

WHEN DATABASE CORRUPTION STRIKES

Presented by Steve Stedman

Founder/Owner of Stedman Solution, LLC.

About Steve

- Blog regularly at <http://SteveStedman.com>
- Founder of the [Database Corruption Challenge](#)
- I work on about 1 actual corrupt database per week
- Author SQL Server – [Common Table Expressions Book](#)
- Creator of [Database Health Monitor](#) (<http://DatabaseHealth.com>)
- 25 years of SQL Server experience
- Freelance Database Consultant (SQL Server)

Database Corruption Challenge

- An about weekly event. I create a corrupt database and confirm that it can be recovered.
- Corrupt database is posted. Challenge runs for 3 to 4 days. Participants send me their solutions.
- Winner are posted. Prizes will be awarded at the end of the 10 weeks.
- After the 10 weeks are up, another challenge will begin.
- To participate signup at SteveStedman.com, click the newsletter signup to participate.

Agenda

- Detecting Corruption
- Tracking Corruption (what has gone bad?)
- Before fixing or removing corruption
- Removing Corruption
- Examples

Detecting Corruption

- DBCC CheckDB

```
DBCC CheckDB();
```

100 %

Messages

There are 0 rows in 0 pages for object "sys.sqlagent_jobs".
DBCC results for 'sys.sqlagent_jobsteps'.
There are 0 rows in 0 pages for object "sys.sqlagent_jobsteps".
DBCC results for 'sys.sqlagent_job_history'.
There are 0 rows in 0 pages for object "sys.sqlagent_job_history".
DBCC results for 'sys.sqlagent_jobsteps_logs'.
There are 0 rows in 0 pages for object "sys.sqlagent_jobsteps_logs".
Msg 8944, Level 16, State 13, Line 1
Table error: Object ID 245575913, index ID 1, partition ID 72057594040614912, alloc un
Msg 8944, Level 16, State 13, Line 1
Table error: Object ID 245575913, index ID 1, partition ID 72057594040614912, alloc un
Msg 8928, Level 16, State 1, Line 1
Object ID 245575913, index ID 1, partition ID 72057594040614912, alloc unit ID 7205759
Msg 8976, Level 16, State 1, Line 1
Table error: Object ID 245575913, index ID 1, partition ID 72057594040614912, alloc un
DBCC results for 'Revenue'.
There are 27 rows in 1 pages for object "Revenue".
CHECKDB found 0 allocation errors and 4 consistency errors in table 'Revenue' (object
DBCC results for 'sys.queue_messages_1977058079'.

100 %

Query completed with errors. | sql14 (12.0 RTM) | sa (54) | CorruptionChallenge1 | 00:00:04 | 0 rows

