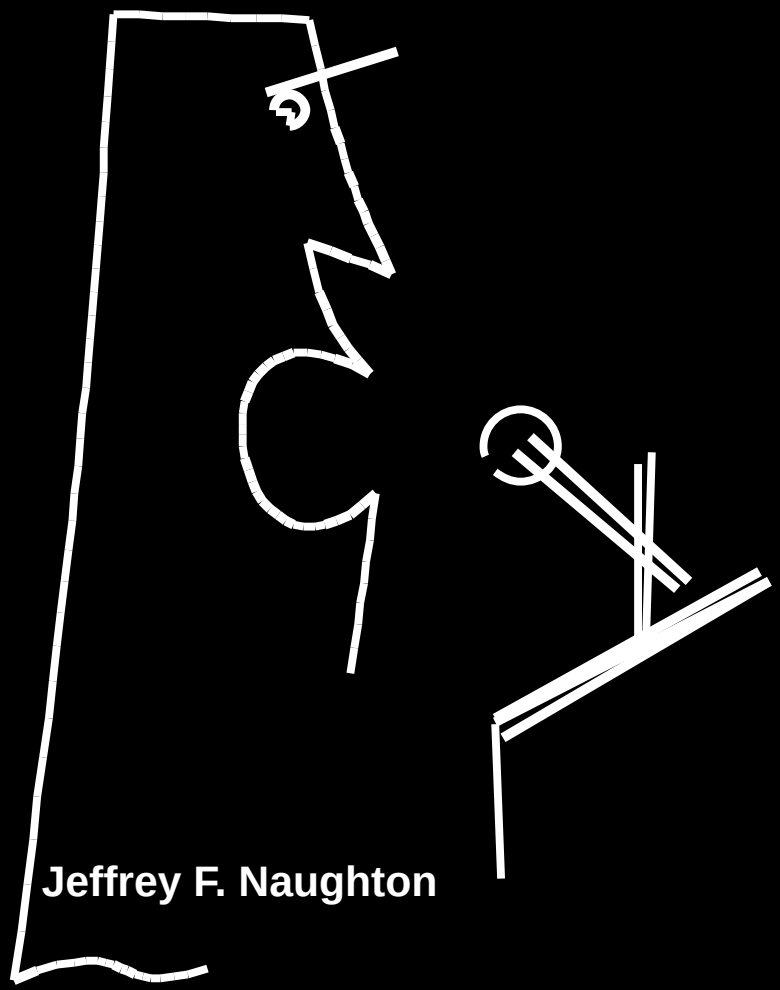



# DBMS Research: First 50 Years, Next 50 Years

Jeffrey F. Naughton

# Reactions to This Keynote From Colleagues…


*Assistant Professor*


*"Cool, I look forward to it."*

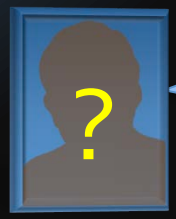
*Associate Professor #1*


*(upon hearing what I proposed to talk about):*
*"But how are you going to make that interesting?"*

*Associate Professor #2*


*"Well, you have reached the age where you can scratch your butt in public, go ahead."*

*Emeritus Professor*




*"Don't do it. Giving a keynote means you are a washed-up has-been."*

# My Goals…

Talk about why I think DBMS research has been so rewarding over the past 50 years

Some personal anecdotes from a "washed-up has-been" about fun work that would be discouraged today

