SHARE26 August 16-20, 2026 PITTSBURGH, PA

# My All Time Db2 Top Ten Lists

Session ID: TECH209s1

Craig S. Mullins
Mullins Consulting, Inc.
www.mullinsconsulting.com

# The Top Ten Lists

And now, from the home office in Sugar Land, Texas… a series of Db2 Top Ten lists about various topics ranging across many different subjects:

- Performance
- Coding
- Design
- Administration
- Management
- Features
- Tools

# Top Ten SQLCODEs to Memorize

1. 000 / +100 successful / “no more rows”
2. -904 resource unavailable
3. -818 timestamp mismatch
4. -101 SQL statement too complex
5. -104 illegal symbol in SQL stmt.
6. -530 / -532 RI constraint violation
7. -803 unique violation (duplicate data)
8. -911 / -913 deadlock or timeout
9. -922 authorization failure
10. -811 more than 1 row for a singleton SELECT

**SQLCA**
**SQLERRD(3)**
00C90088 - deadlock
00C9008E - timeout

# Top Ten Db2 12 Features

1. Continuous Delivery
2. New Global Variables for Continuous Delivery
   - PRODUCTID_EXT – version, release, FL
   - CATALOG_LEVEL
3. Piece-Wise Deletion
   - FETCH FIRST n ROWS ONLY on DELETE
4. RPN Table Spaces
   - 1 TB per partition.
   - Max table size 4 PB
     - with up to 256 trillion rows per table
5. FTBs
6. SQL Pagination (OFFSET syntax)
   - e.g.) OFFSET 10 ROWS
   - causes the first 10 qualifying rows to be skipped
7. Fast INSERT (Insert Algorithm 2)
8. Improved MERGE
9. Improved Profiles
   - Can start then when Db2 starts
   - More global variables
   - Wildcarding of AUTHID, LOCATION, PRDID
10. FL508: ALTER TABLESPACE MOVE TABLE (convert to UTS)


# Top Ten Db2 13 Features

1. SQL Data Insights
2. Utility History in the Db2 Catalog
3. Online REORG to convert PBG to PBR
4. Application-specific lock timeout control *
   - `SET CURRENT LOCK TIMEOUT`
5. Application-specific deadlock detection
   - `SET SYSIBMADM.DEADLOCK_RESOLUTION_PRIORITY`
6. PROFILE support for #4 and #5
7. FTBs supported for indexes w/larger key lengths**
8. Autobind phase-in (FL504)
   - `DEPLEVEL(STATEMENT)`
9. Insert multiple rows using multiple VALUES clauses in one INSERT statement (FL506)
10. Improved clearing of data partitions using LOAD (FL506)
    - `NOKEYDELETE`

** double

* https://db2portal.blogspot.com/2023/03/consider-application-level-lock-control.html

```
INSERT INTO DSN8D10.EMP
  (EMPNO,FIRSTNME,MIDINIT,LASTNAME,WORKDEPT,PHONENO,
   HIREDATE,JOB,EDLEVEL,SEX,BIRTHDATE,SALARY,BONUS,COMM)

   VALUES
     ('000206','ELIZABETH','T','GRACE','D11','2866',
      '2023-02-17','ANALYST',16,'F','1975-06-28',
      16345,500,2300),
     ('000207','JACK','Q','JOHNSON','D11','2867',
      '2023-08-10','ANALYST',16,'M','1979-07-22',
      16345,500,2300),
     ('000208','JENNIFER','K','WHITE','D11','2868',
      '2023-08-10','ANALYST',16,'F','1980-08-13',
      16345,500,2300);
```

# Top Ten Most Significant Features in Db2 History

1. Packages (V2.3)
2. Data Sharing (V4)
3. Referential Integrity (V2.3)
4. Type 2 Indexes (V4)
5. Universal Table Spaces (V9) (segmented were V2.3)
6. Continuous Delivery (V12)
7. Stored Procedures (V4)
8. Multiple Buffer Pools (V3...)
9. Online Schema Change (V8...)
10. DATE / TIME data types (V1.3)

# Top Ten Common SQL Mistakes

1. Assuming an ORDER BY is not necessary for ordered results
2. Forgetting the NULL indicator
3. Coding predicates appropriately in Outer Joins
4. Not coding a cursor for a multi-row result
5. Recompiling but not rebinding
6. Forgetting to use single quotes around strings (instead of double quotes)
7. Trying to modify a Primary Key column
8. Using cursors improperly
   - SELECT from a closed cursor (-501), OPEN an open cursor (-502)
9. Asking for more data than you need (columns and/or rows)
   - Sometimes (erroneously) referred to as the SELECT * problem
10. Incorrect expectations when using the NOT IN predicate with NULLs ‡

‡
```
SELECT C.color
FROM   Colors AS C
WHERE  C.color NOT IN (SELECT P.color
                       FROM Products AS P);
```

# What's Wrong With This SQL?

```
SELECT   LAST_NAME,   FIRST_NAME,
         JOB_CODE,   DEPT,   PHONENO
FROM     DSN8D10.EMP
WHERE    JOB_CODE = 'A'
AND      DEPT = 'MIS';
```

Example of #9 from previous slide

# Top Ten Overall Tips to Ensure Efficient Db2 Performance


(with links)