Detecting Corruption

- DBCC CheckDB
- DBCC CheckTable

```
DBCC CheckTable(Revenue);
```

100 %

Messages

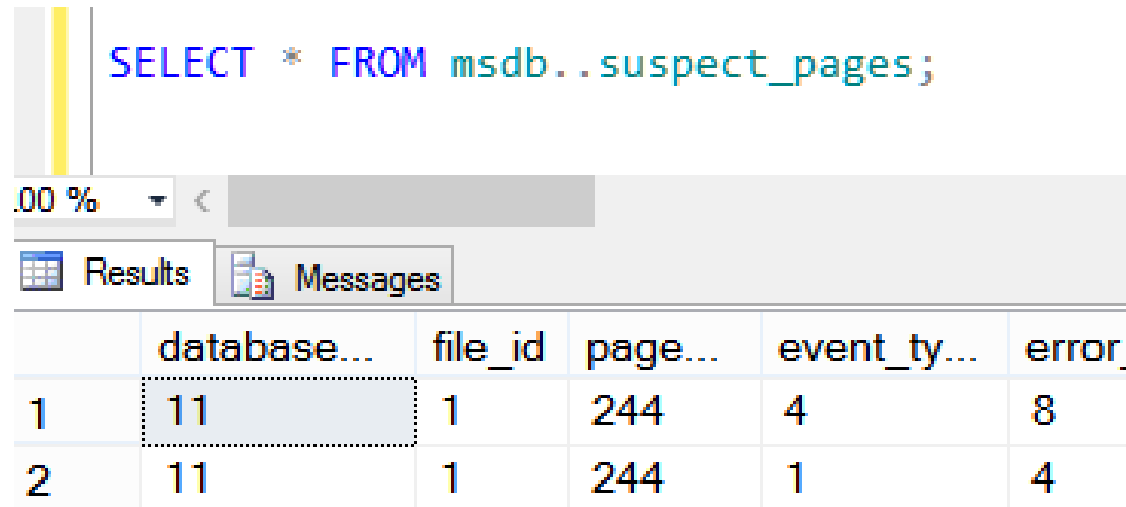
Msg 8944, Level 16, State 13, Line 8
Table error: Object ID 245575913, index ID 1, partition ID 72057594040614912, alloc un
Msg 8944, Level 16, State 13, Line 8
Table error: Object ID 245575913, index ID 1, partition ID 72057594040614912, alloc un
Msg 8928, Level 16, State 1, Line 8
Object ID 245575913, index ID 1, partition ID 72057594040614912, alloc unit ID 7205759
Msg 8976, Level 16, State 1, Line 8
Table error: Object ID 245575913, index ID 1, partition ID 72057594040614912, alloc un
DBCC results for 'Revenue'.
There are 27 rows in 1 pages for object "Revenue".
CHECKTABLE found 0 allocation errors and 4 consistency errors in table 'Revenue' (obje
repair_allow_data_loss is the minimum repair level for the errors found by DBCC CHECKT
DBCC execution completed. If DBCC printed error messages, contact your system administ

Detecting Corruption

- DBCC CheckDB
- DBCC CheckTable
- DBCC Check_____
- Constraints, Catalog, Alloc, FileGroup, Ident

Detecting Corruption

- DBCC CheckDB
- DBCC CheckTable
- DBCC Check_____
- Constraints, Catalog, Alloc, FileGroup, Ident
- msdb..suspect_pages

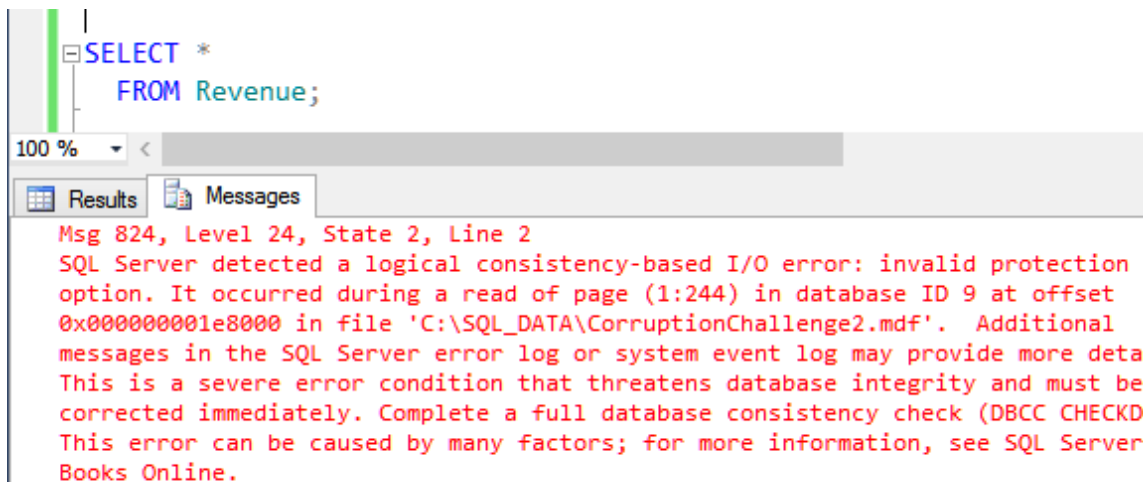


The screenshot shows a SQL Server Enterprise Manager interface. At the top, a query window displays the command: `SELECT * FROM msdb..suspect_pages;`. Below the query window, a progress bar indicates 0.00 % completion. Two tabs are visible: 'Results' and 'Messages'. The 'Results' tab is active, showing a table with the following data:

	database...	file_id	page...	event_ty...	error...
1	11	1	244	4	8
2	11	1	244	1	4

Detecting Corruption

- DBCC CheckDB
- DBCC CheckTable
- DBCC Check_____
- Constraints, Catalog, Alloc, FileGroup, Ident
- msdb..suspect_pages
- Just running a query may show corruption.



