



Successful Session Abstracts

A clear and snappy title paired with a concise abstract focusing on what attendees will learn from your session make for a dynamic submission that is guaranteed to catch the Program Committee - and attendees' - attention. The title should be no more than 75 characters (shorter if possible), and your abstract is limited to a generous 150 words. Keep your abstract short, but informative, targeting how your session will improve attendees' day-to-day work lives. Clever is good, but don't go overboard - when in doubt, simple and clear is best.

Here are some examples of successful titles and abstracts from previous events:

- Top Titles - A Few Examples
- Zero to Cube - Fast Track to SSAS Development
- New Year, New Dirty Dozen PowerShell Scripts
- Rewrite Your T-SQL for Great Good!
- Where Should I Be Encrypting My Data?
- Add It Up: Analysis Services Aggregations
- ETL Smackdown: PowerShell vs SSIS
- Preventing the "Oh, Poop!" Reporting Situation
- All the Magic Knobs
- Important Trace Flags Every DBA Should Know
- Branding Yourself for a Dream Job - The Modern Resume
- Everything Your Developer Won't Tell You About SQL Azure
- No More Bad Dates: Best Practices for Working with Dates and Times.

Successful Abstracts: Examples

Data Mining: It's Not the Size of Your Data - It's What You Do with It

Adam Jorgensen & Devin Knight

This all-demo session will focus on data mining from non-traditional sources. Many of you can mine from a nice clean dataset, but we will use the new Tabular model to consolidate data from disparate sources and perform some basic cleansing and mining that even an end user can do once they know how. This session will be total audience participation - not a slide lecture and not one silly demo!

TempDB: Performance and Manageability

Robert Davis

Misconceptions and inconsistent advice on tempDB abounds - even highly skilled DBAs cannot always agree on how to best configure tempDB. Part of the problem is that there is no single solution that can be applied to every situation. How many files should tempDB have? Should it be on RAID 1? RAID 10? How about Solid State Drives (SSDs)? This session will give you the tools to determine the optimal configuration for tempDB. We'll also cover how to detect, prevent, and monitor for common tempDB performance problems.

Data Warehouse Mistakes You Can't Afford to Make

David Stein

Many data professionals understand the basics of data warehouse design, including Dimension and Fact Tables, slowly changing Dimensions, and the use of meaningless surrogate keys. However, it isn't until you've created a dimensional model and put it into production, that you realize just how much of an impact seemingly trivial mistakes can make. They can hobble performance, allow inaccuracy, and perhaps worst of all, inhibit adoption and usage of the new system. Learn how to avoid many common mistakes, from someone who's made them and then found ways to correct them.

Edge Case Testing for the Database

Vicky Harp

Are you confident that your application performs as expected on Leap Day? Do you know how long it would take to run out of identity values in your major tables? Can you support case-sensitive collations and non-standard sort orders? If not, odds are good that you are not edge and corner testing your application. In this session, learn how to define the edges of your application and start to stretch those boundaries by setting up challenging test and development environments. For database administrators, learn what configurations in your environment may be a source of bugs in application code and what you can do about it. Learn about edge cases that you can start hardening your code against on Monday morning!

Unraveling Tangled Code – A Spellbinding Tale of Victory Over Chaos

Jennifer McCown

Once upon a time, you inherited an application or a database that was filled with chaos and inconsistencies. The T-SQL code is overly complex and impossible to ken. The architecture is painful to behold, and grueling to code for. One might optimistically say that query performance is “spectacularly mediocre.” If you're without good documentation or system architects to guide you, how do you break the curse of confusion? In this session you'll learn several methods for conquering chaotic code, and how to seek and destroy some of the nastier coding mistakes and inefficiencies. We will break complicated queries into key pieces, turn them upside-down, and reform them into something sensible. We will vanquish major issues like data abuse and index negligence. We will restructure joins, tame subqueries, and refactor dynamic SQL. We will emerge victorious!