- Analyze Query Execution Plans
- Optimize Your Database Design
- Efficient Indexing Strategies
- Effective Memory Configuration
- Workload Management & Query Optimization
- Monitoring & Performance Metrics
- Configuration Parameters & Tuning
- I/O Optimization
- Regular Maintenance & Housekeeping
- Stay Up-to-Date

https://db2portal.blogspot.com/2023/12/my-top-ten-db2-performance-posts.html

# Top Ten Rules of Thumb For Db2 Professionals

1. Almost Never Say “Always” or “Never”
2. Don’t Panic!
3. Knowing Where to Look > Memorizing
4. It Depends!
5. Tuning is Iterative
6. Remember the 80/20 Rule
7. All Tuning Optimizes: CPU, I/O or Concurrency
8. Importance vs. Frequency
   - We’ll discuss this more on the next slide (Top 10 list)
9. Be Careful of Standards
10. Always Be Learning

# Top Ten Most Important DBA Tasks

1. Data Integrity and Accuracy
2. Availability
3. Backup (and Recovery)
4. Database Design
5. Security and Data Protection
6. Performance Management
7. Change Management
8. DBMS Maintenance
9. Troubleshooting
10. Education and Mentoring

# Top Ten Most Important  Development Tactics

1. Code Relationally
   - Unlearn the “flat file” mentality
2. Minimize Data Access
   - Passes thru data and network calls
3. Put the work in the SQL not the program code
4. Minimize Coding
5. Understand and code for indexes
6. Sequential versus Random access
7. Optimization: coding, stages, filter factors, EXPLAIN
8. Always check  SQLCODE/SQLSTATE
9. Code for Concurrency
   - Understand Db2 Locking
10. Avoid Bachelor Programming Syndrome →

# Bachelor Programming Syndrome

Fear of COMMITing

- **Plan and Implement a COMMIT strategy**
  - Or experience TIMEOUTs and lock escalation


# Top Ten Most Common Db2 Problems Overall *IMHO*

1. PEBCAK
2. Poorly coded SQL
3. Improper indexing
4. RUNSTATS not up-to-date (or not even run)
5. Bachelor programming syndrome
6. Bad program design
7. REORG needed (index / table space)
8. Improperly defined buffer pools
9. Copied code syndrome
10. Improperly designed database structures

# Top Ten Steps to Proper Indexing

1. Index by workload, not by object
2. Build indexes based on predicates
3. Index most-heavily used queries
4. Index important queries
5. Index to avoid sorting
   - GROUP BY, ORDER BY
6. Create indexes for uniqueness (PK, U)
7. Create indexes for foreign keys
8. Consider adding columns for IXO access
9. Don't arbitrarily limit number of indexes
   - Unused Indexes? SYSIBM.SYSINDEXSPACESTATS.LASTUSED
10. Be aware of I/U/D implications

https://www.dbta.com/Columns/DBA-Corner/Top-10-Steps-to-Building-Useful-Database-Indexes-100498.aspx

# Index Only Example

```
SELECT   DEPTNAME
FROM     DSN8D10.DEPT
WHERE    DEPTNO = 'D01';
```

- One index exists on DEPTNO
  - The query will likely use this index
  - But Db2 will have to go to the table for the DEPTNAME
- Could encourage IXO by adding the DEPTNAME column to the index
  - DEPTNO, DEPTNAME

# Top Ten Things I Commonly Find

1. Lock escalation
2. Lock timeouts and deadlocks
3. RID failures
4. Outdated RUNSTATS
5. Not exploiting zIIPs
6. Outdated DSNZPARM values
7. Working in the Past
   - Unaware of newer Db2 features
8. Lack of declarative RI in Db2 DDL
9. Duplicate key encountered (-803)
10. Fear of Change


# Top Ten Most Common Physical DB2 Database Design Mistakes

1. Relying on the defaults
2. Not basing the physical on a logical model
3. Over-relying on the logical model
4. Clustering choice
5. Too much (or not enough) free space
6. Improper data types
   - Too large/small, dates in CHAR, etc.
7. Poor indexing strategy
   - Indexing by table, not by workload
8. Failing to plan for data purging or archiving
   - And when do you create your backup/recovery plan?
9. Failure to share data
   - Not Data Sharing, but sharing data!
10. Kludging

# Top Ten Most Common Misunderstandings About Db2

1. “There’s a problem with Db2!”
2. Using nulls can save space
3. DB2 is a “database”
4. Db2 is outdated!
5. SQL is simple to learn and code (*properly*)
6. If it uses an index it doesn’t need ORDER BY
7. Db2 for z/OS is too expensive.
8. Using BP0 only performs OK
9. Db2 for z/OS only runs old, batch jobs
10. It depends!