Discuss why the next 50 years are at risk and what we can do about it

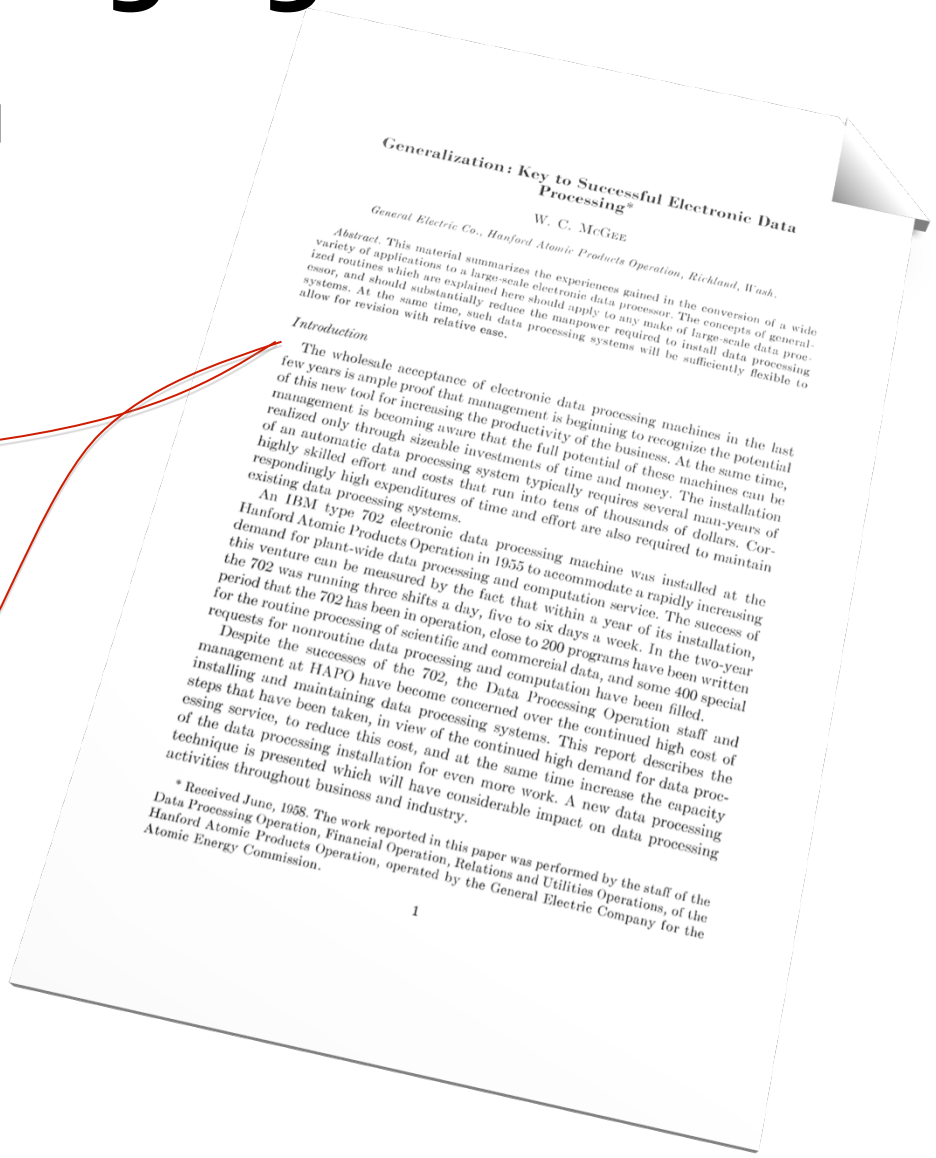
Avoid scratching


Make this "interesting"

# Long, Long Ago…

William McGee published a J. ACM paper "Generalization: Key to Successful Electronic Data Processing."

[Jan. '59]



Generalization: Key to Successful Electronic Data Processing*

W. C. McGee

General Electric Co., Hanford Atomic Products Operation, Richland, Wash.

Abstract. This material summarizes the experiences gained in the conversion of a wide variety of applications to a large-scale electronic data processor. The concepts of generalized routines which are explained here should apply to any make of large-scale data processor, and should substantially reduce the manpower required to install data processing systems. At the same time, such data processing systems will be sufficiently flexible to allow for revision with relative ease.

Introduction

The wholesale acceptance of electronic data processing machines in the last few years is ample proof that management is beginning to recognize the potential of this new tool for increasing the productivity of the business. At the same time, management is becoming aware that the full potential of these machines can be realized only through sizeable investments of time and money. The installation of an automatic data processing system typically requires several man-years of highly skilled effort and costs that run into tens of thousands of dollars. Correspondingly high expenditures of time and effort are also required to maintain existing data processing systems.

An IBM type 702 electronic data processing machine was installed at the Hanford Atomic Products Operation in 1955 to accommodate a rapidly increasing demand for plant-wide data processing and computation service. The success of this venture can be measured by the fact that within a year of its installation, the 702 was running three shifts a day, five to six days a week. In the two-year period that the 702 has been in operation, close to 200 programs have been written for the routine processing of scientific and commercial data, and some 400 special requests for nonroutine data processing and computation have been filled.