Tracking Corruption (what has gone bad?)

- Error messages and the error log

Selected row details:

Date	5/10/2015 4:20:36 PM
Log	SQL Server (Archive #1 - 5/10/2015 8:53:00 PM)
Source	spid52

Message

SQL Server detected a logical consistency-based I/O error: incorrect pageid (expected 1:9; actual 0:0). It occurred during a read of page (1:9) in database ID 8 at offset 0x000000000012000 in file 'C:\SQL_DATA\CorruptionChallenge5.mdf'. Additional messages in the SQL Server error log or system event log may provide more detail. This is a severe error condition that threatens database integrity and must be corrected immediately. Complete a full database consistency check (DBCC CHECKDB). This error can be caused by many factors; for more information, see SQL Server Books Online.

Tracking Corruption (what has gone bad?)

- Error messages and the error log
- See what you can query

-- lets see what we have in the corrupt table

SELECT *

FROM Revenue;

-- 54 rows

Tracking Corruption (what has gone bad?)

- Error Messages and the Error Log
- See what you can query
- Check your non-clustered indexes

```
-- pull from the non-clustered index without  
-- touching the clustered index
```

```
SELECT [id], [DepartmentID], [Revenue]  
FROM Revenue  
WITH (INDEX (ncDeptIdYear) );
```

Before Fixing or Removing Corruption

- Do you have a way to start over if something goes wrong?
- Do you have a backup of the current state?
- If your solution is going to cause data loss, what can you save anything before causing that data loss?
- Do you have someone to review your ideas before proceeding?
- What if you go through the whole process, but determine that part of your cleanup dropped a row that could have been saved in the beginning. Can you go back and do it again?

Removing Corruption

- Drop/Recreate Index – if corruption is in a non-clustered index
- Restore from backup
 - Consider a tail of the log backup
- Truncate table – if you have a way to get the contents back
- `DBCC CheckTable(Revenue, REPAIR_REBUILD);`
- `DBCC CheckTable(Revenue, REPAIR_ALLOW_DATA_LOSS);`
- `DBCC CheckDB(database1, REPAIR_ALLOW_DATA_LOSS);`

Example 1

- You are given a .bak file with a corrupt database.
- That's it... That is all you have access to.
- The original database was already destroyed by a network admin attempting to solve the problem.
- Earlier non-corrupt backups were overwritten with the corrupt backup.

Example 1 – DBCC CheckDB

```
DBCC CheckDB(CorruptionChallenge1) WITH NO_INFOMSGS;
```

100 %



Messages

Msg 8944, Level 16, State 13, Line 2

Table error: Object ID 245575913, index ID 1, partition ID 72057594040614912, alloc unit ID 72057594045857792 (type In-row data), page (1:280), row 3. Test (ColumnOffsets <= (nextRec - pRec)) failed. Values are 3139 and 288.

Msg 8944, Level 16, State 13, Line 2

Table error: Object ID 245575913, index ID 1, partition ID 72057594040614912, alloc unit ID 72057594045857792 (type In-row data), page (1:280), row 3. Test (ColumnOffsets <= (nextRec - pRec)) failed. Values are 3139 and 288.

Msg 8928, Level 16, State 1, Line 2

Object ID 245575913, index ID 1, partition ID 72057594040614912, alloc unit ID 72057594045857792 (type In-row data): Page (1:280) could not be processed. See other errors for details.