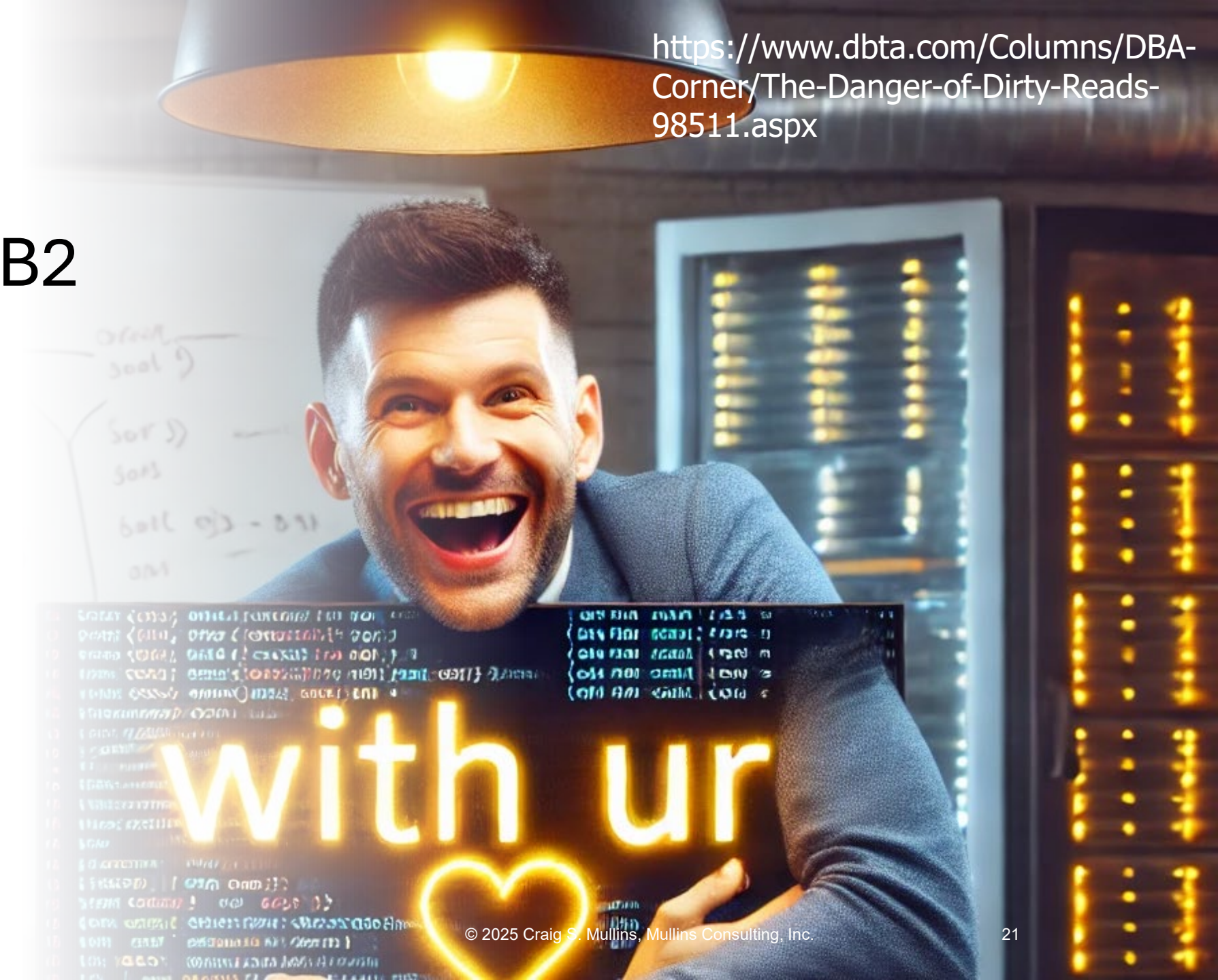
# Top Ten Most Under-Utilized Features of DB2

1. Table Expressions
2. CASE statements
3. Profile tables
4. Recursive SQL
5. Triggers and UDFs
6. IDAA
7. DISTINCT Types
8. Index compression
9. Date/Time Data Types
10. EXPLAIN (by developers)