Despite the successes of the 702, the Data Processing Operation staff and management at HAPO have become concerned over the continued high cost of installing and maintaining data processing systems. This report describes the steps that have been taken, in view of the continued high demand for data processing service, to reduce this cost, and at the same time increase the capacity of the data processing installation for even more work. A new data processing technique is presented which will have considerable impact on data processing activities throughout business and industry.

* Received June, 1958. The work reported in this paper was performed by the staff of the Data Processing Operation, Financial Operation, Relations and Utilities Operations, of the Hanford Atomic Products Operation, operated by the General Electric Company for the Atomic Energy Commission.

1

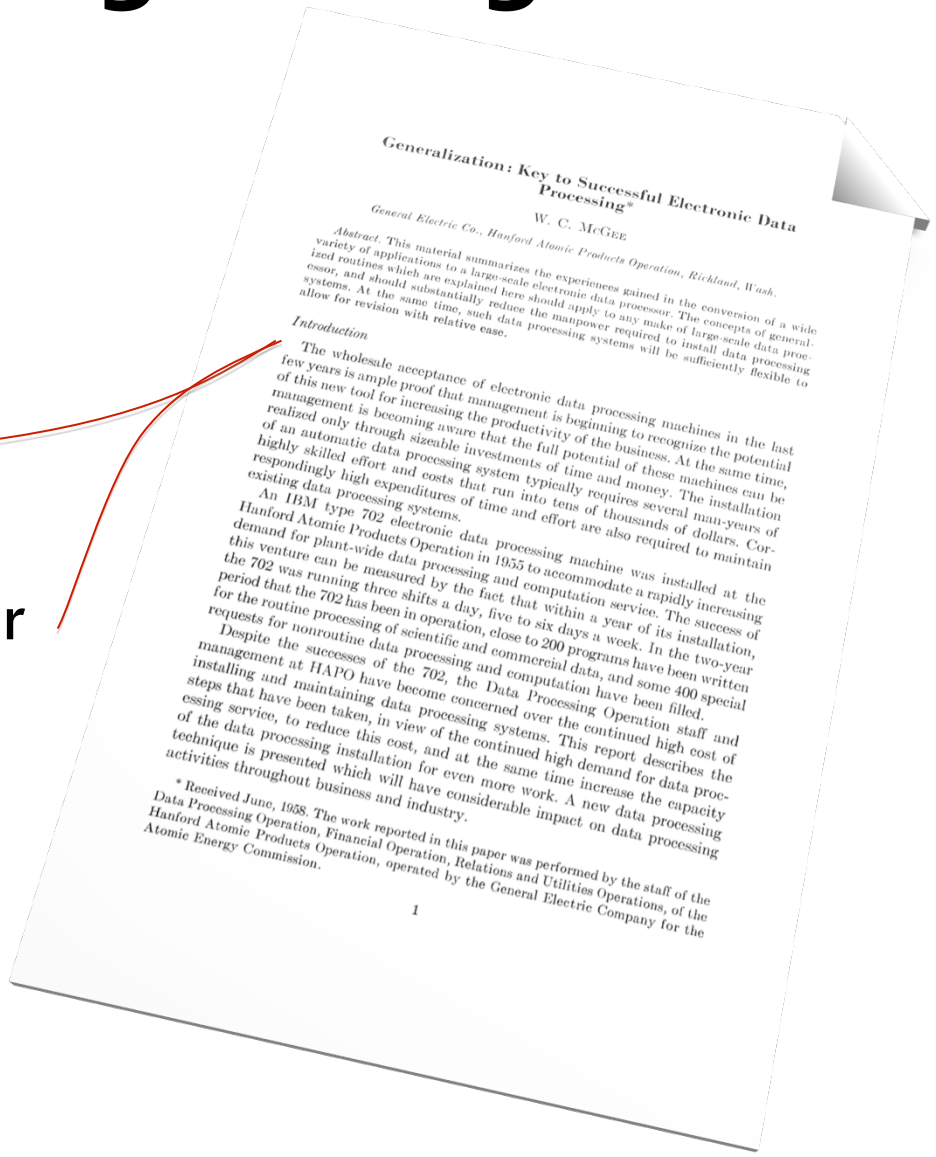
## Great first sentence:

"The wholesale acceptance of electronic data processing machines […] is ample proof that management is beginning to recognize the potential of this new tool."

# The More Things Change…

Another sentence from McGee's intro:

"management is increasingly concerned over the continued high cost of installing and maintaining data processing systems."



