



Official Publication of the Northern California Oracle Users Group

# NoCOUG

JOURNAL

Vol. 33, No. 3 · AUGUST 2019

## Women in Technology

### Interview

*Kellyn Pot'Vin-Gorman calls it like she sees it.  
See page 4.*

### Interview

*Gwen Shapira is no less blunt.  
See page 8.*

### Crushing the IT Gender Bias

*Opinions supported by the data.  
See page 11.*

*Much more inside . . .*

# Amazon Aurora

Performance and availability of commercial-grade databases at 1/10th the cost.



Learn more: [aws.amazon.com/aurora](https://aws.amazon.com/aurora)



# Professionals at Work

First there are the IT professionals who write for the *Journal*. A very special mention goes to Brian Hitchcock, who has written dozens of book reviews over a 12-year period.

Next, the *Journal* is professionally copyedited and proofread by veteran copy-editor Karen Mead of Creative Solutions. Karen polishes phrasing and calls out misused words (such as “reminiscences” instead of “reminisces”). She dots every i, crosses every t, checks every quote, and verifies every URL.

Then, the *Journal* is expertly designed by graphics duo Kenneth Lockerbie and Richard Repas of San Francisco-based Giraffex.

And, finally, the *Journal* is printed and shipped to us. This is the 133<sup>rd</sup> issue of the *NoCOUG Journal*. Enjoy! ▲

## Table of Contents

Interview .....	4	Photo Montage.....	26
Interview .....	8	<b>ADVERTISERS</b>	
Book Excerpt.....	11	Amazon .....	2
Brian’s Notes .....	18	Quest .....	10
From the Archive.....	22	Axxana .....	17
Special Feature.....	24	MemSQL.....	28

### Publication Notices and Submission Format

The *NoCOUG Journal* is published four times a year by the Northern California Oracle Users Group (NoCOUG) approximately two weeks prior to the quarterly educational conferences.

Please send your questions, feedback, and submissions to the *NoCOUG Journal* editor at [journal@nocoug.org](mailto:journal@nocoug.org).

The submission deadline for each issue is eight weeks prior to the quarterly conference. Article submissions should be made in Microsoft Word format via email.

Copyright © by the Northern California Oracle Users Group except where otherwise indicated.

*NoCOUG does not warrant the NoCOUG Journal to be error-free.*

## 2019 NoCOUG Board

Andy Sutan  
*Member at Large*

Babu Srinivasan  
*Member at Large*

Dan Grant  
*Exhibitor Coordinator*

Dan Morgan  
*Conference Chair*

Eric Hutchinson  
*Webmaster*

Iggy Fernandez  
*President, Journal Editor*

Kamran Rassouli  
*Social Director*

Linda Yang  
*Member at Large*

Liqun Sun  
*Membership Director*

Manoj Bansal  
*Member at Large*

Mingyi Wei  
*Catering Coordinator*

Naren Nagtode  
*Secretary, Treasurer, President Emeritus*

Sherry Chen  
*Catering Coordinator*

Tu Le  
*Speaker Coordinator*

Vadim Barilko  
*Webmaster*

### Volunteers

Brian Hitchcock  
*Book Reviewer*

Saibabu Devabhaktuni  
*Board Advisor*

Tim Gorman  
*Board Advisor*

## ADVERTISING RATES

The *NoCOUG Journal* is published quarterly.

Size	Per Issue	Per Year
Quarter Page	\$125	\$400
Half Page	\$250	\$800
Full Page	\$500	\$1,600
Inside Cover	\$750	\$2,400

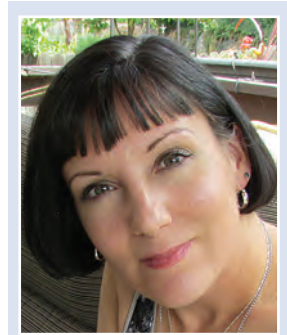
Personnel recruitment ads are not accepted.

[journal@nocoug.org](mailto:journal@nocoug.org)



# Women in Technology

with Kellyn Pot'Vin-Gorman



Kellyn Pot'Vin-Gorman

*Editor's Note: The NoCOUG Journal interviewed Kellyn Pot'Vin-Gorman in 2014. This is an updated version of that interview.*

***It's been five years since your last Women in Tech interview with NoCOUG. What is your view of the number of women in technology now vs. the 2014 interview?***

It's surprising how much has changed for me and yet how much has remained the same for most women in tech. As part of authoring my recent book, *Crushing the IT Gender Bias*, I had to document and provide research data on every statement and in every chapter where it wasn't clearly stated that it was a personal opinion or experience. The numbers for women in technology five years later are very much the same as they were in 2014—right around 23%. I was also surprised when rereading the original NoCOUG interview to realize that the numbers for people of color in technology haven't increased either.

It may surprise your readers that with so many companies having adopted measures that address diversity and inclusion, so little has changed. The real problem is that we are rarely dealing with the real problems. A company is handed a set of checkboxes to mark off, but they are still unable to address unconscious bias. At the same time, we are experiencing a renaissance in backward thinking, with some male leaders justifying removing women from their professional circles because they feel it's "asking for trouble/rumors" or "disrespectful to their wives." These are just excuses to make bias acceptable in some circles and leave women less likely to succeed while their male peers excel. For someone who lives by data and analysis, this is bizarre to observe—but I have also learned that strategy and persistence are the best way to deal with this lack of logic when it comes to gender bias.

We're also failing on the education front. We still don't understand the role that technology needs to play in our children's education goals, and this should begin in elementary school. As of 2018, 37 states still don't require technical education as part of high school graduation credit requirements ([www.ecs.org/](http://www.ecs.org/)

[wp-content/uploads/Career-and-Technical-Education.pdf](#)). There is significant movement to add access to vocational education for technology, but there's little funding or change of focus, even though school boards and higher education are aware that by next year 65% of jobs will require some technical proficiency. To adjust our education standards to this new workforce demand, schools need to start approaching technology as they do science, math, English and history. It's not about making every child into a technologist; it's about exposing them to technology and ensuring that they understand basic technology skills (more than just as a consumer), and giving them the opportunity to embrace it, just as they now do science, history, and math. To do this, it needs to start early, building critical thinking skills by ages eight or nine.

What has changed for me has much to do with the technical community that I'm part of. I loved the Oracle community, but in the last five years, I branched out to do more in development and DevOps, and now I am deeply ingrained in the Microsoft community. This community recognized the deficit of women's involvement early on and started to ask the tough questions of how to address it. It's a more relaxed community; they are younger than some of the communities I've been involved with in the past, allowing them to be more progressive with the solutions, and the small changes they made really did make a difference. Nothing is ever perfect, but women on average are between 30% and 40% of presenters and make up a larger amount of attendees at the events I present at. The women are more likely to write and evangelize, and the men are more likely to support the initiative as allies in the Microsoft community. Due to this, it was a simpler transition, and I felt very welcomed as a technologist. That's often the only thing women in technology want: to be given an equal chance to contribute. The Microsoft community should be given credit for the hard work they've done to try to change the statistics.

***“Gender bias is often an unconscious act. If a manager goes out of their way to prevent an employee from receiving a raise or promotion, that is discrimination. If that same manager doesn't grant the raise or promotion because—when considering the list of the employees to receive raises and/or promotions—they don't think of women in leadership roles or they don't consider women in need of the same pay as a male head of the household, it could be gender bias.”***

***“When I first began in the field, I was instructed to specialize. With the cloud, autonomous database, and machine learning, the day of the specialist is over. The more technical knowledge you gain about the entire stack, the better you are at your own role and the more of an asset you will be to the business. Don’t become technically isolated to a product. The more you know about a competitor’s product, the better you can understand your strengths and weaknesses, and how to improve.”***

Along with being part of the Microsoft community for the last three years, I’ve also been employed by Microsoft for the last year. It doesn’t hurt to have an incredible visionary as the company CEO, as Microsoft has with Satya Nadella, but management was also willing to be part of the significant cultural shift. I’ve never felt so welcome or had so many opportunities to be part of a successful company as I currently do. I have a great team and a great manager that allow me to be the best I can be for myself and the company. This goes a long way toward increasing satisfaction in a tech career, along with being a woman in technology.

***What should companies like Microsoft, Google, Facebook, Twitter, and Yahoo do (or not do) to bring more women into technology? Would you call for affirmative action and outreach, or would that lead to reverse discrimination?***

With education in mind, it is important to discuss the difference between discrimination and gender bias. If a woman doesn’t receive a raise or a promotion she deserves, we might be quick to level accusations of discrimination (which a company policy covers), but the difference between the two is intention. Gender bias is often an unconscious act. If a manager goes out of their way to prevent an employee from receiving a raise or promotion, that is discrimination. If that same manager doesn’t grant the raise or promotion because—when considering the list of the employees to receive raises and/or promotions—they don’t think of women in leadership roles or they don’t consider women in need of the same pay as a male head of the household, it could be gender bias. This same manager, if approached about the issue, might be surprised by the oversight, not even realizing they were grading the female employee differently than her male peers. Without discussion or education, they might not even realize anything was wrong with their decision. The biggest disappointment for me in the last five years is to discover that unless people are held accountable for their bias-driven actions, the needle never moves—and much of society is willing to justify it.

In 2014 I stated that discrimination wasn’t much of a problem, as this is easier to identify—and due to this, companies have worked to build policies to address it. I also have seen the positive difference a full diversity-and-inclusion initiative can bring. Microsoft’s isn’t just a policy that creates a few programs that offer them a clear conscience that they’ll be able to check a few boxes off on the requirements. In my opinion, the leadership actually works to ensure their policies are part of every possible initiative within the company. When difficult situations occur, which is part of human culture, they work to have open communication and diligently address the problem as best they can. Culture is one of the most difficult hurdles to overcome, but through persistence, patience, and education, real change can happen.

***How can we continue learning even after decades spent in technology?***

Technology changes fast and we must change with it. When I first began in the field, I was instructed to specialize. Due to my natural lack of focus, I just never did. I’d always find something new and interesting to learn about and never went as deep, but deep knowledge came naturally with time and experience.

With the cloud, autonomous database, and machine learning, the day of the specialist is over. At the same time, I’m always amazed how often the skills from my past, like shell scripting, automation, and communication, along with my critical thinking skills, are a coveted asset to the technology of tomorrow.

- The more technical knowledge you gain about the entire stack, the better you are at your own role and the more of an asset you will be to the business.
- Look around the role you’re in to find out how to make yourself more valuable. What can you automate, and where can you remove bottlenecks?
- Don’t become technically isolated to a product. The more you know about a competitor’s product, the better you can understand your strengths and weaknesses, and how to improve. Within every database platform I’ve worked with I’ve found something to admire and wish were in another database product.
- Create long-term goals and work to accomplish them. Don’t worry about what peers are doing; just keep your eye on your own path and goals. Persistence pays off.
- As part of those goals, don’t just find mentors, find sponsors. A mentor can guide you, but a sponsor can provide you with access to opportunities. Women are four times more likely to get mentors but three times less likely to get sponsors. Sponsors open doors to promotions, new jobs, networking, and career goals.

***How has your DBA career evolved in the recent past and what were the lessons learned?***

I will always refer to myself as a database administrator, although the last few years I’ve been a DevOps engineer and now I’m in analytics and AI at Microsoft. I love new technology—every chance I’ve had to learn something new, even if I didn’t have any experience in the area going in, I’ve given myself the space to learn (and fail), and I’ve succeeded in the end.

One of the most interesting things I’ve learned is that what you’re told you will no longer need to know often becomes mission critical at some point in the future. The team I’m on now had huge deployments that took weeks with customers to work

***“Create long-term goals and work to accomplish them. Don’t worry about what peers are doing; just keep your eye on your own path and goals. Persistence pays off. As part of those goals, don’t just find mentors, find sponsors. A mentor can guide you, but a sponsor can provide you with access to opportunities. Women are four times more likely to get mentors but three times less likely to get sponsors. Sponsors open doors to promotions, new jobs, networking, and career goals.”***

through; I was able to use shell scripting skills from my previous Oracle DBA role and automation skills from my previous DevOps role to bring it down to a ten-minute task. Never underestimate the value of your skills just because they don’t match what your current peers possess.

***NoCOUG has been around for 28 years. The upcoming conference will be our 112th. That’s an amazing story, but it’s a tale of declining numbers. Is NoCOUG a dinosaur in the internet age? We offer simply amazing quality, but what real reason is there for anybody to skip a day of work to listen to presentations when so many presentations (and entire recordings) are available on the internet? It’s like an episode of Restaurant Impossible. What items should we add to our menu to keep the diners coming back, and how can we reach the thousands of Oracle professionals in our area who have never heard about NoCOUG. We’re envious that the RMOUG conference is so big. What do you have on the menu that we don’t? Would you advise us to switch from our current model of four multiple-track full-day conferences per year to a single conference each year? Additionally, the current board members have been serving for way too long and are close to complete exhaustion because we have no staff—not that we could afford it anyway. We really need new ideas, Chef Kellyn.***

Although I’m no longer with RMOUG, I think there is something to be said for letting go. My husband, Tim Gorman, and I had to walk away at a certain point and not look back. That said, we could walk away. We worked hard to find others that would take over as we walked away, and we had another “heart” of the organization. No user group will survive without at least one person as its heart.