# Top Ten Most Over-Utilized Features of DB2

1. WITH UR
2. WITH UR
3. WITH UR
4. WITH UR
5. WITH UR
6. WITH UR
7. WITH UR
8. WITH UR
9. WITH UR
10. WITH UR

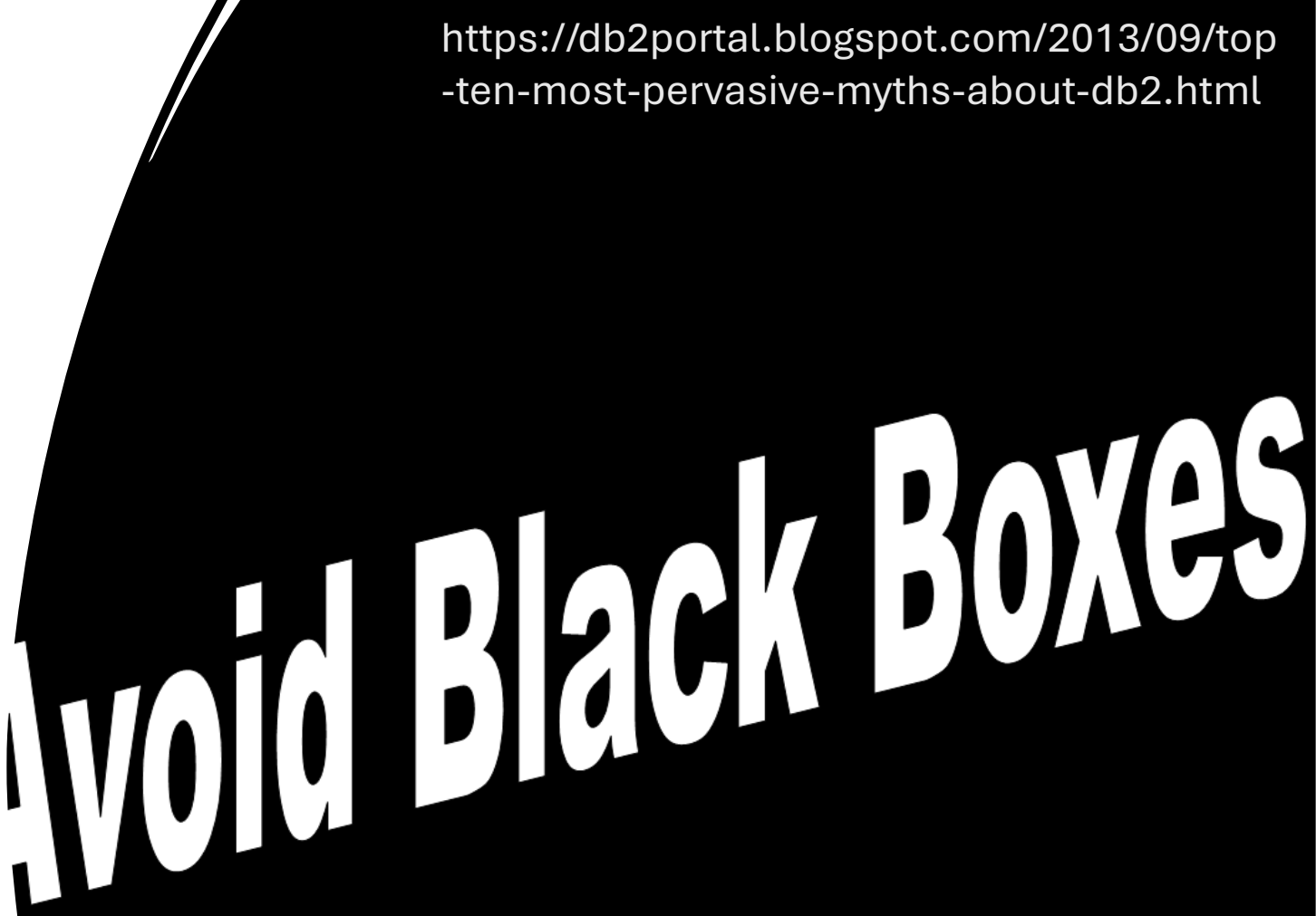
https://www.dbta.com/Columns/DBA-Corner/The-Danger-of-Dirty-Reads-98511.aspx



# Top Ten Db2 Myths

https://db2portal.blogspot.com/2013/09/top-ten-most-pervasive-myths-about-db2.html

1. Use Views to Insulate Programs from Change
2. Locking Problems Are a Database Issue
3. Primary Key is Usually a Good Choice for Clustering
4. Just using the defaults should work out well
5. Programmers don’t need to know SQL tuning
6. Black Boxes work well
7. Using NULLs saves space
8. RUNSTATS Aren’t That Important
9. Db2 is a resource hog
10. It Depends!




# Top Ten Outdated Standards

1. Almost any standard using the words “always” or “never”
2. Limiting indexes per table to 3… or 5… or…
3. Base table views
4. Non-native SQL procedures
5. Limiting number of tables per join (typically for online transactions)
6. Avoiding NULLs
7. Arcane table naming standards (e.g. TXR0031)
8. Just about any buffer pool standard (e.g. BP0 only)
9. Avoiding row-level locking
10. Putting standards in a binder instead of online

# Top Ten Steps to Being a Successful DBA

1. Write Down Everything
2. Keep Everything
3. Automate
4. Share Your Knowledge
5. Focus Your Efforts
6. Don’t Panic!
7. Measure Twice, Cut Once
8. Understand the Business, not Just the Tech
9. Don’t be a Hermit
10. Use All of the Resources at Your Disposal

https://www.dbta.com/Editorial/Think-About-It/The-Guide-to-Being-a-Successful-DBA-78830.aspx

# Top Ten Things to Do Before You Visit the DBA

1. Try to Figure It Out Yourself, Please
2. RTFM
3. Figure out what you are going to say to him/her
4. Be sure it is the truth!
5. Don’t assume you (or your code) are innocent
6. Have a drink (coffee?)
7. Bring the DBA a drink (your choice!)
8. Never say “But IBM said it should work this way.”
9. Never say “But it worked that way yesterday.”
10. Always say “Thank you.”

# Top Ten Programmer Excuses

1. “There’s something wrong with Db2!”
2. “But I copied that from another program.”
3. “It worked yesterday.”
4. “Nothing has changed!”
5. “But I can do that better in COBOL; Java; etc.”
6. “It works that way in Oracle; MySQL; etc.”
7. “It’s too late in the project to re-write that.”
8. “But I heard somewhere it works this way.”
9. “Why do I have to BIND every time?”
10. “Isn’t there something you can do to make it work?”

