Creating a Culture for Curiosity, Creativity, and Critical Thinking

Radia Perlman
Dell Technologies

Radia.Perlman@dell.com

Story behind this talk

- One topic I was told people would be interested in was
  - "Building careers in the infosec space, especially for groups that are not well represented today"
- I said "I could do a talk like that, but I want to also do a technical talk"
  - Which I'll do tomorrow
- This talk is a combination of
  - Ranting about misconceptions caused by hype
  - Story of my career, and lessons learned along the way
  - Tips for creating a positive culture

Critical Thinking

- Not everything you read, or hear is true

How computer networking tends to be taught

- Memorize these standards documents, or the arcane details of some implementation that got deployed

- Nothing else ever existed

- Except possibly to make vague, nontechnical, snide comments about other stuff

My philosophy on teaching (and books)

- Look at each conceptual problem, like how to autoconfigure an address

- Talk about a bunch of approaches to that, with tradeoffs

- Then mention how various protocols (e.g., IPv4, IPv6, Appletalk, IPX, DECnet, …) solve it

But some professors say…

- Why is there stuff in here that my students don't "need to know"?

Things are so confusing

- Comparing technology A vs B
  - Nobody knows both of them
  - Both A and B are moving targets

Standards Bodies…

What about "facts"?

- What if you measure A vs B?

What about "facts"?

- What if you measure A vs B?
- What are you actually measuring?...one implementation of A vs one implementation of B

What about "facts"?

- What if you measure A vs B?
- What are you actually measuring?...one implementation of A vs one implementation of B
- ***So don't believe something unless you can figure out a plausible property of the two protocols that would make that true***

Example of confusion and hype: Blockchain
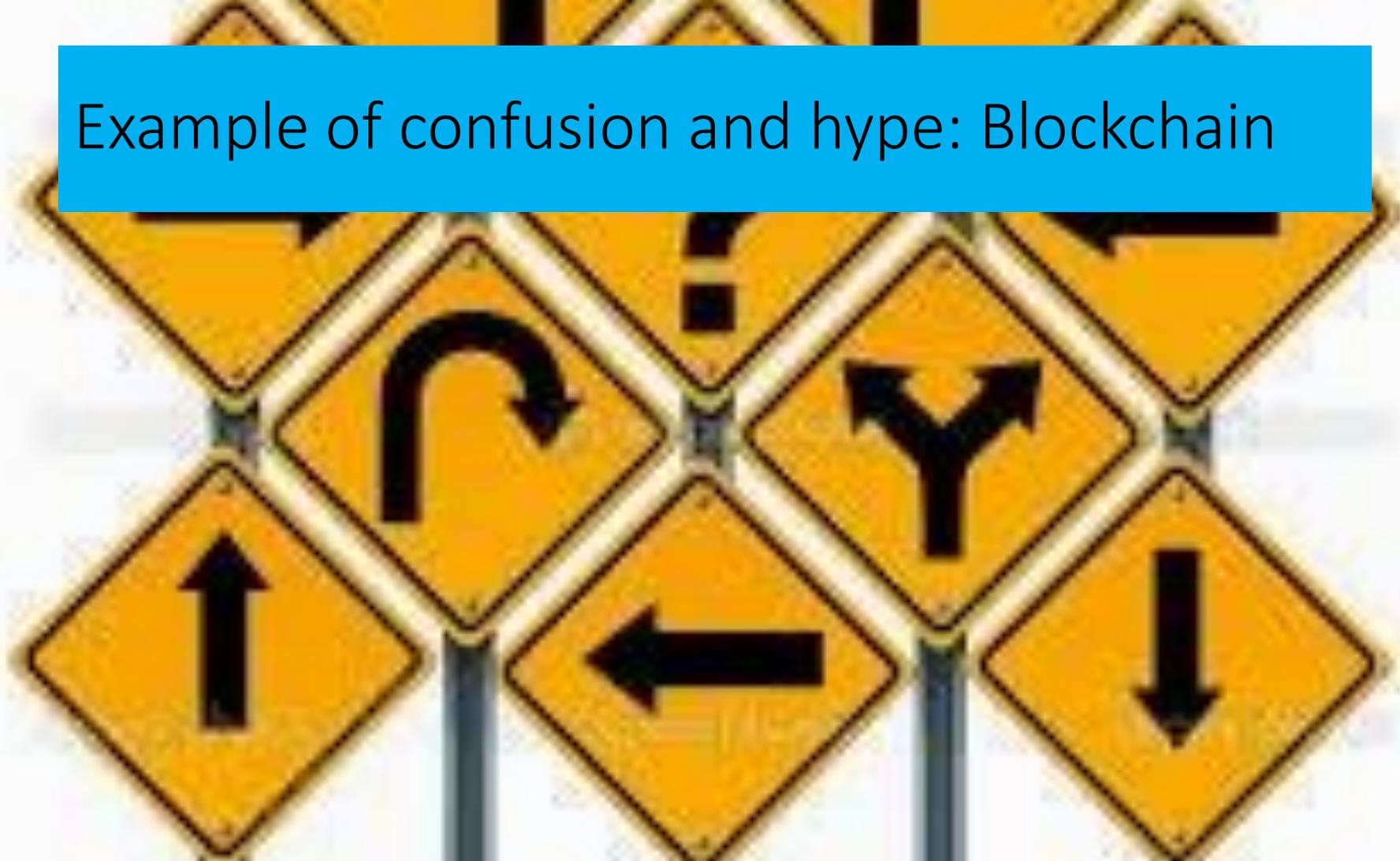
What is blockchain?