I’m now the president of the Denver SQL Server User Group, and the Microsoft regional communities are all connected through one group called PASS (the acronym is all it’s identified by now, but it used to be Professional Association for SQL Server). We are all offered a central portal, a website, an annual “conference in a box” kit, a website for our annual conference, etc. The benefit of this is that we aren’t all siloed and have a tendency to work more closely together. We all have the same conference and can work with each other to improve upon the basic package and share ideas. When something works well for one, it’s socialized to the other groups, too. The support with the website, conference signage, stats page, management package, etc., are offered at a very low price. The community is much closer knit because of this, too. The events are free and the only money is through sponsors and exhibitors.

We’ve just entered into the busy time to plan for our annual SQL Saturday Denver, and it’s a lot of work. There are sponsors, speakers, and exhibitors to contact; a venue to lock down; ab-

stracts to review; schedules to create; and a plan to devise to help us attract more local attendees. We also want to make sure that we create the social exposure for the event, as Denver is kind of secluded from the mid-Eastern and Southern area that experience the larger events. There’s a lot to do, but I have a great board to work with and am really impressed with how they accomplish it all.

The SQL Server regional events are free and held on a weekend, so the venues are cheaper (or cheaper ones like universities are sought out) and attendance is higher. The Denver SQL Server user group does have monthly meetings with 40–125 attendees. The events are broadcast on Twitter (where the SQL Server community has an active presence) and through emails, LinkedIn, and Meetup. The sense of community drives many of the board members to be more active on the board. The activity is rewarded and coveted as part of the community.

Learning from each other is the best way to improve your own event. With all that said, I do feel the pain of trying to find new board members. Although I moved away from Denver, I’m the one and only “remote” president of a SQL Server board.

***You’re a social media expert. What’s your advice for the rest of us social media noobs?***

Social media is still a haven for me. My husband and I sold our house after the last kid left for college and we’ve been living in a 42-foot fifth-wheel trailer for just over a year. Over the Fourth of July weekend we celebrated returning to the starting point of our journey after circumnavigating the United States. We’ve been documenting it via a new blog ([danceswithwinnebagos.com](http://danceswithwinnebagos.com)), and it’s been a challenge to keep updating a personal and professional blog along with social media. I also wrote all or parts of three books last year (one as the sole author), along with numerous articles, and spoke at 25 Microsoft weekend events. You have to be dedicated to your brand and I do believe in it. I had shifted my focus from purely Oracle to purely Microsoft in the last year, and this required a careful switch of followers and readers. I now have a strong mix of both due to the authenticity of my content. My technical blog ([dbakevlar.com](http://dbakevlar.com)) has around 3000 viewers per day—which I only know because I have to make sure my blog server can handle that kind of workload!

Recently a new logo was created for me and I really love it. Search engines use images just as much as words for identification, so having an image or a graphic that you can use on all social media platforms, blogs, etc., works well to build a brand. The previous image was a photograph taken in 2007. It was time for an update, and I loved changing from a photograph to a professional logo that so closely identifies with me.

Find out what social media platform your audience uses to communicate. Oracle users are more likely to use LinkedIn and



Microsoft users are fans of Twitter. Region can also be a determining factor. Pete Sherman is in the Asia-Pacific area and discovered that Facebook was the way to connect with his audience due to the region's high rate of Facebook users.

Top recommendations for social media:

- ▶ I can't stress enough the importance of authenticity and being a little quirky.
- ▶ Don't say anything to someone online that you wouldn't say to them in person.
- ▶ If you feel you're too busy but want to build a brand, consider a social media post-scheduler like HootSuite, Buffer, or Zoho. That way you can schedule posts to go out all day, even if you only have limited time.
- ▶ Never worry about followers. I actually have as many followers blocked as I follow. The quality of your followers directly affects the quality of interactions that you will have online.
- ▶ Follow up to the last bullet point. Don't argue online or let others harass you. You don't owe anyone anything online, including interaction. The mute and block button is your friend. Use it without worry. There are some people in this world we aren't meant to socialize with, especially online.
- ▶ Treat your social brand like your resume or your business. It's all right to have fun with it, but treat it with respect.

**The grand master of self-reliance, Ralph Waldo Emerson, said, "A foolish consistency is the hobgoblin of little minds, adored by little statesmen and philosophers and divines. With consistency a great soul has simply nothing to do. He may as well concern himself with his shadow on the wall. Speak what you think now in hard words, and to-morrow speak what to-morrow thinks in hard words again, though it contradict every thing you said to-day.—'Ah, so you shall be sure to be misunderstood.'—Is it so bad, then, to be misunderstood? Pythagoras was misunderstood, and Socrates, and Jesus, and Luther, and Copernicus, and Galileo, and Newton, and every pure and wise spirit that ever took flesh. To be great is to be misunderstood." Have your opinions changed or become more nuanced since 2014?**

I find this question interesting, as I came to an epiphany when I was writing *Crushing the IT Gender Bias*. People can form opinions without any factual evidence or experience. I challenge someone to call it an opinion when there's data that proves it false. I was a very observant child who learned to test out my theories quite young, and I am naturally analytical. I feel things very deeply, but I've never been one to respond to my emotions. I'm more fascinated as to why I felt the way I did and want to get

to the data behind it. This affects how my opinions are formed. When I have an experience, I do research to understand why it happened and then decide how to strategically address the problem. It results in a person with opinions built from a very scientific approach.

Taking all that into consideration, what has happened since 2014 is that my own opinions, which I assumed were based mainly on the experiences I had with individuals and women I mentored—but which lacked significant research in 2014—have now been backed up by so much diversity and inclusion research data that I was able to build the indices for my book in less than two weeks. Yes, my opinions have changed: most are now backed up by facts and by university and scientific research.

Five years ago I would recognize bias but hesitate to call it out. Now, in 2019, I have no qualms about having those difficult discussions. You must become the change you want to see in the world. ▲

---

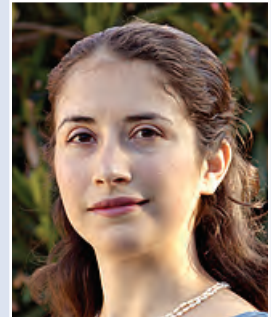
*Kellyn Pot'Vin-Gorman is a talented and accomplished technical evangelist and professional geek, and an Azure Data Platform Architect in Oracle on Azure, Power BI, and AI for EdTech at Microsoft. She is a proud member of Oak Table and an Oracle ACE Director Alumnus and former Idera ACE with over 20 years of extensive experience in database administration, database group management, and cloud migrations. Her specialties are Oracle implementations on Azure "bare metal," architecture/database optimization, automation scripting, DevOps, cloud migration, and heterogeneous database environments management. She is currently working with a larger customer base on multi-database migration to Azure VMs—specifically Oracle but also DB2, SAP HANA, and others. Kellyn has presented at numerous Oracle events, such as Oracle Open World, Collaborate, and KSCOPE. She's recently become better known at Microsoft events like SQL Saturdays and PASS Summit. She presents on Database, DevOps, and Analytics topics, plus maintains a popular technical blog called DBAkevlar at dbakevlar.com. Kellyn is often involved in authoring both technical and non-technical books, having been part of numerous publications around database optimization, DevOps, and the command line. She is deeply involved as a mentor and sponsor in the technical community, including multiple WIT (Women in Technology) groups and initiatives. Her fifth book is the WIT book *Crushing the IT Gender Bias*, documenting two decades of experiences and advice from her career in tech. Kellyn has also developed technical curriculum for Oracle Education Foundation, Devox4Kids, and individual STEM education, introducing single board computing and Python to the next generation through pilot initiatives and volunteer organizations.*

Copyright © 2019, Kellyn Pot'Vin-Gorman

***“Treat your social brand like your resume or your business. It's all right to have fun with it, but treat it with respect. Don't say anything to someone online that you wouldn't say to them in person. The mute and block button is your friend. There are some people in this world we aren't meant to socialize with, especially online. I actually have as many followers blocked as I follow. The quality of your followers directly affects the quality of interactions that you will have online.”***

# Women in Technology

with Gwen Shapira



Gwen Shapira

*Editor's Note: The NoCOUG Journal interviewed Gwen Shapira in 2015. This is an updated version of that interview.*

***How did you get to be a Big Data expert? What were you doing before that? What did you study in school? And, if we may ask, where did you go to school? You don't have a Californian accent.***

It's a very long story with a lot of fortunate coincidences.

I'll start at the end. I got my current job at Confluent by contributing to the open-source Apache Kafka project. I happened to work on Apache Kafka because in mid-2014, the company I worked for at the time (Cloudera) was looking at the possibility of supporting Kafka. At that time, I had just moved from consulting to software engineering and I didn't have a specific project to work on. Apache Kafka looked like a nice project. I mostly wanted to work on it because the community was so inclusive and friendly; at the time I didn't know it would become a technology that would change how companies run their business.

How did I end up at Cloudera? Through Twitter of all things! I was working with a retailer in Japan, as a consultant through

***“Career opportunities happen only when people want to work with you. This means that first, people need to know you—networking was key to every career move I made. Second, you need to bring a lot of value and be accountable for your work. We all want to work with people who, when they say they'll do something, they deliver.”***

Pythian. I was there to help them migrate to Exadata, but their data warehouse system was tightly integrated with Hadoop, and I got to work on this integration too. I got a lot of exposure to Cloudera's distribution—indispensable tools for integrating Hadoop with data warehouses such as Sqoop, Avro, Hive, and Impala. Many of those tools were indispensable but not 100% mature at the time, so I spent a lot of time troubleshooting; since I was alone in Tokyo, I vented on Twitter. I got into interesting and quite useful discussions with a bunch of Clouderans, and eventually Cloudera CEO Mike Olson messaged me and asked to meet when I get back.

We met, we chatted, I talked to his team, and a few months later, I decided to leave Pythian and my beloved Exadatas and go help Cloudera customers make use of new types of data and new data systems. To this day, I'm not sure if I was hired because I'm a good consultant or because they wanted the complaints on Twitter to stop.

My adventures with data started a lot earlier. I got my college degree in computer science and statistics, so you can say I was fascinated by data analysis from pretty early in my career. I started working at Mercury Interactive as a developer and later got promoted to a team lead, but I was not happy leading a team and looked for a way out. When our star Oracle DBA left, I convinced my managers to let me take the DBA role.

The DBA who left was Hanan Hit, of NoCOUG board fame. He continued being a great friend and mentor throughout my career.

A few years later, HP acquired Mercury and I got a chance to relocate to the U.S. HP is where I got started at blogging and later presenting. The job was not very challenging, so I was left with plenty of extra time to learn and to write. As a junior DBA, very few people in HP listened to my ideas—I was surprised to find out that outside of HP people were reading my thoughts, responding, and even inviting me to present!

Presenting at conferences helped me network with a large number of truly brilliant people. One of them is Alex Gorbachev,

<sup>1</sup> Klawe asked, “For women who aren't comfortable with asking for a raise or—sort of saying—who aren't the younger you, let's say, what's your advice for them?” Nadella replied, “You know, the thing that perhaps most influenced me in terms of how do you look at the journey or a career, there was this guy whose name was Mike Naples. He was the president of Microsoft when I joined. And he had this saying where he would say, ‘Look, all HR systems are long-term efficient, short-term inefficient.’ And I thought that that phrase just captured it. Which is, it's not really about asking for the raise, but knowing and having faith that the system will actually give you the right raises as you go along. And that, I think, might be one of the additional superpowers that, quite frankly, women who don't ask for a raise have. Because that's good karma. It'll come back because somebody's going to know that's the kind of person that I want to trust. That's the kind of person that I want to really give more responsibility to. And in the long term efficiency, things catch up. And I wonder—and I'm not saying that that's the only approach—I wonder whether taking the long term helps solve for what might be perceived as this uncomfortable thing of, hey, am I getting paid right? Am I getting rewarded right? Because reality is [that] your best work is not followed with your best rewards. Your best work then has impact, people recognize it, and then you get the rewards. And so you have to somehow think that through, I think.” As the backlash raged, Nadella tweeted “Was inarticulate re how women should ask for raise. Our industry must close gender pay gap so a raise is not needed because of a bias.”



***“I spent a lot of time on the road with customers. This can be incredibly frustrating and challenging at times, and there are no co-workers around to gripe to. My friends and spouse were also very far away. Social media became my release valve. I could express my difficulties and frustrations and get sympathy and sometimes even useful help.”***

who offered me a job at Pythian. Pythian is well known for being a place that hires and supports the best DBAs. I jumped at the opportunity, knowing that I would no longer be bored with my day job. That was certainly true. Pythian helped me learn and grow as a DBA and later as a consultant. Both skills were incredibly useful in my big data career.

