Intro

Bit of wordplay with assistant (or self ☺) Suggestions: Back of my hand, Age, Book hocking

Today, rather than a set of boring slides (yeah, my slides are boring), I present you the WHEEL OF WHAT COUNTS. Each spot will lead us down a path that only the wheel knows;

Our plan is to discuss what makes a GREAT DBA or really , data or any sort of programmer in many cases.

1. These are but a few aspects of what makes a great DB. On my wheel this <time of day> are:
2. 6.
3. 7.
4. 8.
5. 9.

5.

1. And there are 3 **“special”** spots on the wheel for you
   * First, a **Draw 1** spaces – From which you will draw from my collection of unwritten ideas and (ideally) will as a group write them for me. There are also several: suggest idea items to draw from where you can give me ideas you want to see. As a reward I will give the spinner a draw for one of these books **<Introduce books>**
   * Second, 1 **Joker** spot. Get that, you get a book and you get to choose the topic.
2. Once the books are gone the next person who hits a special space will result in a draw from the unwritten pile, with regrettably no free anything…Sorry, the wheel giveth, and the wheel can be a jerk too.

So let’s get started with our first spin…

PASSION (LOVE OF RELATIONAL DATABASES)

Gutter minds

My favorite interview question: WHY DO YOU WANT TO BE A DBA?

cheesy: I want to grace you with my greatness

Too realistic: I have kids with a food hole that keeps needing to be filled.

Best! I LOVE DATABASES

Passion versus Enthusiasm : True passion lasts forever, but enthusiasm dies pretty quick. This is my most important characteristic when looking for a coworker. Someone who really wants to learn and will spend their own free time trying to improve.

But you can go too far. I stand in front of you a cautionary tale. Don't spend your entire life parked in front of a keyboard

SKILL (something you earn)

Vince Lombardi: Perfect practice makes perfect

Is it really so hard to understand normalization? Set based theory? Query optimization? Indexing and performance tuning? Biggest barrier often is previous knowledge, programming skills cultivated before you get started with SQL

Getting skill isn't that hard. You can find training at so many levels from free (like this very event which you have chosen to see my cheesy presentation!) to books (shamelessly display the books)

The point is that you want to be skilled and do things right, not just what **seems** right

HUMILITY (Knowing what you know, but not rubbing one’s nose in it)

Consider the offensive lineman on a football team.

The enemies are many:

* Developers who don't want to deal with > 1 table in a database
* Managers who will easily survive the Zombie apocalypse
* Packaged applications that look good on the outside but were clearly built by a 3 year old
* Hardware that is substandard and regardless fails

Not that all DBA'S are humble, or all devs are evil (I said not!) ..Managers well, I...and not all packaged apps are horrible and hardware doesn't always fail..

In the end the job of a DBA is to be the Janitor of the systems. If systems end up with unclean data, the DBA is generally saddled with the task of fixing it. Humility and sincerity are the keys to surviving

ADVENTUROUSNESS (Learning other technologies!)

NOT! PROBLEM SOLVING

This was my April Fools column last year.

Who in here has a cell phone that ONLY makes phone calls? (if anyone does, give them Target gift card?) DOESN’T send texts? Doesn’t have a web browser? Ad Lib here

This topic is a twofer.

1. If you have never met a DBA that is a luddite of sorts (ancient cell phone, no laptop, etc) you don't get out much

1. This should not be the case (Remember COBOL, it was once the big thing…it isn’t anymore…)

There is so much to learn out there that we have to **keep up.**

As much as I hate programmers (and I do) I don't understand what they do. Some of what they need may be right.

 Plus there are many new database technologies coming, even from Microsoft: Hadoop anyone? Other types of NoSQL? EAV types? Amazon? If you don't know the competition, you will lose. For a good taste, check out Dr Dewitte’s keynote from PASS last year. Awesome stuff.

FAILURE (The best teacher)

 Programming is like playing video games. Failure, failure, failure, …., success.

What are you faced with as a DBA?

Data errors, hardware issues, networking issues, etc.

How many times do you succeed?

**Once**

Failure, especially public failure, is something that by nature we shun. However, the try-fail-learn cycle is one that all good DBAs (and programmers) must accept and embrace without fear

CURIOSITY (about solving the problems around you)

 NOTE: NOT ADVENTUROUS WITH TECHNOLOGY

Did curiosity kill the cat? Well, possibly, but not my cat.

**<picture of cat>**

The fact is curiosity only seem to kill things on TV. Like if you were riding with James Bond on a leisurely trip to Target for a new set of Legos.

Curiosity Not dead: **What is this button?**

Action: Pushes button, ejected at 45 MPH

Curiosity makes you great. Try to figure out problems. Dig deep and find root causes. No one likes the person who asks every day the same questions about how to do the same sorts of things. The only thing worse? A person who pushes the button without knowing exactly what it will do. And that is called stupidity.

LOGIC (Or what managers rarely show)

"There are <hold up picture of 10> kinds of people in the world. Those who will always wonder why there are only two items in my list and those who will figured it out the first time they saw this very old joke."