**Generalization: Key to Successful Electronic Data Processing***

W. C. McGee

*General Electric Co., Hanford Atomic Products Operation, Richland, Wash.*

*Abstract.* This material summarizes the experiences gained in the conversion of a wide variety of applications to a large-scale electronic data processor. The concepts of generalized routines which are explained here should apply to any make of large-scale data processor, and should substantially reduce the manpower required to install data processing systems. At the same time, such data processing systems will be sufficiently flexible to allow for revision with relative ease.

*Introduction*

The wholesale acceptance of electronic data processing machines in the last few years is ample proof that management is beginning to recognize the potential of this new tool for increasing the productivity of the business. At the same time, management is becoming aware that the full potential of these machines can be realized only through sizeable investments of time and money. The installation of an automatic data processing system typically requires several man-years of highly skilled effort and costs that run into tens of thousands of dollars. Correspondingly high expenditures of time and effort are also required to maintain existing data processing systems.

An IBM type 702 electronic data processing machine was installed at the Hanford Atomic Products Operation in 1955 to accommodate a rapidly increasing demand for plant-wide data processing and computation service. The success of this venture can be measured by the fact that within a year of its installation, the 702 was running three shifts a day, five to six days a week. In the two-year period that the 702 has been in operation, close to 200 programs have been written for the routine processing of scientific and commercial data, and some 400 special requests for nonroutine data processing and computation have been filled.

Despite the successes of the 702, the Data Processing Operation staff and management at HAPO have become concerned over the continued high cost of installing and maintaining data processing systems. This report describes the steps that have been taken, in view of the continued high demand for data processing service, to reduce this cost, and at the same time increase the capacity of the data processing installation for even more work. A new data processing technique is presented which will have considerable impact on data processing activities throughout business and industry.

* Received June, 1958. The work reported in this paper was performed by the staff of the Data Processing Operation, Financial Operation, Relations and Utilities Operations, of the Hanford Atomic Products Operation, operated by the General Electric Company for the Atomic Energy Commission.

# Context: McGee's Machine

- IBM 702

# More On The 702

- **Memory** (cathode ray tubes): 20K bytes.
- **Disk** (magnetic drum): 60K bytes.
- **CPU:** 10MIPH
  - MIPH = Million Instructions Per Hour
  - For those of you without calculators, that is about 0.000278 MIPS
  - Or, if you like, a 0.000000278 GHz processor.

# A Note About This Early History…

# Problem: File-Dependent Programming

- Write a sort program for the payroll file
- Write another sort program for the accounts receivable file
- No sharing
  - between files
  - between applications
  - between institutions


**Solution**: "generalized programming"

# What Do You Need for Generalized Programming?

- A description of the record layouts in the file
- Ideally, in some place should also capture elements in common to multiple files

**Schemas**

- Programs that
  - Interpret these descriptions and
  - Make it easy to express generally useful operations (sorting, reporting) on the files

**Data Manipulation Languages**

# Another Problem: Updates

- How do you undo mistakes?
- How do you handle failures in middle of processing?


**Solution**: in a separate file, write before/after images of updated records. (Logging)

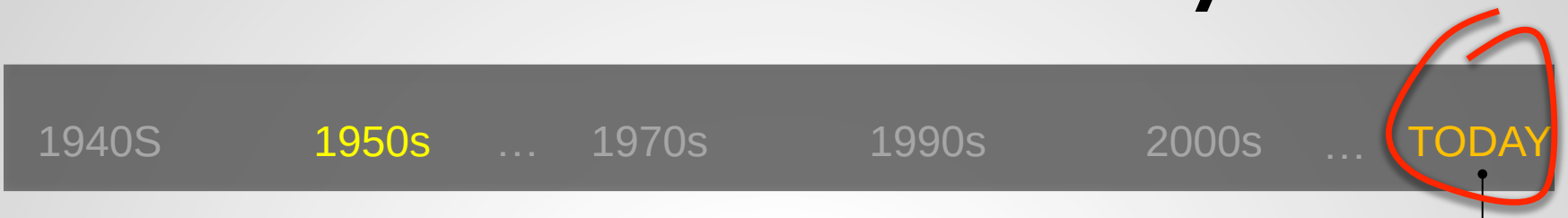
# So What's the Point?