***In your career thus far, did you find any doors closed or less welcoming to women? Have you had to fight to get the pay or the promotions you deserve? What is your opinion about Microsoft CEO Satya Nadella’s advice to women<sup>1</sup> in response to interviewer Maria Klawe’s question? Satya hasn’t had the experiences that women have had, but his response was probably colored by his own experience as an Indian immigrant: eventually the system recognized his potential and promoted him. Do you have different advice for women in technology?***

This is a very difficult question to answer. We all have career opportunities and career setbacks. Who can tell what the influencing factors were? It can be my gender, my accent, my personality, or the culture. It is impossible to tell in many cases. I am certain that if you ask yourself “Do I deserve this?” at every turn, good or bad, it is futile. As an engineer in the San Francisco Bay Area, I easily earn a very comfortable salary. Do I deserve to earn this much when some people work much harder and can barely feed their children? It is really difficult to know. Life is inherently unfair.

I envy Satya for getting all he did without ever having to fight or ask for anything. I’m sure this is his experience, but I don’t think this experience is universal. Some organizations are not very good at recognizing talent without some “pushing.” This is not always a gender issue—although I’ve learned that talent appears in many forms and often talented women will get less recognition for similar work. But a large part of success is recognizing what it takes to achieve your goals at a particular environment, and sometimes it takes asking.

One of Cloudera’s most respected directors gave a presentation called “How to become a technical leader,” and one of the lessons was “Ask for extra responsibility. Let everyone know you want to be a manager.” I prefer this advice. Many people are afraid to appear ambitious, but I’ve never seen anything bad happen to people who ask for extra responsibility.

The other lesson I’ve learned is that career opportunities hap-

pen only when people want to work with you. This means that first, people need to know you—networking was key to every career move I made. Second, you need to bring a lot of value and be accountable for your work. We all want to work with people who, when they say they’ll do something, they deliver. Perhaps this is what Satya meant when he said that good work is rewarded, but while good work is essential, it is rarely sufficient.

***Which big data products do you work with the most? What advantages do they offer over traditional relational database management systems?***

I work for Confluent on Apache Kafka. Apache Kafka isn’t a big data system per-se, although it is frequently part of big data architectures. Some people think Apache Kafka is a “real-time” system, but that can be misleading too. I like to think of Kafka as a very elastic event-streaming system. This means that you use Kafka to collect events about your business: orders, sales, transactions, calls, updates, etc. And you process these events at the rate that matches your business requirements, rather than using arbitrary timelines like hourly batches. We focus on flexibility and adapting to business needs rather than worrying if the data is big or small.

Previously, I worked mostly with HDFS, Hadoop’s distributed file system. I work a lot with Sqoop, which lets me copy data from relational databases to Hadoop. I work with Avro and Parquet, two file formats that define schemas for Hadoop files. Parquet is a columnar file format, which lets me efficiently run analytical queries. I use Hive and Impala; both are SQL engines for Hadoop.

I used all these tools to perform large-scale ETL in Hadoop and to perform some types of analytical queries in Hadoop.

For nerdy DBAs, working with open-source software is a real treat. There’s little need to guess what the optimizer is doing behind your back or why something is super-slow. You can read the code, run a debugger, and find the issue. If you are lucky, you can even fix it. I’m not sure I can go back to working on closed-source databases and all the strange tricks we did to figure out how the database actually works.

I’ve been working a lot on streaming ETL. This mode of ETL doesn’t have the large-scale efficiency of the usual batch ETL, but it gives organizations some of the analysis they need in real time. The tools I use for this are Spark Streaming as the stream-processing engine, Kafka as a reliable source of streaming data, and

***“The best [Big Data] conferences are Strata Data, QCon, and GOTO. However conferences are rarely for beginners, so I recommend that you start visiting after you have a bit of experience. This has been similar to my experience with Oracle conferences: when I was a complete newbie, I couldn’t find any session I could understand.”***

***“I like to think of Kafka as a very elastic event-streaming system. This means that you use Kafka to collect events about your business: orders, sales, transactions, calls, updates, etc. And you process these events at the rate that matches your business requirements, rather than using arbitrary timelines like hourly batches.”***

HBase as a NoSQL database where I can rapidly update “materialized views” with the streaming information. I’m still at the beginning of this journey, but so far I’m enjoying the change of pace to a radically lower latency.

***How and where can our readers learn about these products? Any books in particular? Tutorials? Conferences?***

I have to mention my own books: *Kafka: The Definitive Guide* and *Hadoop Applications Architectures*, which were published by O’Reilly. These books go over the open-source products in depth and teach you how to use them to build real-world applications and architectures.

Software vendors, including Confluent, have courses and certifications. I found Confluent’s classes very useful, and in general I take all the classes I can when learning something new. I feel this gives me a solid start when switching to a new technology.

The best conferences are Strata Data, QCon, and GOTO. However conferences are rarely for beginners, so I recommend that you start visiting after you have a bit of experience.

This has been similar to my experience with Oracle conferences: when I was a complete newbie, I couldn’t find any session

I could understand. It took me two years to appreciate what I heard in my first Oracle conference. The only presenter who is completely clear and interesting to attendees of all levels is Tom Kyte. I wish I knew how he does this.

***You’re an active user of social media, aren’t you? Of late, I’ve been getting “friend” requests from people I only know at a professional level. This seems strange to me because they’re colleagues, not friends. What’s the role, if any, of social media in our professional lives?***

As I mentioned in the beginning of the interview, social media plays a huge role in my career.

I spent a lot of time on the road with customers. This can be incredibly frustrating and challenging at times, and there are no co-workers around to gripe to. My friends and spouse were also very far away.

Social media became my release valve. I could express my difficulties and frustrations and get sympathy and sometimes even useful help. These days I tweet some of my ideas when preparing slides for conferences. This generates interest in my talks but also lets me discuss my ideas with people. My talks are often much improved as a result.

I also love it when people use social media to give me feedback about my presentations—what worked, what didn’t. I recently got book reviews via Twitter!

***Do you have a life outside work? What are you passionate about other than big data?***

When I’m not writing a book, I hike, bike, run, climb, and camp. California has amazing nature and I can’t get enough. ▲

*Gwen Shapira is a software engineer at Confluent. She studied computer science, statistics, and operations research at the University of Tel Aviv, and then went on to spend the next 15 years in different technical positions in the IT industry. She specializes in scalable and resilient solutions and helps her customers build high-performance large-scale data architectures using Apache Kafka. Shapira is a frequent presenter at conferences and regularly publishes articles in technical magazines and her blog.*

Copyright © 2019, Gwen Shapira



## DATABASE MANAGEMENT SOLUTIONS

Develop | Manage | Optimize | Monitor | Replicate

Maximize your  
Database Investments.



Quest™

# Crushing the IT Gender Bias

by Kellyn Pot'Vin-Gorman

*This article is an excerpt from the book Crushing the IT Gender Bias by Apress, April 24, 2019, ISBN 978-1484244142; Copyright © 2019 by Kellyn Pot'Vin-Gorman. Reprinted with permission.*

When my career first began, I was on a team of five women Database Administrators (DBAs). Within 9 months, one of the five who was hired at the same time as I was left the industry. She was young, single, had a degree in Computer Science (CS) with a focus on database technology, and had no children. I had difficulty understanding how I, a divorced mother of three, with a young baby and no CS degree, would make it if she couldn't.

Over the next 6 years, I was too busy with my own career and raising children to notice that I'd gone from an all-female DBA team to the lone woman on the team. As my career continued to advance, my traditional idea that women were Database Administrators and men were in networking and server administration changed to a point where the cultural norm that men were in IT and women were a rarity in the industry became the reality around me.

It wasn't until an unsettling situation experienced by a peer forced me to examine what the real culprit was, and I discovered most policies around discrimination and harassment were rarely capable of deterring from gender bias impacting diversity. I felt the need to speak out, but it was clear that I needed to begin research to understand it all. Bias is a complex biological mechanism that developed historically as part of heuristics and is built out of experiences and cultural upbringing. Historically, heuristic traits are built to protect us from consuming plants that look similar to those we already know are poisonous or avoid similar environment situations that previously put us in danger. We learn by experience and example, but to do so, a human will simplify and categorize their surroundings to ease the demand for deep investigation of safety concerns. To give you a less dangerous example of how heuristics works, if I were to give you a lime, you expect to be handed a small, round, green citrus. If instead you were offered a red, fingerling lime, a banana-shaped minority of the lime family originally from Australia, a percent-

age of people will have great difficulty accepting the fact that it's a lime. It doesn't fit within their expectation of what a lime looks like. This protective process that bias sources from has a significant purpose though, as it is used to identify environmental dangers, such as poisonous foods and physical threats, will also arise in this benign situation, resulting in a percentage of individuals rejecting the lime that doesn't meet their expected criteria. We are all subject to heuristics, both men and women, although some personality types are more dependent upon their heuristic tendencies. They don't like to be outside their comfort zone, breaking with tradition or open to change.

The overall goal of this book is If we don't know there's a problem, how can we fix it? As our use of heuristics is with a protective measure, it can be unconscious, and as part of bias is built off of culture and upbringing, heuristics can be integrated into our society seamlessly. A second goal of this book is to share with women in the industry insight, knowledge, and support to help them succeed.

My own journey has led me to realize how difficult culture and bias are to address, let alone change, but with change can come great growth and benefits for everyone. Without this change, the tech industry's growth is stunted; it's unable to evolve and develop to its full potential. The products produced by a nondiverse technical industry will be limited in the ability to fulfill the needs of the customers they're expected to represent.

## The Numbers Behind the Stories

A fact is a piece of information backed by evidence and data, unlike an opinion, which is based on personal experience and views. As bias is sourced from a person's experiences and point of view, it is only natural that some readers may doubt its existence. This chapter will include experiences, as well as the data behind them, to help us along on our journey.

## My Realization of Bias

After I'd been in the industry for a decade, I had one of the last opportunities to work with a female peer, who we'll call "Ann." Ann had been my lead DBA back in 2004, and I had a great re-

***“Culture and bias are difficult to address, let alone change, but with change can come great growth and benefits for everyone. Without this change, the tech industry's growth is stunted; it's unable to evolve and develop to its full potential. The products produced by a nondiverse technical industry will be limited in the ability to fulfill the needs of the customers they're expected to represent.”***



spect for her technical skills, as well as her capabilities as a leader. After I'd been a witness to her previous year filled with challenges and confusing management decisions, she was forced to leave the company we were employed at. A conflict had arisen between her and a male peer, "George," and continued to escalate without relief from management or HR until she finally resigned. I felt helpless on how to assist her as I was friends with both individuals involved and had recommended both to their roles which lead to their employment at the company. I knew, without a doubt, I was observing George's insecurities around Ann's natural leadership skills.

A database administrator's job is often high stress, and the company we worked for was more so than the average due to the demands. When disaster struck, Ann was calm and thoughtful, while George, no matter how technically skilled, was short-tempered and tended to lash out at others around him. As frustrated as I was with George's behavior, I was more frustrated with the management who couldn't see Ann's leadership skills and had promoted George over her just a month after hiring them both. When I'd made my recommendation to hire them, I'd clearly recommended Ann for the lead role and George to take on the demand of duties, as he was what I deemed, "a workhorse" (a role I relate myself to, so it's easy for me to recognize the type).

As the situation between Ann and George degraded, Human Resources had little ability to manage the situation, and Ann left the company, retiring from the technical world shortly after. I took Ann's departure hard as I attempted to decipher how I could have helped more. I departed the company in the coming months, losing both two peers and losing one friend. I lost Ann as a peer because I felt I had let her down, and I lost George as both because I wasn't sure who had let all of us down.

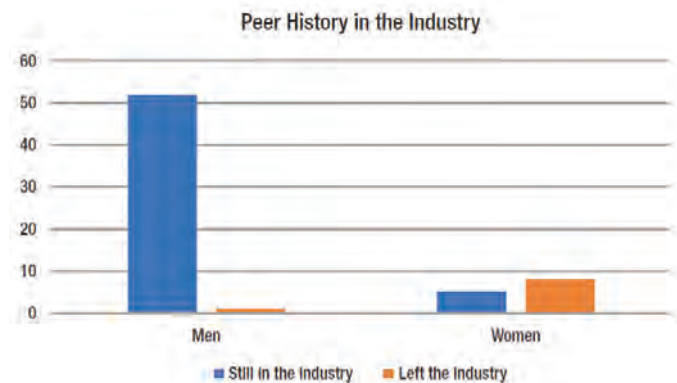
I soon after reached out to Ann and asked her to lunch. Upon our first meeting, the reasons she gave for leaving technology appeared valid, but the way she spoke, the sadness in her voice, and from my own knowledge of what she went through told me there was a lot more to it.

Ann was first hesitant to say more, but as I stressed, I was conflicted about what had happened; she seemed to feel relieved to have someone to confide in. I admitted that I had my own failure in helping her with what transpired, uncertain of how I could have better supported her at our previous company. I continued to speak with her at length, and in numerous conversations during the next few months, she described decades of frustrations and small, consistent challenges that had hindered her career. None of the situations were outright discrimination but were clear bias that were difficult to pin down. She'd attempted to address it but repeatedly found she failed miserably due to the gray area these situations fell into, and the more she confided in me, the more I realized I'd experienced many of the same challenges myself. Like Ann, I hadn't identified them as being gender related but simply thought it was something only I was experiencing. At this stage in my career, bias was a term used rarely, if ever, and here we were voicing what so many other women were silently cursing.