- Started as the technology behind Bitcoin

- People made money on Bitcoin

- The more hype, the more money gets poured in

- And with hype, "blockchain" becomes powerful for attracting venture capital

- So, people assume the technology must be important

Hype

- Articles about how "blockchain" is:
  - Biggest advancement in technology since the Internet
  - "Being considered for" all sorts of problems
  - "Even the US Military is looking at blockchain technology to secure nuclear weapons"

Possible questions

- I claim these are wrong questions
  - What kinds of applications can I build on blockchain?
  - How can I apply blockchain to this problem?
- Right approach
  - What problem am I solving?
  - What are various ways of solving it
  - Compare approaches
  - If 'blockchain' turns out to be the best solution, OK…

Misleading statements

- Look how many applications I can build using blockchain!
  - if you assume 'blockchain' is a black box with API "store data" and "retrieve data", there are lots of applications you could build
  - But these applications could as easily (and more efficiently) have used a disk, or a cloud, or a storage array
- Societal problem, e.g., e-coli…solution…put supply chain history on blockchain!
  - Problems: doesn't answer any of the hard problems, like how the farmers get the credentials to write data, how to know which lettuce came from where…
  - And simply using a database would do the same thing (more efficiently)
- Look how much faster this application is once we moved it to blockchain
  - Problem: it used to be done with hand-carrying documents up mountains to be notarized by … all they're saying is if only they had computers and networks they could do stuff faster
- Governments can't regulate or outlaw cryptocurrency
  - Yes they can…maybe if you're careful you won't get caught
  - Just like with murder

People hear

- "Centralized" = "Bad"
- "Distributed" = "Good"
- "Distributed" means "blockchain"
- People not even sure what "distributed" means

Is centralized "bad"?

- Centralized is the most efficient

- And it's clear who to blame

- Most applications require "adult supervision"

- So, most of the time, centralized is <u>exactly</u> what is needed

- For instance, when you withdraw $20 from an ATM, your own bank makes the decision, not a consortium of banks voting

- And "centralized" can still have lots of servers so the application is always up

- And good storage systems (like public clouds) store data in multiple places, geo-replicated

What does "distributed" mean?

- Store data in lots of places? (industry knew how to do that before "blockchain")

- Have multiple instances of a server, to split load, and for resiliency? (industry knew how to do that before blockchain)

- Distributed trust…that's interesting and subtle

Distributed Trust

- One organization can become evil (evil employee, someone evil steals their private key and impersonates them)

- A single organization might need to prove something, other than by saying "trust me"

- Does "blockchain" solve that?
  - Not really…51% attack on public blockchain, completely trust node you query, no enforcement against a node refusing to record a transaction

- Are there other ways of protecting against a malicious participant? (next slide)

Example: Credit Rating Agencies

- Current method:
  - Several totally independent organizations
  - Each with their own sources of data, and algorithms for computing a score
  - Someone checking a credit rating can go to whichever they trust, or multiple and compare
  - Plus laws that allow consumers to contest data they think is wrong
- With blockchain, presumably
  - Some consortium…who chooses the participants?
  - They'd have to argue about things, vote, etc., and it would be far less efficient

More Hype

Example proposed application

- "Could blockchain be the answer to healthcare?"
  - "Imagine this: Your entire medical record is on the blockchain. Monitoring systems and IoT devices automatically update your data, so when you go for diagnostic tests, the results are recorded without a third party"

I can't imagine an application *less* suitable for blockchain

- World readable database?