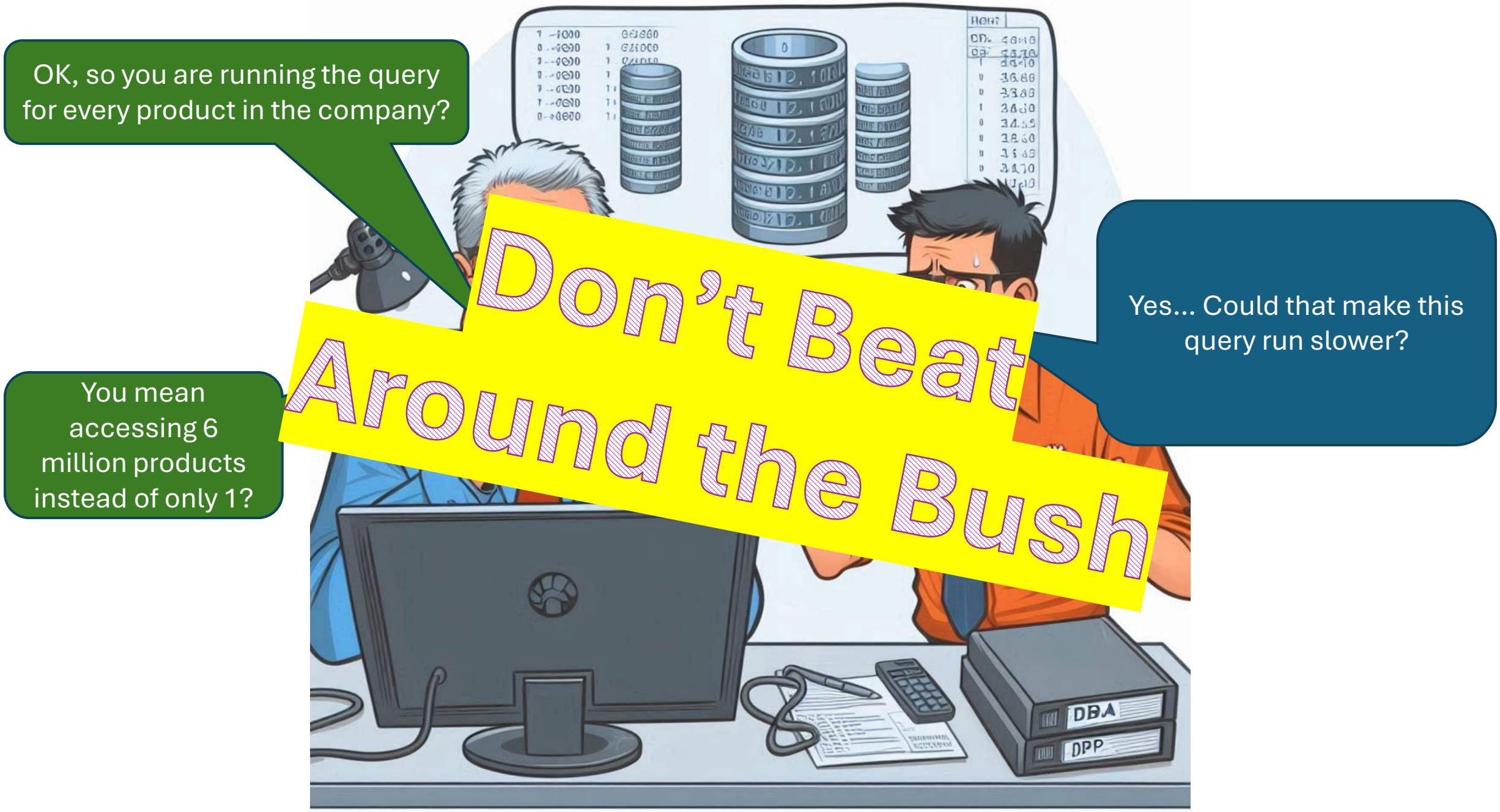
# Top Ten DBA Excuses


1. It depends.
2. RTFM
3. “Did you fill out the form?”
4. “I’m busy.”
5. “It’s working as designed – leave me alone.”
6. IBM says…
7. You couldn’t possibly understand why…
8. “No!”
9. “Our standards say we do it this way.”
10. “Because I’m the DBA, that’s why!”

# Top Ten Things to Never Say to Your DBAs

1. “Hello?”
2. “There’s something wrong with Db2!”
3. “Is there locking going on in the system?”
4. “Do you know where the backups are stored?”
5. “Are you busy?”
6. “This worked yesterday…”
7. Nothing
8. “My program keeps getting -803, what should I do?”
9. “Can you refresh my test data with production data?”
10. Beating around the bush


Grrrr.. What did I do? *Calm down…*
This query has run quickly every day since we created it and it's been around for a couple years now. But when I run it today it runs for a long time and then times out. What did you do?
Well, did you change anything about the query at all recently?
I don't think so…
What about the data? Is there more of it?
Hmmm… Could be, wait-a-minute… we commented out one of the HERE clauses, but the query is v the same other than that.
Which WHERE clause?
The one for products…



# Top ~~Ten~~ Eleven Rules of the Road for DBAs

From zJournal article available at:
http://www.craigsmullins.com/zjdp_042.htm

1. Write Down Everything
2. Keep Everything
3. Automate
4. Share Your Knowledge
5. Focus Your Efforts
6. Don’t Panic!
7. Measure Twice, Cut Once
8. Understand the Business
9. Don’t Be a Hermit
10. Use All of the Resources at Your Disposal
11. Keep Up-to-Date

# The Ten Database Management Commandments

1. Thou shalt always design databases from a logical data model.
2. Thou shalt always document thy database design.
3. *(There are many important aspects of database administration, but of them all,)* data integrity shalt always remain the most important.
4. Thou shalt encrypt personal and sensitive information, both at rest and in transit.
5. Thou shalt implement appropriate security within thy database and between thy DBMS and thy operating systems.