### The Disappearance of Female Peers

My experience with Ann happened in 2011 and two years before Sheryl Sandberg published her successful book *Lean In*. There was significant little being published about the challenges of women in technology in print or on the Web. As I discovered

more and more women at risk, I started to take a deeper look into the challenges to understand the source of bias in our world. At this time and in my own professional circle, I'd worked with over 50 men and 13 women as Database Administrators (DBA). With some quick research (Figure 1-1), I discovered that one of my previous male peers had left the database administration industry, having retired, but for the women, there was a drastic difference. Eight of the thirteen women I'd worked with had left the technical industry. Three had retired but retired early (in their 50s), while the rest had left to pursue different industries, often with more diversity. These were intriguing numbers, and I realized I needed to dig in and understand why.



*Figure 1-1. Peer history in the industry and the disappearance of female peers in my database career*

My natural curiosity makes me an acute observer, but even your own bias can put blinders on you. I suddenly faced this truth, realizing that my own bias had been present since my first job as a Database Administrator. I'd been hired, along with another woman, "Debbie," as a DBA, but neither of us had previous experience. Debbie had just graduated with a Computer Science (CS) degree specializing in database technology. She was single and didn't have any children, while I was married with three children, including a newborn, and I possessed no degree or certification in database technology. While she seemed to quickly acclimate with years of formal education, I was surviving on my natural knack and intelligence, in hopes no one would notice how little knowledge I possessed about databases. Debbie had skills, and I was the one "faking it until you make it."

Nine months into our employment, Debbie came to me and said, "Kellyn, I'm a black woman working in technology. It's just not working, and I've decided to go over to the project management team." Although surprised, I understood why she was leaving. We had poor leadership for our team, while the project management team was led by an incredibly gifted woman who was a natural leader. I remember thinking to myself, "If she can't make it with all this going for her, how the heck am I going to?" I accepted why she was leaving from a high level but didn't correlate how much the impact of bias, women in technology, and diversity was part of the equation. My own bias told me that it was just her unique decision and my own white privilege or similar challenges had nothing to do with my own successes and failures.

### Power in Numbers

The numbers (Figure 1-2) show how people of color are some of the least found in technology. Undistinguishing by gender,

blacks only make up less than 3% of tech positions, while Hispanics make up less than 5%.<sup>1</sup> The National Science Foundation calculated data on people who were employed in fields outside their tech degree (or unemployed with a technical degree) and found that the percentage of men and women of color were much higher than for white men.<sup>2</sup>

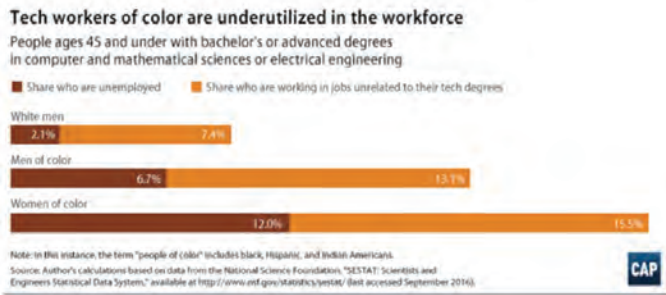


Figure 1-2. Underutilization of people in color in the technical industry

Many of us in the technical community have already noticed the lacking diversity, not just how few women there were in the industry. As I began to present at technical events, I noted how the few people of color would connect and that no matter the strength of initiatives for Diversity and Inclusion appeared to be, the ability to break through and have a true representation from people of color hasn't occurred.

The challenge may partly lie in economics and how we fund schools (Figure 1-3). Along with increased risk of poverty for people of color, schools are funded based on property taxes for a local district. If a student is already in a low-income family and belongs to a district that's low income, the school district will receive less funds in the way of local business and property taxes toward per student education expenditures than a district that has a larger influx of tax dollars.

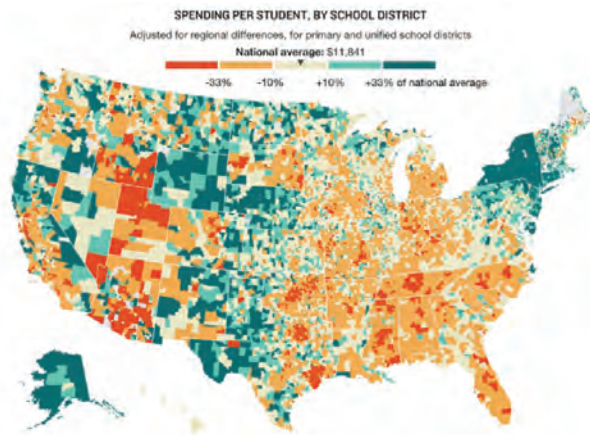


Figure 1-3. Spending per student, by school district in the United States

Adding to the challenge are the limited resources schools have and how education is focused vs. the workforce (Figure 1-4). When I was in high school in 1980, I was offered programming classes to learn Basic, as well as computer architecture. My children are attending school 30 years later, yet we have 41 states that still don't require any technical education toward a high school diploma.

Kids who are offered computer classes in school are more likely to be taught office technology (spreadsheets, word process-

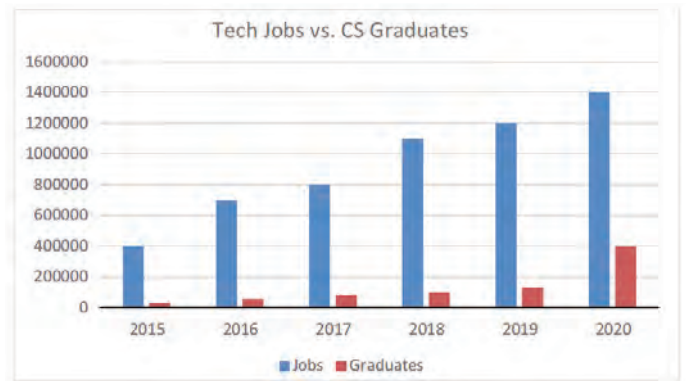


Figure 1-4. Open technical positions vs. Computer Science graduates

ing documents, and presentations) vs. actual critical thinking or programming education. Colorado has one of the highest average college educated populations in the United States, yet for my children to attend a class on Java programming, they would need to attend a vocational school. The vocational school in my district already has the stigma of attendance by those who can't graduate with the average population, so what does this say about the future of technology?

The Bureau of Labor and Statistics has estimated 1.4 million new technical jobs by 2020 that we will require a skilled workforce to fill, but per CS Education Statistics, the United States will have approximately 400,000 graduates in Computer Science to fill them.<sup>3</sup> The goal of education is to prepare young adults for the future, but if we continue to focus the bulk of secondary education (reading, writing, and arithmetic) on literature, history, and biology and still see technology as a "club" or "after school meetup," we will fail to ever teach technology as the center of our job industry future. How many English Lit majors and History majors are currently working in their field or without a job? We teach literature, history, and biology with the idea that a student may have the opportunity and passion to make a career of it. Shouldn't we be teaching technology with the same investment toward graduation? With the percentage of technologists that we'll require in the next decade, the way we view technical education has to change.

Therefore, I support many of the grassroots programs to bring technology to kids outside of the public-school system. Black Girls Code is well known, but Atlanta's Jeremy Harms, who runs Vine City Code Crew, is an incredible example of bringing code to inner city kids that may not have the opportunity to gain a passion from code at home, due to low income, or in schools due to lacking resources. Founder Kira Wetzel brought us Girls + Data, which I was first introduced to by Mindy Curnett, which offers girls the opportunity to learn in an environment that teaches code in a less intimidating way, finding the technologists of tomorrow. Locating these grassroots organizations who introduce technology to kids who might not otherwise get the chance

<sup>1</sup> "Delivering Growth through Diversity," McKinsey, January, 2018, <https://www.mckinsey.com/business-functions/organization/our-insights/delivering-through-diversity>.

<sup>2</sup> National Science Foundation Report, S&E Indicators, 2018, <https://www.nsf.gov/statistics/2018/nsb20181/report/sections/science-and-engineering-labor-force/s-e-labor-market-conditions>.

<sup>3</sup> 3CS Education Statistics, 2018 Estimates, <http://www.exploringcs.org/archives/resources/cs-statistics>.

isn't too difficult. Just open an account on meetup.com and do a search; more are being created every day as more realize how important it is to create builders of tech, not just consumers.

### Tech—A Woman's Place

As the years progressed, I found myself migrating from my first team of all female Database Administrators to practically the only female DBA in a team of senior DBAs (Figure 1-5).

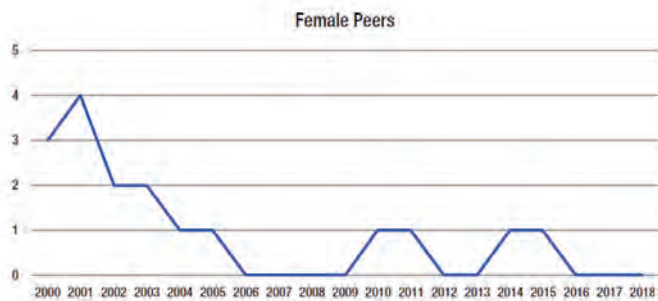


Figure 1-5. Number of women peers on my teams, past and present

Due to financial and personal demands, it fell to me to provide for my children financially, and a career as a DBA allowed me a flexible work schedule with comp time and remote work to address days when kids were home ill, parent teacher meetings, or other demands on my time as the sole-custodial parent.

***“The goal of education is to prepare young adults for the future, but if we continue to focus the bulk of secondary education (reading, writing, and arithmetic) on literature, history, and biology and still see technology as a “club” or “after school meetup,” we will fail to ever teach technology as the center of our job industry future.”***

When the average person thinks of a technical professional, the image that appears is often of two conflicting appearances:

- ▶ The first is of a guy, mid-40s with long hair, sloppy dress, sitting in the dark, in front of a gaming screen with a bag of Cheetos and an energy drink.
- ▶ The second is the traditional “nerd” with his neat but outdated apparel, pocket protector, and dark-framed glasses.

Neither of these “stereotypes” are true to form, as there is significant diversity coming into technology and more so every day, but due to this, technical jobs might not sound like the place for a young mother with children. The truth is, I found my database administration job more supportive of the demands of a working mother than most of my nursing and realtor friends had. If I had to work on a weekend or at night, I was often work-

ing from home. If I needed to take my child to the doctors or for a parent/teacher meeting, it was easy to do, as was staying home with a sick child and working from there. As technology advances, it gets easier to do so, too. I've been able to telecommute for the last decade, making the most of my hours and less in traffic, which allows me to be more productive for both the company and my family.

For those that see the old-school tech jobs of working 60–80 hours a week and needing to be in the office the first one in and the last one out, this is not because technology is a poor career choice but because the examples are a poor work environment and have poor leadership. It's not that it doesn't happen, but as my career matured, I found that I learned it's not so much about work/life balance as it is about your balance needs.

I've been judged harshly by peers and managers that questioned my dedication to my role when I'd need to work remotely or take time off to take a kid to the doctors, but I'd remind them to focus on what I accomplished in productivity and how accessible I was, even while maintaining my family responsibilities. That's what's important and what will correlate from work performed and the bottom line. There is a clear bias that correlates value to hours spent per week on the job, but it's a huge fallacy in the workplace. If you can do in 40–45 what another employee can do in 60–90 per week, which employee is more valuable? Is staying till 7–9 p.m. each night really providing any value if an employee that works from home can be available whenever needed? Are we assigning value for the productivity vs. busy work? We all need to learn how to interview companies to find out which they value, as it will decide how satisfied you are with your job, your career, and in the end, the company's bottom line.

As you guessed, I'm a strong proponent for flexible work schedules, especially those that can incorporate telecommuting. Although there is incredible value in face-to-face time between peers, managers, and customers, for technical jobs, there's considerable isolated work time that doesn't require an employee to be in an office setting. For these types of times, the employee was hired as a professional and should be treated as such, allowing for telecommuting opportunities. There is significant savings to companies by doing so (in the way of workman's compensation insurance, office space, and resources); the benefit is not one way. There are differences in how each gender is received when requesting flexible work schedules though.

In 2017, Furman University performed a survey of 600 work age adults, when presented what they thought was a conversation between an employee and HR, changing the conversation to include different requests for flexible work schedules or none at all, but would change the gender on the requests. Some of the requests were nonfamily, others were due to childcare constraints, but the reviewers were asked to judge the request or on likability, dedication to their job, dependability, and if they were the HR representative, would they approve the request. The reviewers scored the men as more likable by an average of 70% while the women only 57%. They deemed the men more dedicated and dependable, even though the requests were the same and only the gender had been changed.