The seeds of a rich and vibrant database management research community were planted

In particular,

- Strong commercial interest with $$$ behind it
- Common non-trivial technical challenges related to data management
- A set of problems amenable to abstraction and generalization

# Fast Forward to Today…




- These three key factors
  - commercial interest
  - common data management challenges
  - attractive problems

  are present now more than ever

**That is the good news!**

# What About the Research Community?

- Maybe it is in good shape
  - *But I am not so sure*
- Maybe it is headed in a good direction
  - *But I am less sure*
- We worry a lot about what to work on
  - *I am not worried about this*
- We are increasingly valuing and rewarding the wrong things
  - *I am worried about this*


# The Problem I Worry About

The combination of:

- Pressure to publish lots of papers +
- Low acceptance rates +
- Bad reviewing

is sucking the air out of our community

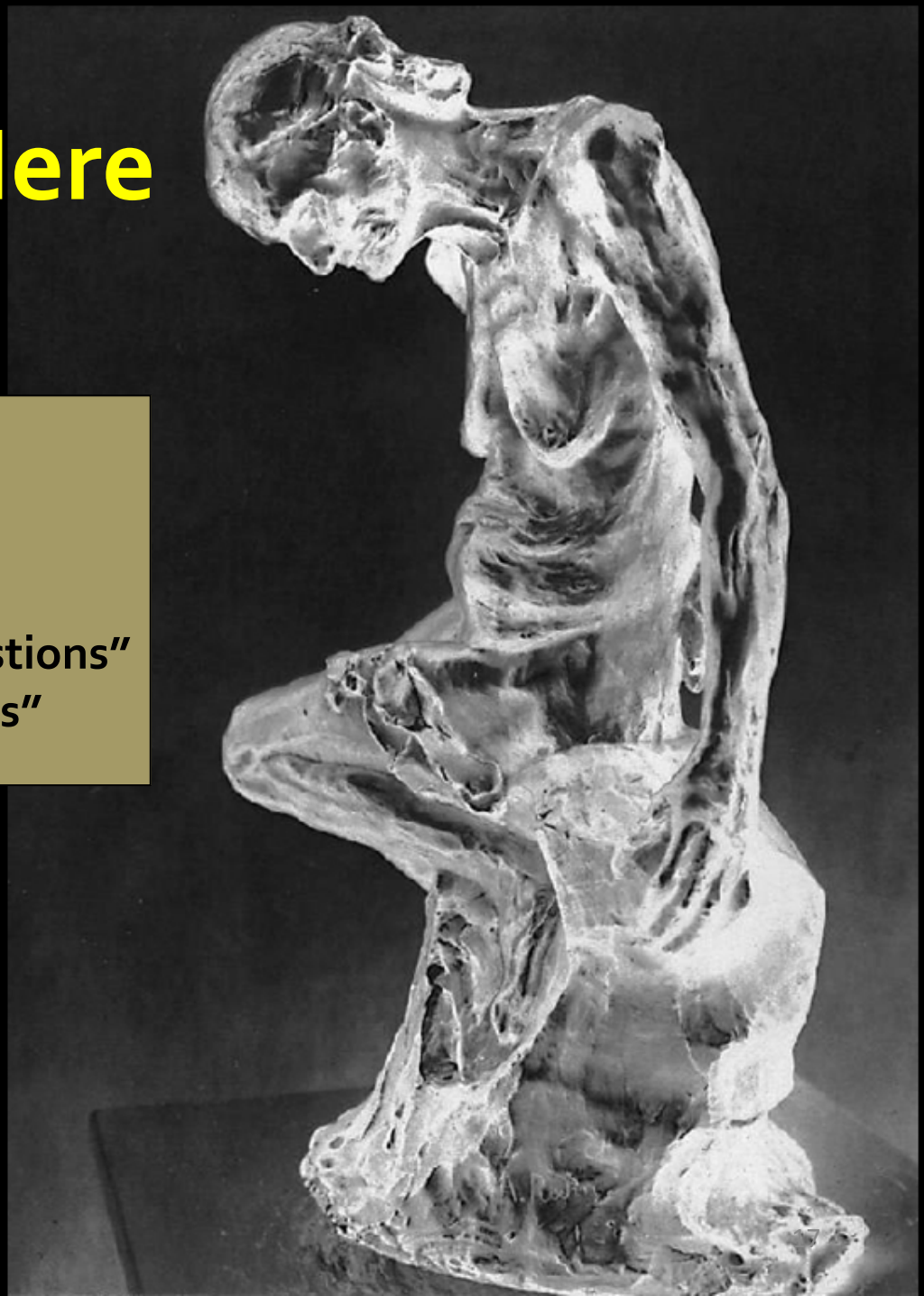
# Doomsday Scenario

**Being a database researcher means a life filled with all the joy, meaning, and creativity of studying for and taking college entrance exams**