Msg 8976, Level 16, State 1, Line 2

Table error: Object ID 245575913, index ID 1, partition ID 72057594040614912, alloc unit ID 72057594045857792 (type In-row data). Page (1:280) was not seen in the scan although its parent (1:281) and CHECKDB found 0 allocation errors and 4 consistency errors in table 'Revenue' (object ID 245575913).

CHECKDB found 0 allocation errors and 4 consistency errors in database 'CorruptionChallenge1'.

repair_allow_data_loss is the minimum repair level for the errors found by DBCC CHECKDB

(CorruptionChallenge1).

Stop - Think

- What would I do first?
- If my work makes something worse, how can I undo what I try?
- What can I do to determine what is corrupt?
- What can I do to recover missing data caused by the corruption?
- Demo: Example 1



Example 2

- The story... A backup was run yesterday, after that backup several changes were made to the database, inserts deletes, updates, and more. Sometime today corruption was encountered when someone attempted to run the following query:

```
SELECT *  
FROM Revenue;
```

Example 2 – Additional Details

- You have two backups, the latest is a backup of the current corrupt database.
- The Two Days Ago Backup is the only other backup file available, and it does not contain any corruption.

Name	Type
 CorruptionChallenge2_LatestBackup.bak	BAK File
 CorruptionChallenge2_TwoDaysAgoBackup.bak	BAK File

Example 2 - Errors

```
USE CorruptionChallenge2;  
GO
```

```
SELECT *  
FROM Revenue;
```

100 %

Results Messages

Msg 824, Level 24, State 2, Line 1

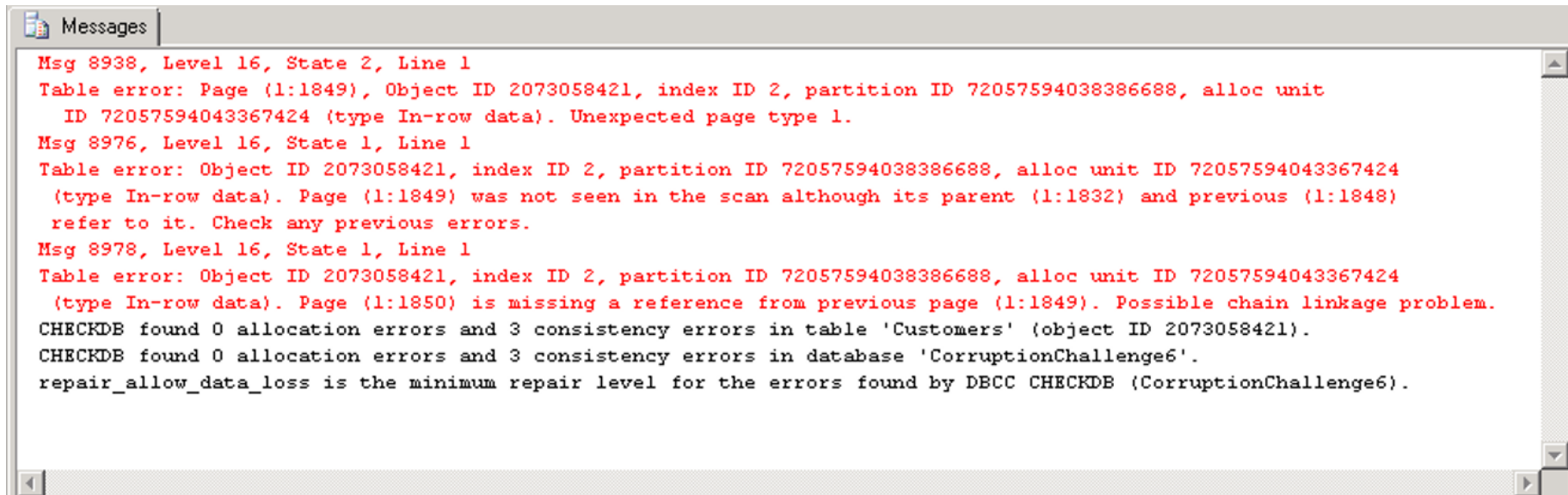
SQL Server detected a logical consistency-based I/O error: invalid protection option. It occurred during a read of page (1:244) in database ID 9 at offset 0x000000001e8000 in file 'C:\SQL_DATA\CorruptionChallenge2.mdf'. Additional messages in the SQL Server error log or system event log may provide more detail. This is a severe error condition that threatens database integrity and must be corrected immediately. Complete a full database consistency check (DBCC CHECKDB). This error can be caused by many factors; for more information, see SQL Server Books Online.