- World writeable database? (who organizes credentials so you know data being written is traceable to the sensor that wrote it?)

- No organization to the data other than append-only log, mixing every human's records and sensor readings?

- Data kept forever, in its entirety

I can't imagine an application *less* suitable for blockchain

- And yeah, I'm sure someone will think of using encryption to get around the world-readable thing
  - But with what keys?
  - Do you use trusted third parties?
    - Then why not have them manage the database?

My advice for designs

- Start with "what problem am I solving"
- Then consider several types of solutions, and compare
- Don't say "can we use this technology for this application"?
- What I say to engineers who are being pressured to use "blockchain"
  - Do the right technical solution
  - And then call it "blockchain"

Some stories from my past

"What do you want to be when you grow up?"

- I certainly would not have said "designing computer network protocols"

"What do you want to be when you grow up?"

- I certainly would not have said "designing computer network protocols"

- It was just a bunch of random events that led me here

"What do you want to be when you grow up?"

- I certainly would not have said "designing computer network protocols"

- It was just a bunch of random events that led me here

- And I love how it turned out!

Lesson

- Don't worry about finding "the thing you were born to do"
- Making decisions:
  - You'll never have enough information to make an informed decision
  - There are two types of people:
    - Those that would be happy with either decision
    - Those that would be unhappy with either decision

School before college

- Loved logic problems
  - And writing
  - And music
  - And humor

- I certainly never took anything apart

- I knew nothing about computers
  - If I'd known more, I probably would have said "I'd be interested in anything….except computers"

Frankly…

- I think this industry could use more people that hate computers
- Engineers should actually meet some humans, then they'd stop having programs ask questions like:
  - Do you want POP or IMAP?
  - Do you want to display both the secure and insecure items?

My first computer class

- In high school, a teacher noticed there was a programming class at a nearby university
  - She could drive us there after school
  - Aren't teachers awesome!!!

Then…my first computer class

- After years of always being the top student in any math/science class, I walk into this computer programming class…
  - Totally intimidated
    - by students saying they'd built "ham radios" when they were 7
    - And asking questions with fancy words like "input"
- I wound up learning nothing in that class
- But it made me a much better teacher

Lesson

- Most of success at learning is self-confidence

How I learned to program

- In college, took a physics class
  - TA (teaching assistant) to me:
    - I need a programmer for a project. Would you like to be my programmer?
  - Me: "I don't know how to program"
  - TA: "I know. That's why I'm asking you. I have no money to pay you. You're obviously smart, and I'm sure you can learn. If you knew how to program, you'd expect to get paid."

Why I agreed

- I had a friend who knew how to program
- In that environment, I had no trouble learning…

Continuing with grad school

- Did everything but my thesis
- My perception…
- Never found an advisor
- Then an old friend asked me if I was enjoying grad school
- I joined his group, at BBN, designing routing for ARPA packet radio network
- How I got the job at Digital
- Right place at the right time

At Digital

- Routing Protocol innovations (and specifically, IS-IS)
    - Self-stabilization
    - Scalability
    - Manageability
- Spanning tree bridging (which is what "Ethernet" is today)

Continuing with my career path

- For very weird reasons, I went back to grad school and did get a PhD (in computer science, rather than math)
- So much better the 2nd time!
  - I'd learned "secrets"
    - When professor is droning on, ask questions!
    - Start problem sets as soon as you get them
    - If you've done the homework and followed the class, you'll do fine on tests
- Despite everything I'd done, the way I actually got known….writing my book "Interconnections: Bridges, Routers, Switches, and Internetworking Protocols"
- Later co-wrote "Network Security: Private Communication in a Public World"

Now on to culture…

- How can we improve things to attract, retain, and ensure XXX type people (e.g., women, minorities, etc.) thrive?

- WRONG QUESTION!!

- Right question: "How can we improve things to attract, retain, and ensure all good people thrive."

Culture

- It must be safe to ask questions
- Example of the opposite…
- How to answer a naïve question…
- But better yet, be a role model

Don't reward bullies!