As obvious as the research has shown greater potential for men to receive flexible work schedules without the same negative judgment, I've rarely experienced my male peers requesting the time off. I was more likely to observe them delegating family responsibilities to their wives, so they could avoid having to ask



***“How many English Lit majors and History majors are currently working in their field or without a job? We teach literature, history, and biology with the idea that a student may have the opportunity and passion to make a career of it. Shouldn’t we be teaching technology with the same investment toward graduation? With the percentage of technologists that we’ll require in the next decade, the way we view technical education has to change.”***

the boss for flexibility. I understood why they did so—it’s uncomfortable to have to ask for time off. As a divorced parent, I rarely had that luxury until I received a flexible work schedule as part of my employment. My ex-husband tended to be unavailable, so I admit, to avoid a disagreement with him, I found it easier to seek out companies that were supportive of flexible work schedules.

Along with flexible schedules, pay was important, but it is for most people. As I stated, I was the financially responsible parent, along with custodial one, so it rated high on requirements for a potential position.

It took me years to learn how to demand what I needed to be successful in tech. I first started with asking for what I needed to be more productive in a given day, then worked toward how to ask for what I needed to be more successful with my life and then from there, my career. Many of these changes to what is commonly offered us has to be sold to the company. Making the business see that investing in an employee can provide value to them should be a distinct and highly sought-after skill on any resume.

### **The STEM Challenge**

2011 was a significant year for me. After the departure of “Ann,” I began to reach out to numerous women that I’d discovered had left the industry. I was able to have some open and honest conversations with them, gathering answers to my questions to understand why, as well as document cultural patterns when they existed. Some of these peers were ready to talk with me about bias, seemingly relieved of the burden of silence, while others were more private about why they’d left. More often, I found they had; just like me, we were taught not to discuss topic of bias, and it wasn’t something we acknowledged. Society had taught us to be skilled in creating excuses to justify what had occurred and if we did raise our voices to the reality would risk deemed being “uncool.”

Many women in technology don’t fit into the traditional mold of the girl next door. We may have odd traits, interests, and behaviors that make people try to understand what makes us tick. To combat this, we still fall into an expected behavior. We become the “cool girl” instead of simply embracing who we are, and we may pretend to like sports, juvenile humor, and traditional male pastimes. As we like STEM (Science, Technology, Engineering, and Math), our more traditional female peers may have refused to accept us into their groups. Everyone wants to be liked, so it’s natural for us to try to find a group to be accepted into. If those of our gender won’t accept us, we will make concessions to be viewed as “not like other girls” in the cool image most accepted by the men around us. We often choose more masculine type pastimes and adopt less feminine traits which offer some protection from unwanted advances. No matter how far we stray

from it all, we are still products of culture and bias. As much as we like to claim we are our own person, we’re shaped from the time we are delivered from the womb—placed in blue or pink clothing, given boy or girl toys, while boys are told not to cry and girls are told how pretty they are, all before we are even crawling.

The numbers are astounding and will be built upon as we proceed through the chapters in this book. Between culture and bias, there is an ongoing challenge of acceptance, but with the initiatives of the last few years, I’ve gone from one of the few to speak out loud to one of thousands who speak up regularly. With all the initiatives, many worry that the numbers aren’t increasing as they expected, but when it’s grown stagnant, I haven’t been that surprised. I have been very successful with the mentoring that I’ve offered to others, but it’s never been traditional mentoring, and with the low numbers we began with, it simply wasn’t self-sustaining at this point. Similar to a small seedling, it requires care and feeding to build into something more until it has the roots and strong base to support itself.

I hope the book will offer the insight to others on how to mentor women in the way they need to succeed but also demonstrate how important sponsorship, networking, and giving women options that often are so unspoken by everyone around us. Culturally, boys are raised with “skills,” but also these important opportunities, where girls are left to figure it out and traverse this on our own. Women have little idea how to gain sponsorship from a professional peer, having more likely been taught how to flirt. We are instructed how to gain someone attention, but not how to network and build our contacts in a way that will promote our career. We are rewarded for being seen, but not heard, and are taught how to actively listen while others talk in a business world, those same individuals that wonder why we don’t speak up more often, without the upbringing to teach us how to succeed in so many unknown areas of business and at the same time often told that some of our greatest strengths are drawbacks in the business world.

Lucky for me, I’ve always been an individual to respect data over opinion. It’s not that I don’t hold value in experiences, quite the contrary. I have an incredibly observant and accurate gut instinct about situations but have always used data to ensure that my observations aren’t blinded by my own or other’s bias. This skill in observation has served me and those that search me out for this insight well. It’s a skill that is also used to inspect technical environments—leaving nothing off the table and gathering all data to understand complex situations. As bias is often unconscious, my own upbringing, gender, and experiences could cloud my view, so it’s important to collect data and research before making decisions, creating a plan of attack or creating a solution.

As I stated, when I first started noticing how many female peers were departing the industry, I began to do research. There was surprisingly limited data back in 2011. I often wasn’t positive

what to research, but I knew patterns were evolving in my observations, and there had to be reasons for them.

The Stanford gender studies site and subsequent Ted Talks from Sherry Cornell offered the first data that explained the patterns I was privy to. The site articles were some of the first documented findings on gender bias in a scientific manner, providing the numbers that matched the patterns I'd started to document. Sherry's talks were well received by me as she viewed the data

***“The numbers show how people of color are some of the least found in technology. Undistinguishing by gender, blacks only make up less than 3% of tech positions, while Hispanics make up less than 5%. The National Science Foundation calculated data on people who were employed in fields outside their tech degree (or unemployed with a technical degree) and found that the percentage of men and women of color were much higher than for white men.”***

with a scientific approach based on educating everyone. Her talks are enlightening, and the data is difficult to ignore, even if there are many who have trouble accepting it. This is an overwhelming topic, so I'm not one to push but always open to discuss. One of the most positive aspects of focusing on education and open conversation is the number of people who want to have these conversations.

There are significant differences in the challenges that not only depends on culture, but location. I live in Colorado, which has a significantly high percentage of women leaving the industry over the national average, even though we have one of the highest percentages of college educated residents. While the national average is women will leave the technical industry within 4 years in a career, Colorado is less than four. The state of Utah is our neighbor, yet due to location and culture, a fellow president for a user group there isn't as worried about retaining women in tech. Her biggest challenge is to get women into secondary education, as many of them are expected to marry right out of college. The difference is cultural, as there isn't just an educational difference between the two states, but a religious one.

### **The Accidental Technologist**

I came into technology approaching the age of 30, in the late 1990s, which was a significant time for database careers. I'm unashamed to admit that I don't have a Computer Science degree. I consider myself having little to no college that I can fall back on due to a medical condition that impacted the first half of my 20s.

I was 22 years old when I started to experience odd migraines that didn't follow the pattern of the migraines I'd suffered from

since my early teens. These new “migraines” had occurred about 6 months apart, were a crushing pain, often slicing up through the right, back side of my brain, and included numbness and vision loss. The loss of vision was primarily the left side, as was the loss of mobility. The pain would last a few hours, and once it was gone, I was listless and would sleep for a day or two. As doctor visits were quite costly and I was an auditor at the time without a lot of financial stability and caveats for everything insurance didn't want to cover, I was hesitant to seek medical attention unless it was necessary.

After a few of these migraines and a listless 2-day sleep, I realized I was still missing about 10% of my left visual field which was reason enough to seek out professional medical attention. I was referred to a neurologist, who after performing an MRI, informed me I'd had a stroke and it wasn't my first one. The 10% visual loss was permanent, but my “stroke” wasn't the standard variety. Most strokes occur due to high blood pressure and blockage, yet blood tests revealed that I had a bleeding problem and my blood pressure was incredibly low. After further blood tests revealed that I was autoimmune positive, a rheumatologist was brought in to start researching.

Over the next 4 years, I suffered five documented strokes. By 1993, I was told that I'd either be blind by the time I was 40 (best case scenario) or dead (worst case scenario). I promptly sought out a second opinion and, luckily, found a neurologist that was interested in figuring out what was the cause, and with autoimmune, you find out that it's unique for everyone. For me, my pregnancy with my first son, born at the very end of 1994, offered a large part of the answer by putting me into remission, and once my autoimmune returned in 2004, it was first joint related (similar to rheumatoid arthritis) and then settled into a connective tissue disorder by 2014 but has never returned to the debilitating vascular condition it once was.

I'd still suffered significant damage from the years of strokes. I was missing almost 9 years of memories; 48% of my left visual field required speech and physical therapy and could no longer perform the duties of my previous position as an auditor. While my brain was healing, I needed to continue to make a living, so I worked at Kinney Shoes. Yes, I sold shoes in a retail store. I was slowly recovering as I went through physical therapy, and my speech patterns were impacted, so speech therapy was required, too. Most don't realize it today, as I am a professional speaker, but I still have some issues with “the thesaurus of my brain” and will falter to pronounce some words. Just be patient with me—it's no big deal considering how far I've come. Outside of this and my missing visual field, not many signs of what I went through over two decades ago.

It wasn't a simple recovery though. I recall working at the shoe store 25 years ago and sitting in front of a pile of shoes that I was to organize by size on a set of shoe racks. It took me a couple days to perform this simple task. I was incredibly frustrated as my brain simply couldn't figure out how to put the size five first, then the five and a ½, then the six, and so on and so forth. The thing was, I've always been incredibly persistent. I've had bosses call me stubborn and sometimes a bit obsessive, but it served me well as I recovered from my strokes. I just simply refused to give up and just kept trying until I worked my way through the shoe rack and placed all the shoes in order. I approached every challenge this way as my brain healed, succeeding rarely the first time, but refusing to give up and persevering in the end.

*“I found my database administration job more supportive of the demands of a working mother than most of my nursing and realtor friends had. . . . I’ve been able to telecommute for the last decade, making the most of my hours and less in traffic, which allows me to be more productive for both the company and my family.”*

I had a long way to go, too. I no longer had any of the knowledge I had acquired as a young adult. I had to relearn how to drive, and my sense of direction had been impacted. We won’t discuss how many times I found myself on the opposite side of the city vs. where I was going, and if there’s one thing you learn about Denver, it is that it’s quite a simple city to navigate. There’re mountains only on the west side, and you can see them from almost any location. It’s hard for me to recall how I couldn’t tell which direction I was heading. I no longer knew how to balance a check book or other basic skills I’d learned. As with the shoe rack, I continued to make an attempt until I finally succeeded.

With the loss in memory, about 6 months after my last stroke, my brain fog began to lift. I discovered myself married to a man that I didn’t really remember dating, let alone marrying, and I was 5 months pregnant with my first child. I worked to heal, become independent, and became a single parent less than a year after my last stroke. It should have been a terrifying prospect, but I think I was too busy working toward succeeding at little things daily to think too much about the risk I was taking. I look back now and can say, without a doubt, that nothing seems as daunting as where I was then, and it may be why I seem so fearless now.

As my brain healed, I went from selling shoes to selling computers at Circuit City. The company discovered that I had a knack for software, and I began doing desktop support. I had married the father of my two younger children, and he advised me to become a desktop support specialist for US West (now Centurylink). He made it clear that I was ready, as he had “guys making \$40K a year that can’t load Microsoft Windows at a C:/ prompt.” I started as a desktop support person with no experience working in a networked or office environment. As is the case now, I took on the challenges no one else would touch, which turned out to be database applications. Few appeared to have expertise in the area, but I read whatever I could find and worked hard to figure out what I could on my own. Oracle was 16 bit at the time, version 7.3.4 had just been released, and I found it ridiculous that those using more than one Oracle application with different versions were expected to run each application on a separate workstation. I figured out a way to switch the win.ini file in Windows 95 to allow a change to the path to the Oracle directory, allowing users to do what they said shouldn’t be possible—one workstation/multiple Oracle applications. Oracle asked if they could send a representative to come in and review my scripts and configuration and upon completing the review advised the Chief Technical Manager at US West to “make a DBA” of me.

Little did I know, as I worked for multiple companies, progressing my career, that I would end up working for Oracle 15 years later. My career delivered incredible opportunities to work with fantastic individuals, mentors, and people I idolized in the industry. It taught me how to avoid some of the pitfalls that I experienced earlier on, and my learned persistence, mentors, and sponsorship were the difference between my continued success in technology and missed opportunity for others. My own expe-

rience, my ability to have others share their own experiences with me, and the data that I’ve collected have provided me an incredibly big picture of the challenges and opportunities for women in technology (WIT).

What I learned from all my experiences is that everything happens for a reason, every challenge has a benefit, and that there is incredible power in the simple act of doing.

My own journey, with bizarre medical history and nontraditional path into technology, also relates in many ways to why these stories need to be shared. A majority in the field aren’t college graduates with Computer Science degrees. In fact, it’s more likely that they’ve come from diverse backgrounds, were discovered to have a natural affinity to technology, or fulfilled a need and became an “accidental technologist.” Each is a complex life form, and understanding who each of them are requires a deep discussion around gender, upbringing, and culture and must include research data. My goal here is to broach topics honestly and discuss that we’re a product of our upbringing, our experiences, and the opportunities we’ve been given and how it can make the difference between success and failure. ▲

Copyright © 2019, Kellyn Pot’Vin-Gorman

## DATABASE RECOVERY HAS NEVER BEEN SO **SIMPLE!**

Axxana’s award winning **Phoenix System** offering unprecedented Data Protection, Cross-application consistency, Storage and Replication agnostic.