# The Ten Database Management Commandments

6. Thou shalt always maintain the recoverability of thy databases with sufficient backups to meet the availability requirements of the business supported by the data.
7. Thou shalt document a consistent workflow process for database change that assures data integrity while minimizing downtime.
8. Thou shalt always consider transaction performance and data availability in thy database design.
9. Thou shalt work with application developers to ensure efficient code is written to access thy databases.
10. Thou shalt implement appropriate security and controls to adhere to pertinent regulations.

# Top Ten Database Trends

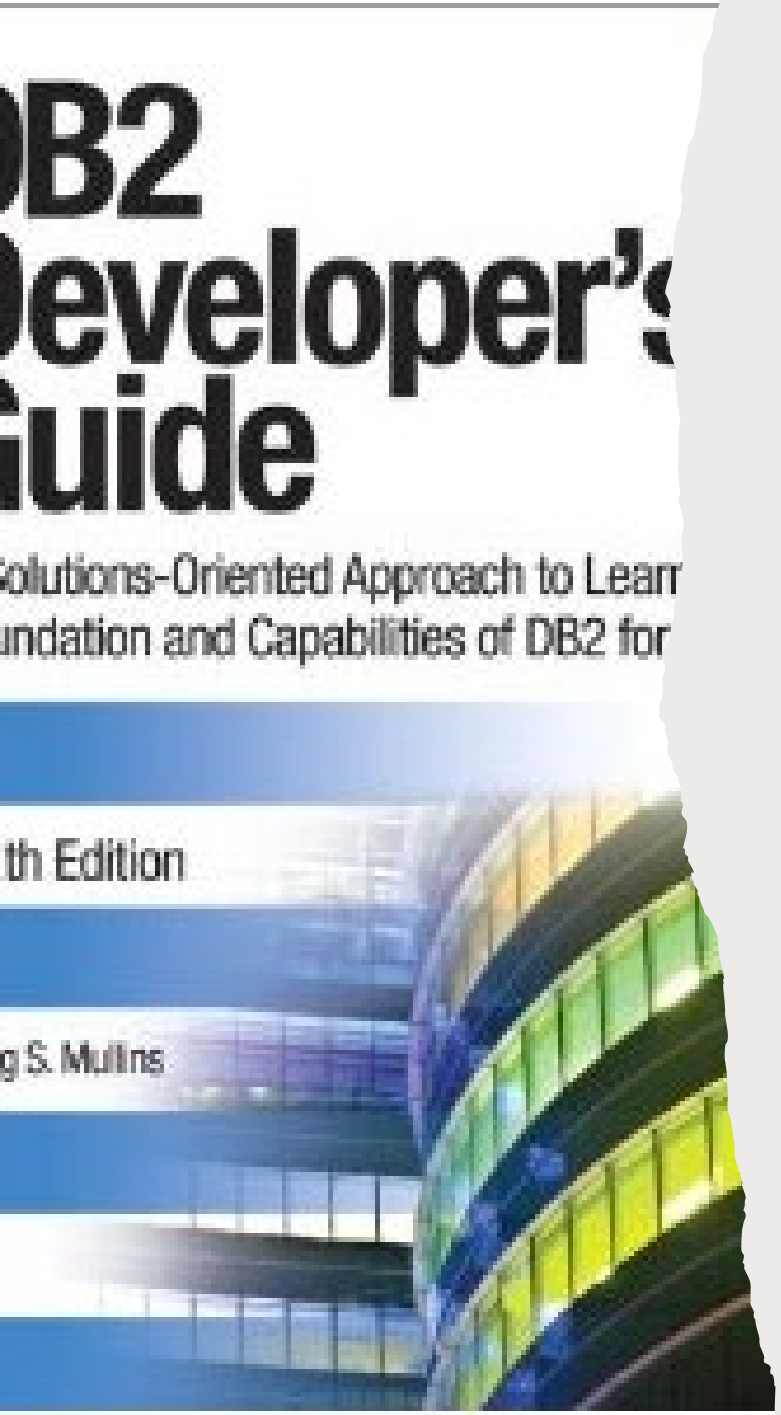
1. AI
2. Cloud
3. Cloud Repatriation
4. From Relational to NoSQL and beyond
5. Running Many DBMSes
   - Polyglot Persistence
6. Big Data and Analytics
7. Data Breaches and Security
8. In-memory
9. DevOps
10. The Role of the DBA is Changing

# Top Ten Trends Impacting Db2 Change Management

1. DEV ops
2. Treating Database Objects like Application Code!
3. Agile
4. Deployment Frequency
5. Compliance Pressure
6. Big Data
7. Larger Partitions
8. Unstructured Data
9. Online Schema Change
10. Tooling