- Relentlessly self-promoting bullies tend to do well for themselves, in terms of recognition, titles, etc.
- Even if they were brilliant, they are toxic to those around them
- However… I've never met anyone like that, that was actually good technically
- Note: This is not a gender thing

Lesson

- It must be easy to learn things

Buzzwords

- For example, SDN, AI, object-oriented, hyper-converged…
- Useful for impressing people
- E.g., "our company utilizes AI-enhanced blockchain technology"
- Buzzwords don't necessarily mean anything.
- They traumatize most engineers
- People who are sure they know the definition disagree, so…
- Be _way_ more specific when talking, or when thinking about a problem

Cross-fertilization

Cross-fertilization

- Learning about other groups, other technology

- Getting other points of view on your technology

- How to make this happen in an industry of introverts?

Ideas

- At lunch, sit with people you don't know
- Talk to people while you're waiting in lines
- Maybe set aside tables specifically NOT for friends to chat with each other
- Maybe have topics chosen for such tables, or a moderator getting everyone involved and meeting each other
- Brainstorming sessions ("half-baked talks")

Mentoring

- Strict hierarchy bothers me
- I think everyone should be mentoring everyone, when appropriate
- Pass on wisdom to next generation….
- But even better…ask them for help

Growing Leaders

- There are some people that will volunteer for leading anything, and take credit for any work the group does

- And there are those that would never think to volunteer

- Look for those, and encourage them to volunteer

- If they express nervousness, assure them others will help (and find people that will actually help them)

- Have a culture where helping others (without trying to take over, or worrying about getting the credit), is valued

Hints for problem-solving and communication

- Someone once asked me for advice on how to be a good speaker

Hints for problem-solving and communication

- Someone once asked me for advice on how to be a good speaker

- My advice
  - Given a topic you know something about
  - A level of the audience
  - An amount of time

- Come up with an appropriate talk

Suggestion

- People could be encouraged to practice this technique, e.g.,
  - Talks of various lengths (5, 10, 15, 30 minutes) on topics such as:
    - What's a battery? What might cause it to catch fire?
    - What's public key cryptography?
    - What is garbage collection?
    - What is agile computing?
    - What is ML?
  - Get constructive feedback
  - Then give these talks to a fairly large audience (once people are fairly good at doing this)

Problem-solving

- What problem are you solving?

- What is the best problem to solve?

- Remove all irrelevant details

- And yes…other things you have to worry about:
  - What implementation things you can leverage
  - Time to market

Now some woman-y thoughts

Why are there so few technical women?

- The more senior, the fewer there are
- It doesn't seem to be changing
- I'm so used to being the only woman in the room I don't even notice
- I asked some senior women who switched to the management track why they did that
  - So many things they could have said
  - Every one of them said "I wasn't smart enough to stay on the technical track"
- Women think they wouldn't like it
- Women think they wouldn't be good at it
- Inner circle: networking at golf, old friends, social compatibility
- Subconscious impressions
- Perception today: women (and minorities) get special treatment

A recruitment letter I got

- "We are particularly interested in you as a female thought leader"
- My fantasy reply (I didn't reply because I wasn't interested in the job)
- "Thank you for your interest in me as a 'female thought leader'. Although my credentials as a thought leader are impeccable, I must warn you I am not that qualified as a female. I cannot walk in heels, I have no clothing sense, and I'm not particularly decorative. What aspects of female are important for this job?"

Visible things that don't actually help

- "Take your daughter to work day"
- I'm so glad they changed it to "take your children to work day"

How to feel like a group

- Sometimes diversity makes it hard…
  - Languages
  - Interests
  - Group activities

What is "diversity"

- Not just skin color and body shapes
- But different skills, different life experiences, different background
- A group should leverage each person's strengths

Problems

- The Dilbert cartoon about "what is an engineer" is actually corrosive

- Hiring managers that can only respect someone who reminds them of a younger version of themselves

- Events that make it seem like a particular minority is getting preferential treatment creating resentment, reinforcing assumptions

School vs Life

- In school, emphasis on "must be your own work"
- In industry, the point is to get the job done
  - Great to ask for, and give help
- In academia, "novelty" is important
- In industry, build on (good) previous ideas

Fostering and Practicing Critical Thinking

- A lot of what "everyone knows" is not actually true

- Advice to professors:
  - Don't just have students read good papers
  - Have them read horrible papers in prestigious venues
  - Have them read excellent, ground-breaking papers
    - Along with the reviews from the first 4 times the paper was rejected

Lessons

- It must be safe to ask questions
- Leaders should ask questions
- Life isn't fair, but we can try
  - Don't mistake bullies for geniuses
  - Don't reward bullies
- If you believe you can't do something, you won't
- The most effective form of mentoring is to ask others for help
- Know what problem you're solving before you try to solve it

Thank you!