**ORACLE**  
PARTNER NETWORK

**AXXANA**  
BUILT TO LAST



info@axxana.com • www.axxana.com



# Exam Ref 70-535 Architecting Microsoft Azure Solutions

Book notes by Brian Hitchcock

## Details

**Author:** Haishi Bai

**ISBN-13:** 978-1-5093-0468-4

**Publication Date:** June 2018

**Publisher:** Microsoft Press

## Introduction

This book, we are told, will teach us how to take a set of requirements and deploy a best-in-class solution running in an Azure environment. The organization of the book and Microsoft certifications are reviewed. Links are provided for more training via Microsoft Virtual Academy and other online references. The final section is labelled “Important” and tells us how to use this book to prepare for the exam.

## Chapter 1—Design Compute Infrastructure

We start off with the author’s definition of the cloud: “The cloud is a huge pool of resources that supports a variety of services.” Sounds like something I should remember for the interview! These resources are storage, compute, and networking, and you can get any amount of them you need. You don’t have to worry about managing any of the infrastructure that supports these resources. Similarly, you can reduce the resources you are using at any time and they “return to the cloud.”

You can run services that consume these resources, such as virtual machines (VMs) and virtual networks to build Infrastructure as a Service (IaaS). Platform as a Service (PaaS) and Software as a Service (SaaS) can be built from there. This chapter then presents five “skills,” each of which is a topic for the exam.

### Skill 1.1: Design Solutions Using Virtual Machines

Azure supports both Windows and Linux VMs. Creating a new VM is easy; you don’t have to worry about hardware. Note that Azure does not apply software patches. The revolution will be televised in the cloud but it will not auto-patch! Images are available for various combinations of processor and memory to support your specific project.

Azure provides tools and processes for every challenge of DevOps. You create VMs using three architectural constructs: Availability sets, Fault Domains, and Update Domains. These are, respectively, resources grouped across multiple Availability sets, resources that fail as a group, and resources that get upgraded as a set.

For example, a patch would be applied to all resources in a single Update Domain. A great deal of detail is provided to explain how to set all this up. Screenshots show the actual steps. I’m not sure how fast these screens are changing, so what is presented may be out of date.

There are many tables of options, one of which is virtual machine sizing. We have a real alphabet soup of virtual machine sizing, such as B, Dsv3, F, Ls, and A8-11, among many others. I don’t see any order or pattern to these VM sizes, which is confusing.

Further sections cover managing images, custom images, and custom script extension. State management is used to check the consistency of a large number of VMs, part of DevOps using Chef and Puppet. Azure Batch supports High Performance Computing and other large jobs. Backup options are covered.

### Skill 1.2: Design Solutions for Serverless Computing

Serverless computing is new to me. It’s another level of abstraction where you don’t set up VMs but they are created in the background. This can be triggered in near real time by events in the cloud. This sounds pretty cool—just be aware, though, that you may set up something that when triggered creates a whole bunch of VMs . . . and you will get a bill later.

Azure Functions will execute small sets of code in the cloud that support triggers and can be used to speed up development. Many examples and options are shown.

API Management is provided to publish APIs to developers that may be external or with partners as well as employees.

### Skill 1.3: Design Microservices-Based Solutions

Microservices-based solutions are small solutions that are loosely coupled; each is a part of a larger, more complex task. This allows you to pick the best solution for each part of the complex task. The solution options are many and include container-based, container-orchestration, and Azure Service Fabric as well as Web APIs. Each of these is discussed as well as when each is the best solution to use.

You need to decide how you will migrate existing assets to the cloud. For example, you can move your existing VMs into Azure as they are, but you don’t get the advantages of a cloud-native deployment. I hadn’t thought about this. If you don’t have time, you just lift and shift. If you do have time, you can “go native” and get more of the benefits of the cloud. One example is that in the cloud your application can have a dedicated network separate from all other applications. If you just lift and shift, you don’t get this dedicated separation.

DevOps is discussed in detail, covering how to best integrate people, processes, and tools. Lifecycle strategy is also discussed.

### Skill 1.4: Design Web Applications

Here we learn about all the back-end services and constructs to support web applications. There are sections on how to design web applications that run in the Azure world. The service-level agreements (SLAs) for these services are documented; all are

99.9% available, by the way, for services such as Active Directory and Virtual Network. We are also referred to a link that presents the legal definition of an SLA. Other topics include how to improve availability from single-component to multiple instances. I liked the comments on transient errors and how these are quite elusive. I'm not sure how these comments figure into that legal-SLA discussion. Loose coupling, health monitoring, and system-level availability are discussed. You can design a custom Web API as well, and the REST model is described in detail. Secure Web API, scalability, and performance of a website in Azure are also covered.

### **Skill 1.5: Create Compute-Intensive Applications**

This refers to Big Compute, a term I had not seen before, which means coordinating a lot of tasks in a distributed manner. This includes fluid-dynamics simulation and genome search, tasks that require many hours of compute resources. How to design high-performance computing (HPC) applications using Azure Services is shown first. Azure Batch allows you to schedule and manage large parallel workloads without managing the cluster infrastructure on your own. When to use Azure Batch is explained. To support scalability, you use stateless components in your design. Azure Batch also has a design lifecycle of its own.

Each chapter ends with a Thought Experiment in which you are given a situation and asked questions related to the chapter. Answers are provided.

## **Chapter 2—Design Data Implementation**

Here we see how to store data in Azure. You can choose from storage for unstructured data in files to databases and what is called Big Data storage.

### **Skill 2.1: Design for Azure Storage Solutions**

All workflows use and process data, most of which needs to be stored and retrieved later. You have Azure File and Azure Blob to choose from.

Azure Blob stores a set of binary data as a file; there are three different types: block, append, and page. “Block” is meant for large files, “append” means you can only add to the file, and “page” is optimized for read and write. Data can be stored as Hot, Cool, and Archive access, with each being progressively slower and less costly.

There are many options for redundant storage to provide high availability and recovery.

Azure Files are used to store data that will be accessed by users and applications. Files are set up with four levels: a Storage Account and at least one Share, followed by directories and then files. This seems overly complicated to me.

Azure Disks are created when a VM is created—one for the operating system and one as a temporary disk—and there are four types of disks based on the I/O speed you need and whether Azure manages the disks or you do.

### **Skill 2.2: Design for Azure Data Services**

Once data has been stored, it needs to be processed, transformed, and shared. Azure provides Data Services to support a central data source repository, orchestrate data transformation, store historical data for further analysis, and present data after it has been processed.

Data Catalog provides central storage and management of all your data sources. Data consumers and producers can find out about each other.

Data Factory is used with Big Data to store, process, and transform lots of raw data—data that may not have much value until it has been processed.

SQL Data Warehouse service provides relational tables for columnar storage for data once it has been processed by the Data Factory.

Data Lake Analytics provides on-demand analytics of the data sets produced from Big Data and uses an evolution of SQL called U-SQL that extends SQL with C# capabilities.

Analysis Services are used to present data after it has been processed and modeled; it is based on SQL Server Analysis Services.

HDInsight is a cloud-native solution for deploying frameworks such as Hadoop and Kafka.

### **Skill 2.3: Design for Relational Database Storage**

Azure offers the option of moving databases to the cloud via a managed service. The Azure SQL Database, based on SQL Server, can deploy and consume databases without users needing to know the details of how to deploy SQL Server. You can also use a Stretch Database to move your on-premises, rarely accessed data to the cloud. Sharding is supported for scale-out, and options include single-tenant and multi-tenant. Support is also available for MySQL or PostgreSQL engines. Various options are explained to deal with failures, such as multiple replicas, direct storage, and remote storage. How to design a backup-and-recovery strategy is covered as are tips for using columnar storage.

### **Skill 2.4: Design for NoSQL Storage**

Some use cases, such as objects that don't have defined schemas, don't fit well into a relational database, and this is where NoSQL storage fits in. The benefits include simpler horizontal scale, flexible data structure, schema-free, and simpler APIs. You still have SQL features such as indexes and use of a structured query language. Among the NoSQL options we have Redis Cache, an open-source in-memory data structure storage engine. Table Storage is a service that stores key-value data in the cloud, supporting a schemaless design. Data Lake Store is based on Hadoop Distributed File System (HDFS), useful for Big Data. Azure Search can easily add search capabilities to your applications. Time Series Insights helps analyze data that is time-based, which supports data streams from IoT devices.

### **Skill 2.5: Design for Cosmos DB Storage**

Cosmos DB is a NoSQL service that offers features not available in Table Storage, such as secondary indexes, global distribution, and a latency guarantee, and supports moving MongoDB or Cassandra storage to Azure. Many details about Cosmos DB are presented. There are sections covering the API for MongoDB, SQL, Graph, and Azure Tables.

There are also tips on how to plan the deployment of a Cosmos DB to address cost, performance, and availability concerns.

## **Chapter 3—Design Networking Implementation**

When you create a new VM on Azure, physical access to the hosting machine is prevented, and you use the machine through remote connections such as SSH. Azure Virtual Network supports private networks.

### **Skill 3.1: Design Azure Virtual Networks**

We're told that it's easy to set up a new virtual network, and a detailed example of a new network with two subnets is given.

Azure CLI is the command-line tool provided to manage virtual networks. IP and private IP address options are discussed, and two sample topologies are shown. How name resolution is handled is also shown.

The options for load balancing are explained. Availability sets consist of multiple fault domains and update domains. Internal and public load balancers are provided. The ARM object model—wherein all Azure artifacts are seen as resources with hard and weak dependencies—is reviewed. Traffic Manager provides global routing services for your globally available web applications. Since Azure has servers in 30+ parts of the world, you will also need to understand the options for CDN within Azure.

Routing determines how data gets moved around your VMs within Azure. You can set up System Routes, VM-to-VM routes, VM-to-Internet routes, and User Defined Routes (UDRs).

### **Skill 3.2: Design External Connectivity for Azure Virtual Networks**

It takes time to move existing enterprise applications to the cloud, and not all of them will ever be fully moved. Some applications will remain in-house and some will be in the cloud, a hybrid solution. This requires connectivity. Azure offers several VPN topologies: Point-to-Site, Site-to-Site, ExpressRoute for Azure datacenters, and several others. Each is explained with diagrams. Azure offers Network peering, which reduces latency by eliminating the need for gateways.

### **Skill 3.3: Design Security Strategies**

Azure offers the same network security features you would have in your on-premises networks, such as isolation, firewall, and access controls. Network Security Groups (NSGs), their associated rules, and applying NSGs are covered with some very detailed examples. The Azure Application Gateway is a Level-7 load balancer with many security features, each of which is covered. Scenarios are discussed, ranging from simple situations to protecting against common web vulnerabilities and end-to-end SSL.

### **Skill 3.4: Design Connectivity for Hybrid Applications**

In addition to the network capabilities discussed so far, there are other services needed for the hybrid cloud world. Azure Service Bus Relay allows your applications to run on both Azure and your own on-premises data centers. Hybrid connections connect Azure App Services to your on-premises resources. This supports frameworks such as .NET, PHP, Java, Python, and Node.js. Azure also offers Web Apps virtual private network capability as well as options for domain-joining virtual machines.

## **Chapter 4—Design Security and Identity Solutions**

This chapter begins by telling us that the internet is a hostile environment; in order to set up security we must have deep network security knowledge and lots of experience in anti-hacking techniques.

### **Skill 4.1: Design an Identity Solution**

The basic idea here is that you should delegate identity management and authentication to a trusted service so you can focus on, as the book puts it, “business logics.” I hadn’t seen the term “logics” before. We first learn the definitions of managed identities and claim-based architecture. These are identities that are managed by a trusted party and form the basis of the claim-based architecture. The basic authentication and authorization work-

flow are explained. If you have any experience with Microsoft products, you won’t be shocked to learn that Azure provides Active Directory to manage identities. The differences between Active Directory that you have on-premises and Azure Active Directory are covered. A detailed example of how to set up a basic authentication/authorization workflow is presented. Microsoft Graph API is offered as the way to interact with user profiles in the cloud. How to do this, including code samples, is shown.

### **Skill 4.2: Secure Resources by Using Identity Providers**

Here we see how to design solutions to use external identity providers such as Facebook and Google. Given what has gone on in recent years, it’s hard for me to think of Facebook as a trusted party. A simple scenario is covered where you configure ASP.NET to use Microsoft Account as the identity provider. Additional sections explain when to use Azure B2C, the identity management solution for your applications, and Azure B2B, for work with other organizations.

### **Skill 4.3: Design a Data Security Solution**

Having secured identities in Azure, it’s time to secure your data as well. Data protection covers data at rest, in transit, and in use. Azure data centers provide the best protections for customer data, we are told. Next, we learn about data encryption. Azure has several data repositories, each of which has different encryption support. Options include Azure Storage, one of the fundamental services of Azure, which allows you to use your own encryption keys versus using Azure-managed keys. SQL Database offers Transparent Data Encryption (TDE) and Column-Level Encryption (CLE). You can keep encryption keys in the master database or an extensible key management architecture. Other features covered include access control in Azure Storage, data reliability and disaster recovery, Azure Backup, and Azure Rights Management Services.