Stop - Think

- What would I do first?
- If my work makes something worse, how can I undo what I try?
- What can I do to determine what is corrupt?
- What can I do to recover missing data caused by the corruption?
- Demo: Example 2

Example 3

- This was Week 6 in the Database Corruption Challenge.
- The story, corruption was encountered running DBCC CheckDB that looked like this:

A screenshot of a SQL Server Messages window. The window has a title bar with a small icon and the text "Messages". The main area contains several lines of text in red, which are error messages from a DBCC CHECKDB operation. The messages are as follows:

```
Msg 8938, Level 16, State 2, Line 1
Table error: Page (1:1849), Object ID 2073058421, index ID 2, partition ID 72057594038386688, alloc unit
ID 72057594043367424 (type In-row data). Unexpected page type 1.
Msg 8976, Level 16, State 1, Line 1
Table error: Object ID 2073058421, index ID 2, partition ID 72057594038386688, alloc unit ID 72057594043367424
(type In-row data). Page (1:1849) was not seen in the scan although its parent (1:1832) and previous (1:1848)
refer to it. Check any previous errors.
Msg 8978, Level 16, State 1, Line 1
Table error: Object ID 2073058421, index ID 2, partition ID 72057594038386688, alloc unit ID 72057594043367424
(type In-row data). Page (1:1850) is missing a reference from previous page (1:1849). Possible chain linkage problem.
CHECKDB found 0 allocation errors and 3 consistency errors in table 'Customers' (object ID 2073058421).
CHECKDB found 0 allocation errors and 3 consistency errors in database 'CorruptionChallenge6'.
repair_allow_data_loss is the minimum repair level for the errors found by DBCC CHECKDB (CorruptionChallenge6).
```

Example 3 - Details

- Someone had disabled the SQL Server Agent, so there were no recent automated backups run on this database.
- After the corruption was encountered, a backup was performed in order to distribute the corrupt database for the competition.
- There were no transaction log backups, only the single full database backup that contains the corruption.

Example 3

- There is an old joke
 - “You either need to have good backup and recovery or a good resume, pick one”
 - Is there a third option?
- In this case there are no backups, so all you have is your skills...
 - Your skills to fix the database, not your skills to write a resume.

Stop - Think

- What would I do first?
- If my work makes something worse, how can I undo what I try?
- What can I do to determine what is corrupt?
- What can I do to recover missing data caused by the corruption?
- Demo: Example 2