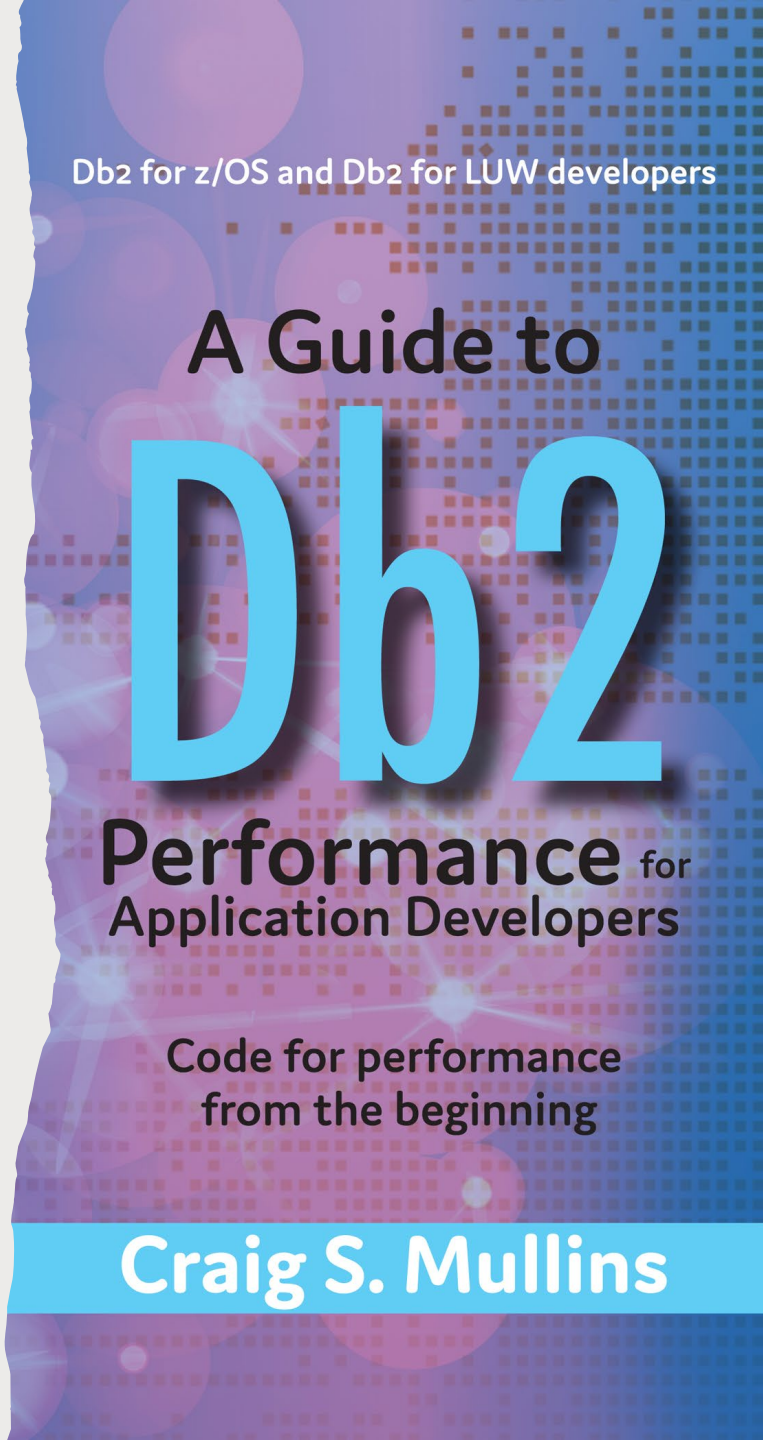
# Top Ten Types of Db2 Tools

1. Change Manager
2. Catalog Visibility
3. SQL Performance Monitor
4. SQL Access Path Analysis
5. System Performance Monitor
6. DBA Automation / Utility Automation
7. Table Editor
8. Log Analysis
9. Database Structure Analysis
10. Application Restart Control




# Top Ten Books for Db2 Professionals

1. Db2 Developer's Guide
2. A Guide to Db2 Performance
3. Db2 Developer's Guide
4. A Guide to Db2 Performance
5. Db2 Developer's Guide
6. A Guide to Db2 Performance
7. Db2 Developer's Guide
8. A Guide to Db2 Performance
9. Db2 Developer's Guide
10. A Guide to Db2 Performance