### **Skill 4.4: Design a Mechanism of Governance and Policies for Administering Azure Resources**

This section explains how to use Azure services to manage large numbers of users whose access spans many services, locations, and devices, and the changes needed to handle all of these users and their access. Sections cover when to use Azure RBAC roles, how to set up an RBAC strategy, and resource policies. RBAC self-service allows users to take on some access management tasks, such as resetting passwords. Azure AD supports Privileged Identity Management, Managed Service Identity, and HSM-backed keys. All these options are covered.

### **Skill 4.5: Manage Security Risks by Using an Appropriate Security Solution**

You need to identify and deal with attacks using smart strategies. I like the diagram of the red team kill chain attacking while the blue team kill chain defends. Be careful, though: using “kill chain” too often in emails and messaging apps might bring some unwanted scrutiny from corporate. How to do this, when to use Azure AD Identity Protection and Advanced Threat Detection, and how to design your endpoint protection strategy are all covered. The Azure Security Center is discussed, and screenshots of the Operation Management Suite Security and Audit Solution are shown.

## **Chapter 5—Design Solutions by Using Platform Services**

Azure provides platform services for AI, global messaging,



and massive media processing. Simple APIs are provided for you to use all of these services. You don't have to create your own services and you can build applications that use these services at scale.

### Skill 5.1: Bring AI into Your Applications

I've been around long enough to remember when AI was the wave of the future, back in the 1980s. The text tells us that AI has gained tremendous momentum in recent years. This section covers the basic concepts of AI and machine learning. The amount of data available is rapidly increasing, and the ability to derive value from all that data is increasing. I appreciated the quote regarding valuable outcomes, "many of which are not well understood or anticipated." Indeed. Be careful on the road to the future.

### Skill 5.2: Design Scalable, Reliable, and Performant IoT Solutions

Here we learn the structure of an IoT solution. We see the control plane where devices are managed; the compute plane where devices are connected through a mesh; and the server pipeline where we extract, aggregate, and generate new business value. I learned that just deploying some code on a single device does not qualify as an IoT solution. How to do all this using Microsoft IoT Suite is covered. This includes Azure IoT Hub and Event Hub as well as Time Series Insights, IoT Edge, and many other services.

### Skill 5.3: System Integration and Reactive Systems Through Messaging

Messaging systems have been used for a long time; recently, with the Actor Model paradigm and Reactive Programming, they are being used in new applications in the microservices world. Azure has lots of options for messaging systems, and this section covers how to design message-based integration solutions and how to use Azure Image Build reactive systems, Azure Functions, and Logic Apps as well as send push notifications to mobile devices. System integration patterns are supplied, and specific messaging services are covered in detail.

### Skill 5.4: Build Large-Scale Media Processing Applications

The goal here is to understand Azure Media Services, how to use Azure Media Indexer, and how to monitor media services. Media Services is a PaaS that can be extended for scalable media delivery and management applications. This means you can securely upload, store, and package audio and/or video for streaming delivery. Entire workflows are possible. The Azure Media Indexer is used to make your media files searchable and to index your files.

## Chapter 6—Design for Operations

Azure provides tools to monitor and manage your systems and automate deployment.

### Skill 6.1: Design an Application Monitoring and Alerting Strategy

Applications and systems generate data that is collected and analyzed by monitoring services. Acting on that data is how availability, performance, and overall health are ensured. You need to figure out which of the Azure services are needed and how you will analyze logs and set up alerts using Log Analytics. Azure Monitor analyzes performance metrics and enables alerts.

Application Insights monitors applications. There are many services to choose from and all of them are covered here. Several examples walk us through the process of configuring some of these services.

### Skill 6.2: Design a Platform Monitoring and Alerting Strategy

Similar concepts for platform monitoring are presented here. Azure Health, Advisor, and Activity Log are explained. Also covered is how to set up a monitoring solution for Azure Networks using Log Analytics and Network Watcher. The Azure Security Center is used to monitor security, and screenshots of all the options are shown.

### Skill 6.3: Design an Operations Automation Strategy

Here we see how to automatically execute tasks, which is the basis of DevOps. Options include Azure Automation, Chef, Puppet, PowerShell, and several others. When to use each of these is explained. You also will need a strategy to determine when these automated tasks are enabled. Azure Automation executes PowerShell or Python scripts using runbooks that you create graphically. Desired State Configuration (DSC) is an automated method of forcing systems to conform to a specified configuration. Azure DSC is discussed and an example is shown with code samples.

## Conclusion

Now that I have read about both Amazon AWS and Microsoft Azure, one thing is clear: I don't see how anyone could make an informed decision as to which cloud vendor to choose. Why? There are too many details related to how each provider implements a vast array of services, solutions, and features—and all of them are constantly changing. How can you know that the one feature (or many features) that caused you to choose one vendor will be available from that same vendor next month? Or that the other vendors won't have those features next month? And all of the features are evolving (mutating?) all the time. I don't envy anyone who has to make such a decision.

I also found that the Azure world is more complicated and a lot more confusing than the AWS world. This may not be fair; I've only seen one written description of each. From this book, the Azure world seems to have a lot more pieces and parts, and it wasn't clear to me how they all fit together. ▲

---

*Brian Hitchcock previously worked for Oracle Corporation, where he had been supporting Fusion middleware since 2013. Before that, he supported Fusion applications and the Federal OnDemand group. He was with Sun Microsystems for 15 years (before it was acquired by Oracle Corporation), where he supported Oracle databases and Oracle applications. Brian's contact information and all of his book reviews and presentations are available at [www.brianhitchcock.net/oracle-dbfmw/](http://www.brianhitchcock.net/oracle-dbfmw/). The statements and opinions expressed here are the author's and do not necessarily represent those of Oracle Corporation.*

Copyright © 2019, Brian Hitchcock

# Third International NoCOUG SQL Challenge

Sponsored by Pythian—*Love Your Data™*

Originally published in the November 2012 issue

The Third International NoCOUG SQL Challenge was published in the May 2012 issue of the *NoCOUG Journal*. The winner, Lukasz Pluta of Poland, will receive an Amazon Kindle from contest sponsor Pythian and—in keeping with the pithy pronouncement of Steven Feuerstein that “*some people can perform seeming miracles with straight SQL, but the statements end up looking like pretzels created by somebody who is experimenting with hallucinogens*”—he will be made a knight of the August Order of the Wooden Pretzel.

## Mind-Boggling Puzzle

The Wicked Witch of the West had invited six friends to the Third Annual Witching & Wizarding Ball at Pythian Academy of Es-Cue-El & No-Es-Cue-El. Burdock Muldoon and Carlotta Pinkstone both said they would come if Albus Dumbledore came. Albus Dumbledore and Daisy Dodderidge both said they would come if Carlotta Pinkstone came. Albus Dumbledore, Burdock Muldoon, and Carlotta Pinkstone all said they would come if Elfrida Clagg came. Carlotta Pinkstone and Daisy Dodderidge both said they would come if Falco Aesalon came. Burdock Muldoon, Elfrida Clagg, and Falco Aesalon all said they would come if Carlotta Pinkstone and Daisy Dodderidge both came. Daisy Dodderidge said she would come if Albus Dumbledore and Burdock Muldoon both came. The Wicked Witch of the West needed an Es-Cue-El or No-Es-Cue-El spell to determine whom she needed to persuade to attend the wizarding ball in order to ensure that all her invitees attend.

## Mind-Blowing Solution

Master sorcerer Lukasz Pluta simply noted that if we start with a *minimal* set of invitees, then we can *augment* it—exactly one invitee at a time—by applying a sequence of rules until *everybody* has been included. Therefore, if we start with the *complete* list of invitees, we can *prune* it—exactly one invitee at a time—by applying the *same* sequence of rules but in *reverse* order until we find a *minimal* set of invitees. Lukasz assigned a power of two to each invitee (Albus = 1, Burdock = 2, Carlotta = 4, Daisy = 8, ...) and used binary arithmetic in a recursive SQL query; the anchor member of the query produces the complete list of invitees, while the recursive member prunes one invitee at a time. Here is his stunningly simple solution (with modifications for readability and efficiency); if there is more than one minimal set, then all of them are listed:

```
CREATE TABLE invitees
(
  invitee_id INTEGER,
  invitee_name VARCHAR2(128)
);

INSERT INTO invitees VALUES (1, 'Albus Dumbledore');
INSERT INTO invitees VALUES (2, 'Burdock Muldoon');
INSERT INTO invitees VALUES (4, 'Carlotta Pinkstone');
INSERT INTO invitees VALUES (8, 'Daisy Dodderidge');
INSERT INTO invitees VALUES (16, 'Elfrida Clagg');
INSERT INTO invitees VALUES (32, 'Falco Aesalon');

CREATE TABLE rules
(
  rule_id INTEGER,
  they_will_come INTEGER,
  if_they_come INTEGER
);

-- Burdock and Carlotta will come if Albus comes
INSERT INTO rules VALUES (1, 2 + 4, 1);

-- Albus and Daisy Dodderidge will come if Carlotta comes
INSERT INTO rules VALUES (2, 1 + 8, 4);

-- Albus, Burdock, and Carlotta will come if Elfrida comes
INSERT INTO rules VALUES (3, 1 + 2 + 4, 16);

-- Carlotta and Daisy will come if Falco comes
INSERT INTO rules VALUES (4, 4 + 8, 32);

-- Burdock, Elfrida, and Falco will come if Carlotta and Daisy come
INSERT INTO rules VALUES (5, 2 + 16 + 32, 4 + 8);

-- Daisy will come if Albus and Burdock come
INSERT INTO rules VALUES (6, 8, 1 + 2);

WITH working_sets(lvl, working_set, removed_id, is_leaf) AS
(
  -- The anchor member of the recursive query
  SELECT
    1 AS lvl,
    -- The complete list of invitees
    SUM(invitee_id) AS working_set,
    0 AS removed_id,
    0 AS is_leaf
  FROM invitees
  UNION ALL
  -- The recursive member of the query
  SELECT
    lvl + 1 AS lvl,
    -- Remove one invitee from the list
    CASE WHEN i.invitee_id IS NOT NULL
      THEN s.working_set - i.invitee_id
      ELSE s.working_set
    END AS working_set,
    invitee_id AS removed_id,
    -- Flag the leaf nodes
    CASE WHEN i.invitee_id IS NOT NULL
```

```

THEN 0
ELSE 1
END AS is_leaf

FROM

working_sets s LEFT OUTER JOIN invitees i ON
(
-- The invitee is in the working set
BITAND(s.working_set, i.invitee_id) = i.invitee_id

-- The invitee will come since some others are coming
AND EXISTS
(
SELECT * FROM rules r
WHERE BITAND(r.they_will_come, i.invitee_id) = i.invitee_id
AND BITAND(s.working_set, r.if_they_come) = r.if_they_come
)
)

-- Terminate the recursion at leaf nodes
WHERE s.is_leaf = 0
)
SEARCH DEPTH FIRST BY working_set SET seq
CYCLE lvl SET cycle TO 1 DEFAULT 0,

-- Eliminate duplicates
candidate_solutions AS
(
SELECT DISTINCT working_set AS candidate_solution
FROM working_sets
-- Only consider leaf nodes
WHERE is_leaf = 1
)

-- List minimal sets only
SELECT s1.candidate_solution AS solution
FROM candidate_solutions s1
WHERE NOT EXISTS
(
SELECT * FROM candidate_solutions s2
WHERE s2.candidate_solution < s1.candidate_solution
AND BITAND(s1.candidate_solution, s2.candidate_solution)
= s2.candidate_solution
);

```

The above solution limits the number of invitees to the number of bits in the numeric data type. The limitation can be overcome using multisets as follows:

```

CREATE TYPE invitees_t AS TABLE OF VARCHAR2(32)
/
CREATE TYPE solutions_t AS TABLE OF invitees_t
/

CREATE TABLE invitees
(
invitee_id INTEGER,
invitee VARCHAR2(32)
);

INSERT INTO invitees VALUES (1, 'A');
INSERT INTO invitees VALUES (2, 'B');
INSERT INTO invitees VALUES (3, 'C');
INSERT INTO invitees VALUES (4, 'D');
INSERT INTO invitees VALUES (5, 'E');
INSERT INTO invitees VALUES (6, 'F');

CREATE TABLE rules
(
rule_id INTEGER,
they_will_come invitees_t,
if_they_come invitees_t
)
NESTED TABLE if_they_come STORE AS if_they_come_nt
NESTED TABLE they_will_come STORE AS they_will_come_nt;

INSERT INTO rules VALUES (1, invitees_t('B', 'C'), invitees_t('A'));
INSERT INTO rules VALUES (2, invitees_t('A', 'D'), invitees_t('C'));
INSERT INTO rules VALUES (3, invitees_t('A', 'B', 'C'), invitees_t('E'));
INSERT INTO rules VALUES (4, invitees_t('C', 'D'), invitees_t('F'));
INSERT INTO rules VALUES (5, invitees_t('B', 'E', 'F'), invitees_t('C', 'D'));
INSERT INTO rules VALUES (6, invitees_t('D'), invitees_t('A', 'B'));

WITH working_sets(lvl, working_set, removed, is_leaf) AS
(