Logic is necessary in all parts of your programming life/career, mostly for problem solving. **Different from Curiosity as you can be curious about banana farming and never try to solve banana farmer’s problems!**

A very annoying trait is to always jump to the same conclusions regardless of the problem. (Like how it is always the databases' fault)

Use logic to start at the obvious causes of life's litte problems and find the issues.

Yes, if you didn't see it, 10 is binary for 2.

BLINDNESS (Ignoring race, gender, but not stupidity)

"Ever since I was a newbie, I wrote code like a storm

My databases rendered in the fifth normal form

But I ain't seen anything like him,

On any IT team, that deft DBA can,

Code T-Sequel up a storm"

Ok, so I am not talking about artificially making yourself deaf dumb and blind to cut out the real world to code better. More talking about being blind to personalities, skin color, which bathroom they go into, etc.

I started because of the Women In Technology panels. I hate hearing people label themselves as "women", "men", "minority","physically challenged", etc, unless it is truly necessary.

There are 2 kinds of people in the word: DBAS, not quite/potential DBAs, how things work in the restroom is not important to me

Equal pay should be for equal work, regardless of anything physical characteristics.

Observant (y’know, watching for signs of trouble…)

We nerdy types (or maybe it was just me?) generally learned on the school playground to keep an eye open for unprovoked attack from simpler, but more muscular souls, and take steps to avoid messy confrontations well in advance.

Those skills of observance now come in a lot more handy worrying about Gremlins (not the car), Hackers, Managers (the bad ones, only, of course), and other evil beings bullying our applications into giving up their lunch money and or person information of it’s users

Imagination (Using your brain for more than just hoping Friday is here)

One little spark, of inspiration, is at the heart, of all creation…

As DBAs, our jobs can seem mind numbing, and not important. Our jobs can be excruciatingly boring in nature. Every database I have created has been more or less a big old spreadsheet. But by realizing that spreadsheet is actually a very good spreadsheet, and that is a part of a much larger picture… it keeps me sane…

The second sort of imagination that is useful is when solving problems. A lot of dbas get stuck in the past and never try new stuff. In order to solve the crazy problems that you will come across, some imagination can help to come up with “out of the box” solutions..

Time (is NOT on our side)

You have to stay ahead of the curve.

The big piece of the equation is balance. It is important to:

1. Keep ahead of the curve, learn the new features of SQL Server

2. Keeps yourself healthy

3. Keep your personal life straight

4. Earn a living

Use time wisely is a topic we start out with when we are in Kindergarten, and it will always be with you.

Luckily, we can squeeze in more and more out of the day with more and more ability to carry technology with us. Books don’t weigh 100 pounds now, they weigh the same amount no matter how many books you have (hold up mobile device).

[](http://www.amazon.com/gp/product/images/B0041P3ME6/ref=dp_image_z_0?ie=UTF8&s=miscellaneous)

Imagination

Logic

Hat 1 - Book

Blindness

Observant

Failure

Curiosity

Joker - Book

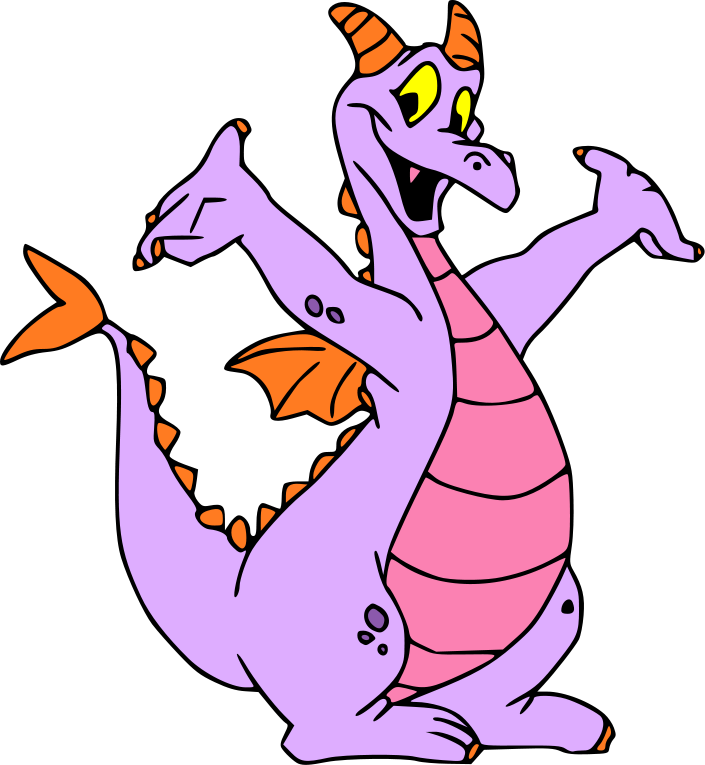
Passion

Skill

Humility

Time

10





|  |  |  |
| --- | --- | --- |
| Memory | Experience | Calm |
| Tenacity | Motivation | Thickness |
| Laziness | Ego | Honest |
| Detail | You Tell Me | You Tell Me |
| Vigilence | You Tell Me | Academics |