153AF8C: 38361703 00000000 00800400 f0030019 001b0023 86.....ð.....#

000000001153AFA0: 00574138 38365741 32303132 30343231 361c0300
.WA886WA201204216...

000000001153AFB4: 00000000 800400f0 03001900 1b002300 57413935 .....ð.....#.WA95

000000001153AFC8: 37574132 30313330 36323536 22030000 00000080 7WA201306256".....

000000001153AFDC: 0400f003 3132311b 00230057 41393830 57413230
..ð.121..#.WA980WA20

000000001153AFF0: 31323131 30363623 03000000 00008004 00f00300 1211066#.....ð..

000000001153B004: 19001b00 23005741 38393457 41323031 30303232
....#.WA894WA2010022

Anyway, I think that’s about it. I recreated the NCIX on Record1000, and truncated Record1035 once I had its rows in a different spot.

The checksums submitted were.

1000	526	0	72766	1051256	0	0	0	53

1000	0	0	877708	-2030100812	2	-1756609606	0	0	5920

And that is how Rob solved it.

Nice work to everyone involved. I will post some other solutions over the next week.

Related Links

- Current Scores
- Newsletter Sign up
- Database Corruption Challenge T-Shirt
- Week 1, Week 2, Week 3, Week 4, Week 5, Week 6, Week 7, Week 8
- Slow DBCC CheckDB
- Database Corruption Worksheet