```

```

-- The anchor member of the recursive query
SELECT
1 AS lvl,
-- The complete list of invitees
CAST(MULTISET(SELECT invitee FROM invitees) AS invitees_t)
AS working_set,
NULL AS removed,
0 AS is_leaf
FROM dual

UNION ALL

-- The recursive member of the query
SELECT

lvl + 1 AS lvl,

-- Remove one invitee from the list
CASE WHEN i.invitee IS NULL
THEN s.working_set
ELSE s.working_set MULTISET EXCEPT invitees_t(i.invitee)
END AS working_set,

i.invitee AS removed,

-- Flag the leaf nodes
CASE WHEN i.invitee IS NULL
THEN 1
ELSE 0
END AS is_leaf

FROM

working_sets s LEFT OUTER JOIN invitees i ON
(
-- The invitee is in the working set
invitees_t(i.invitee) SUBMULTISET OF s.working_set

-- The invitee will come since some others are coming
AND EXISTS
(
SELECT * FROM rules r
WHERE invitees_t(i.invitee) SUBMULTISET OF r.they_will_come
AND r.if_they_come SUBMULTISET OF s.working_set
)
)

WHERE s.is_leaf = 0
)
CYCLE lvl SET cycle_flag TO 1 DEFAULT 0,

-- Eliminate duplicates
candidate_solutions AS
(
SELECT column_value AS candidate_solution
FROM
TABLE(SET(CAST(MULTISET(
SELECT working_set FROM working_sets
-- Only consider leaf nodes
WHERE is_leaf = 1
) AS solutions_t)))
)

-- List minimal sets only
SELECT s1.candidate_solution AS solution
FROM candidate_solutions s1
WHERE NOT EXISTS
(
SELECT * FROM candidate_solutions s2
WHERE s2.candidate_solution SUBMULTISET OF s1.candidate_solution
AND s2.candidate_solution != s1.candidate_solution
);

```

Lukasz Pluta becomes the fifth knight of the august order, following in the footsteps of Alberto Dell’Era (Italy), Andre Araujo (Australia), Rob van Wijk (Netherlands), and Ilya Chuhnakov (Russia).

### Credits

The idea for the challenge came from a 1995 paper by C. J. Date titled “*Functional Dependencies are Fun: An Informal Look at the Formal World of FDs.*” ▲



# Words I Don't Use

By Cary Millsap



Cary Millsap

## Methodology

Approximately 100% of the time in the (mostly non-scientific) Oracle world that I live in, when people say “methodology,” they’re using it in the form that the *American Heritage Dictionary* describes as a pretentious substitute for “method.” But methodology is not the same as method. *Methods* are processes. Sequences of steps. *Methodology* is the scientific study of methods.

## Holistic

When people use the word “holistic” in my industry (Oracle), it means that they’re paying attention to not just an individual subcomponent of a system but to a whole system, including (I hope) even the people it serves. But trying to differentiate technology services by saying “we take a holistic view of your system” is about like differentiating myself by saying I’ll wear clothes to work. Saying “holistic” would make it look like I’ve only just recently become aware that optimizing a system’s individual subsystems is not a reliable way to optimize the system itself. *This* should not be a distinctive revelation.

## Best Practice

The “best practice” serves a vital need in any industry. It is the answer to, “Please don’t make me learn about this; just tell me what to *do*.” The “best practice” is a fine idea in spirit, but here’s the thing: many practices labeled “best” don’t deserve the adjective. They’re often containers for bad advice. The most common problem with “best practices” is that they’re not parameterized like they should be. A good practice usually *depends* on something: *if* this is true, *then* do that; *otherwise*, do this other thing. But most “best practices” don’t come with conditions of execution—they often contain no *if* statements at all. They come disguised as recipes that can save you time, but they often encourage you to skip past thinking about things that you really ought to be thinking about.

Most of my objections to “best practices” go away when the practices being prescribed are actually good. But the ones I see are often not, like the old SQL “avoid full-table scans” advice. Enforcing practices like this yields applications that don’t run as well as they should and developers that don’t learn the things they should. Practices like “Measure the efficiency of your SQL at every phase of the software life cycle” are actually “best”-worthy, but alas, they’re less popular because they sound like real work.

## Expert

When I was a young boy, my dad would sometimes drive me to school. It was 17 miles of country roads and two-lane high-

ways, so it gave us time to talk. At least once a year, and always on the first day of school, he would tell me, “Son, there are two answers to every test question. There’s the correct answer, and there’s the answer that the teacher expects. . . . They’re not always the same.” He would continue, “And I expect you to know them *both*.”

He wanted me to make perfect grades, but he expected me to understand my responsibility to know the difference between *authority* and *truth*. My dad thus taught me from a young age to be skeptical of experts.

The word *expert* always warns me of a potentially dangerous type of thinking. The word is used to confer authority upon the person it describes. But it’s *ideas* that are right or wrong, not *people*. You should evaluate an idea on its own merit, not on the merits of the person who conveys it. For every expert, there is an equal and opposite expert—but for every fact, there is not necessarily an equal and opposite fact. A big problem with *expert* is corruption—when self-congratulators hijack the label to confer authority upon themselves. But, of course, misusing the word erodes the word. After too much abuse within a community, *expert* makes sense only with finger quotes. It becomes a word that critical thinkers use only ironically, to describe people they want to avoid.

## Wait

In 1991, Oracle Corporation released some of the most important software instrumentation of all time: the “wait” statistics that were implemented in Oracle 7.0. Here’s part of the story, in Juan Loaiza’s words, as told in Nørgaard *et al.* (2004), *Oracle Insights: Tales of the Oak Table*.

*“This stuff was developed because we were running a benchmark that we could not get to perform. We had spent several weeks trying to figure out what was happening with no success. The symptoms were clear—the system was mostly idle—we just couldn’t figure out why. We looked at the statistics and ratios and kept coming up with theories, the trouble was that none of them were right. So we wasted weeks tuning and fixing things that were not the problem. Finally we ran out of ideas and were forced to go back and instrument the code to figure out what the problem was. Once the waits were instrumented the problem was diagnosed in minutes. We were having ‘free buffer’ waits because the DBWR was not writing blocks fast enough. It’s amazing how hard that was to figure out with statistics, and how easy it was to figure out once the waits were instrumented. . . . In retrospect a lot of the names could be greatly improved. The wait interface was added after the freeze date as a ‘stealth’ project so it did not get as well thought through as it should have. Like I said, we were just trying to solve a problem in the course of a benchmark. The trouble is that so many people use this stuff now that if you change the names it will break all sorts of tuning tools, so we have to leave them alone.”*

Before Juan’s team added this code, the Oracle kernel would show you only how much time its *user* calls (like *parse*, *exec*, and *fetch*) were taking. The new instrumentation, which included a set of new fixed views like *v\$session\_wait* and new WAIT lines in our trace files, showed how much time Oracle’s *system* calls (like reads, writes, and semops) were taking. The wait interface begat a whole new mental model about Oracle performance, based on the principle of *working* versus *waiting*:

$$\text{Response Time} = \text{Service Time} + \text{Wait Time}$$

In this formula, Oracle defines *service time* as the duration of the CPU used by your Oracle session (the duration Oracle spent *working*), and *wait time* as the sum of the durations of your Oracle wait events (the duration that Oracle spent *waiting*). Of course, *response time* in this formula means the duration spent inside the Oracle Database kernel.

There are two reasons I don’t use the word *wait*. The first is simply that it’s ambiguous. The Oracle formula is okay for talking about database time, but the scope of my attention is almost never just Oracle’s response time—I’m interested in the *business’s* response time. And when you think about the whole stack (which, of course you do; see holistic), there are events we could call *wait events* all the way up and down:

- The customer *waits* for an answer from a user.
- The user *waits* for a screen from the browser.
- The browser *waits* for an HTML page from the application server.
- The application server *waits* for a database call from the Oracle kernel.
- The Oracle kernel *waits* for a system call from the operating system.
- The operating system’s I/O request *waits* to clear the device’s queue before receiving service.
- ...

If I say *waits*, the users in the room will think I’m talking about application response time, the Oracle people will think I’m talking about Oracle system calls, and the hardware people will think I’m talking about device queueing delays. Even when I’m not. There is a deeper problem with *wait* than just ambiguity, though. The word *wait* invites a mental model that *actually obscures your thinking about performance*. Here’s the problem: *waiting* sounds like something you’d want to avoid, and *working* sounds like something you’d want more of. Your program is *waiting*?! Unacceptable. You want it to be *working*. The connotations of the words *working* and *waiting* are unavoidable. It sounds like, if a program is waiting a lot, then you need to fix it; if it’s working a lot, then it’s probably okay. Right? Actually, no.

The connotations “work is virtuous” and “waits are abhorrent” are *false* connotations in Oracle. One is not inherently better or worse than the other. *Working* and *waiting* are not accurate value judgments about Oracle software. On the contrary, they’re not even *meaningful*; they’re just arbitrary labels. We could just as well have been taught to say that an Oracle program is “working on disk I/O” and “waiting to finish its CPU instructions.” The terms *working* and *waiting* really just refer to different subroutine call types:

“Oracle is <i>working</i> ”	means	“your Oracle kernel process is executing a <i>user</i> call”
-----------------------------	-------	--

“Oracle is <i>waiting</i> ”	means	“your Oracle kernel process is executing a <i>system</i> call”
-----------------------------	-------	--

The working-waiting model implies a distinction that does not exist, because these two call types have equal footing. One is no worse than the other, except by virtue of how much time it consumes. *It doesn’t matter whether a program is working or waiting; it only matters how long it takes.*

The working-waiting paradigm is a flawed analogy. I’ll illustrate. Imagine two programs that consume 100 seconds apiece when you run them:

Program A		Program B	
Duration	Call Type	Duration	Call Type
98	system calls (waiting)	90	user calls (working)
2	user calls (working)	2	system calls (waiting)
<b>100</b>	<b>Total</b>	<b>100</b>	<b>Total</b>

To improve program A, you should seek to eliminate unnecessary system calls, because that’s where most of A’s time has gone. To improve B, you should seek to eliminate unnecessary user calls, because that’s where most of B’s time has gone. That’s it. Your diagnostic priority shouldn’t be based on your calls’ names; it should be based solely on your calls’ contributions to total duration. *Specifically, conclusions like, “Program B is okay because it doesn’t spend much time waiting,” are false.* I find that discarding the working-waiting model helps people optimize better. Here’s how you can do it. First, understand the substitute phrasing: *working* means executing a *user* call; and *waiting* means executing a *system* call. Second, understand that the excellent ideas people use to optimize other software are excellent ideas for optimizing Oracle, too:

1. Any program’s duration is a function of all of its subroutine call durations (both user calls and system calls), and
2. A program is running as fast as possible only when (1) its unnecessary calls have been eliminated, and (2) its necessary calls are running at hardware speed.

Oracle’s wait interface is vital because it helps us measure an Oracle program’s *complete* execution duration—not just Oracle’s *user* calls, but its *system* calls as well. But I avoid saying *wait* to help people steer clear of the incorrect bias introduced by the working-waiting analogy. ▲

*Cary Millsap is an entrepreneur, leader, teacher, software technology advisor, software designer and developer, writer, and Oracle software performance specialist. He wrote the book Optimizing Oracle Performance (2003), for which he and co-author Jeff Holt were named Oracle Magazine’s 2004 Authors of the Year, and the book The Method R Guide to Mastering Oracle Trace Data (2011, 2016). His software company, Method R Corporation, sells easy-to-use, high-precision time measurement software for software development, code reviews, performance tests, concept proofs, hardware and software evaluations, upgrades, troubleshooting, and more, for Oracle developers, DBAs, and decision-makers in every phase of the software life cycle. He is VP of User Experience Services and Solutions at Cintra Software and Services, responsible for integrating Method R software and capabilities into Cintra practices worldwide and creating new performance-focused software and services.*

Copyright © 2019, Cary Millsap



# NoCOUG Conference #130

Post-Conference Reception Hosted by Quest





# NoCOUG Conference #130

Post-Conference Reception Hosted by Quest



**NoCOUG**

P.O. Box 3282

Danville, CA 94526

RETURN SERVICE REQUESTED

memSQL 

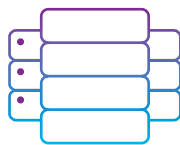
# The No-Limits Database™

The cloud-native, operational database  
built for speed and scale



## SPEED

Accelerate time to insight  
with a database built for  
ultra fast ingest and  
high performance query



## SCALE

Build on a cloud-native  
data platform designed for  
today's most demanding  
applications and  
analytical systems



## SQL

Get the familiarity & ease  
of integration of a traditional  
RDBMS and SQL, but with  
a groundbreaking,  
modern architecture