Example 4 - On Your Own

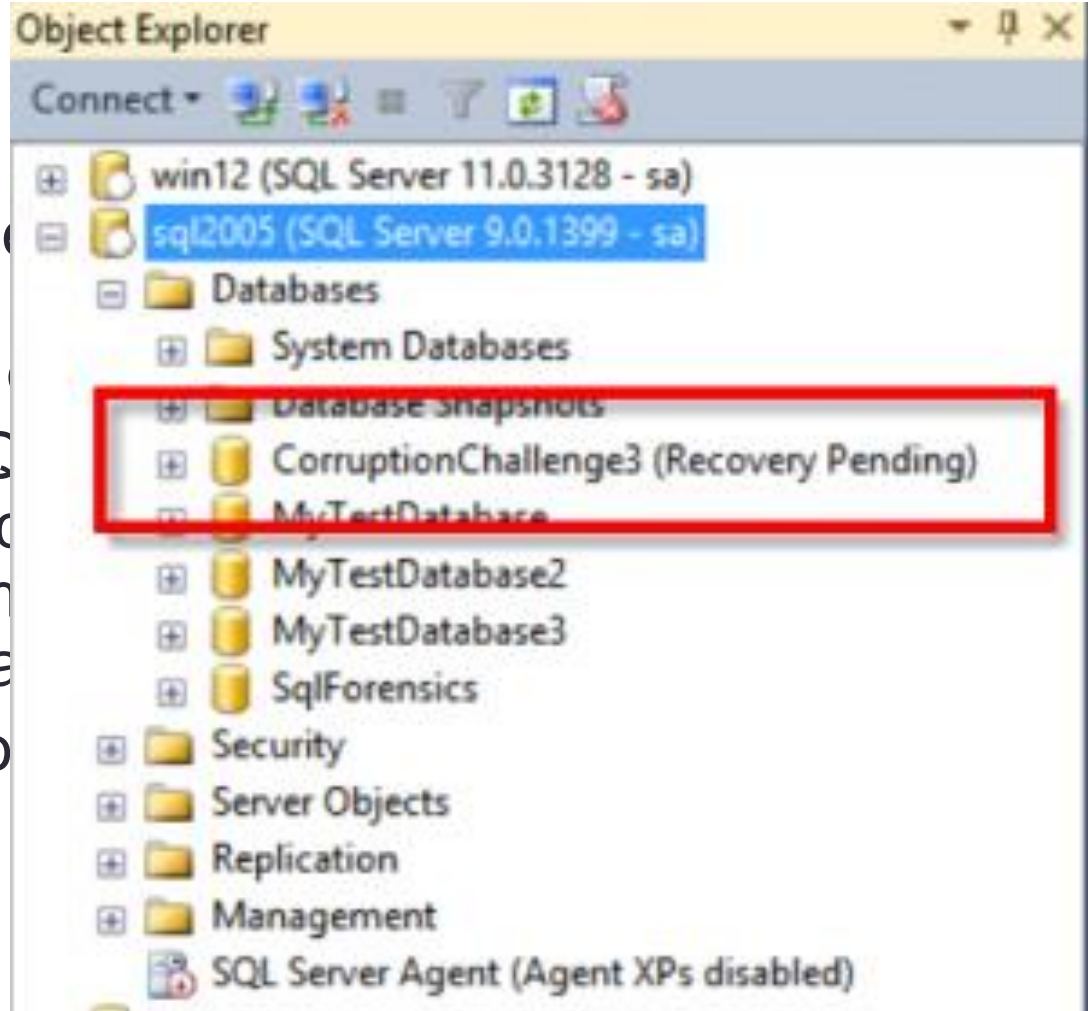
- This was week 3 in the Database Corruption Challenge
- <http://stevestedman.com/fFQvC>
- Download the challenge and try it on your own.

Example 4 – Timeline

- 2:53pm - full backup
- 2:54pm - transaction log backup
- 3:01pm - transaction log backup
- 3:12pm - transaction log backup
- 3:19 a catastrophic failure occurred causing the operating system running SQL Server to reboot.
 - Let's blame it on a power surge, or maybe it was just someone who accidentally bumped the power cord. I like the power surge idea, so that I don't get blamed for kicking the power cord.

Example 4

- After the server rebooted, SQL Server recovered CorruptionChallenge3.
- After examining the SC, it discovered that the .mdf file was corrupted. The only thing that remained was the .ldf file and previous backup.
- This was caused by software corruption of the storage.
- The database is in the



Stop - Think

- What would I do first?
- If my work makes something worse, how can I undo what I try?
- What can I do to determine what is corrupt?
- What can I do to recover missing data caused by the corruption?
- Demo: Example 3

Summary

- Detecting Corruption
- Tracking Corruption (what has gone bad?)
- Before fixing or removing corruption
- Removing Corruption
- Examples

More Examples

- More Corruption Challenges available on my website.
- My website: <http://SteveStedman.com/Corruption>
- Connect
 - Twitter @SqlEmt
 - LinkedIn <http://linkedin.com/in/stevestedman>
 - Email Steve@StedmanSolutions.com