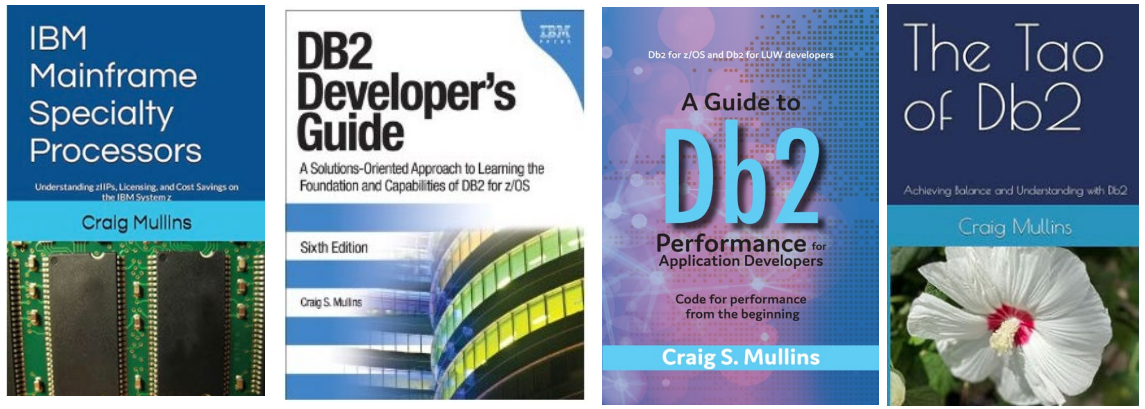


**And if you read those books your office won't have to look like this!**

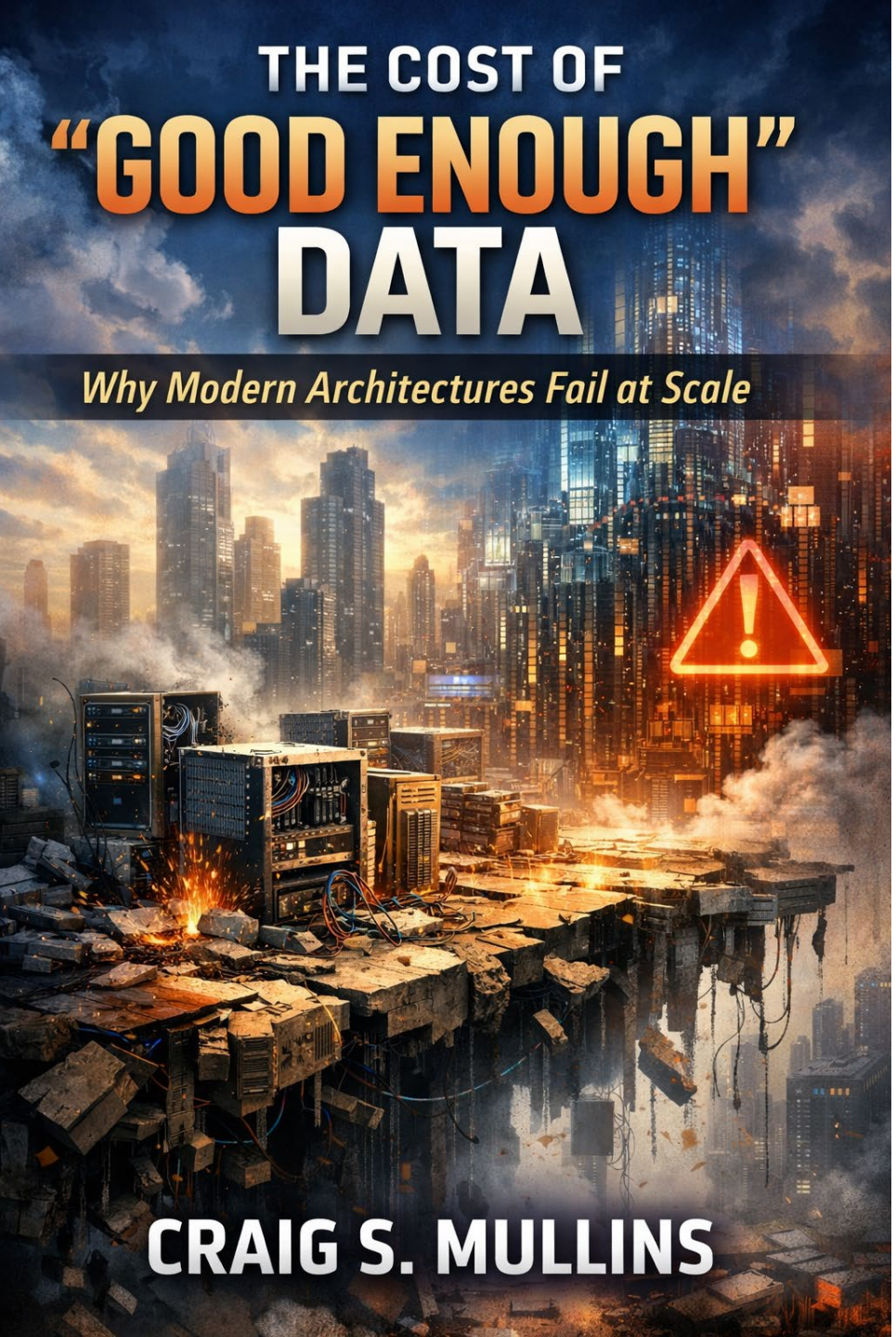
My All-Time
# Db2 Top Ten Lists

- Craig S. Mullins
- President & Principal Consultant
- IBM Champion for Data & AI
- IBM Gold Consultant
- mullc@craigsmullins.com

www.mullinsconsulting.com




https://www.mullinsconsulting.com/books.htm




# I have a new book just published in July 2026!

When did "good enough" take over? This book discusses the impact of accepting "good enough" data and system design driven by the desire to implement quickly and fix things later instead of taking the time to do it right upfront.

Is not an argument against modern data platforms, cloud computing, or new architectural approaches. It is an argument against complacency.

The book examines how and why "good enough" became acceptable. And perhaps more importantly, what "good enough" is costing us.

And how to move forward merging new techniques with solid, time-tested design principles.

SHARE26 August 16-20, 2026 PITTSBURGH, PA

# My All Time Db2 Top Ten Lists

Session ID: TECH209s1

Craig S. Mullins
Mullins Consulting, Inc.
www.mullinsconsulting.com