# Doomsday is Here

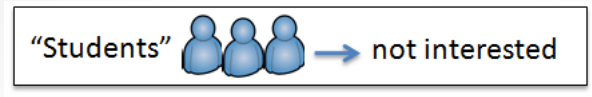
- We behave as if
  - Researchers are "students"
  - PC members are "graders"
  - Publications pose "exam questions"
  - New submissions are "answers"

- These "exams" are perceived as the only path to success in the community

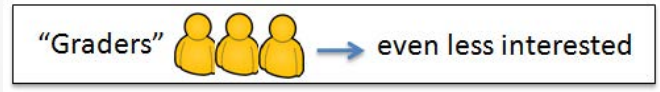


# Some More on the Analogy…

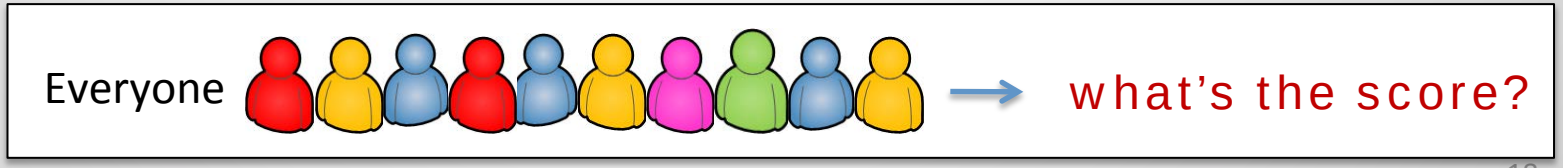
- The **"students"** are not that interested in the questions




- The **"graders"** are even less interested in the answers



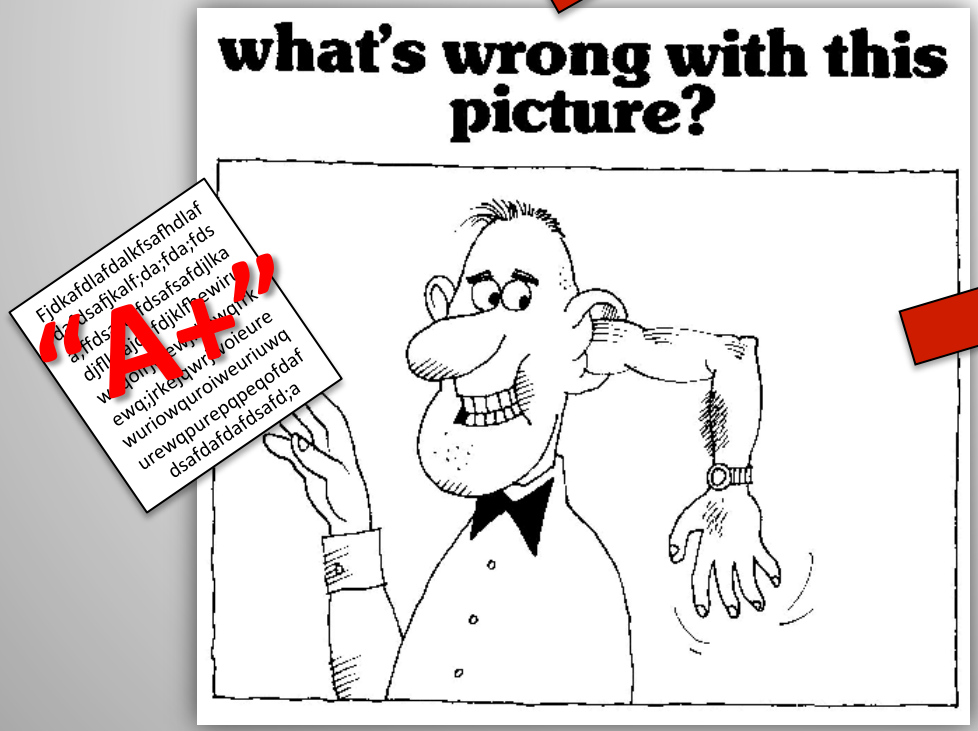

- **No one else** is interested in either, caring only about the scores



# So What is Wrong With This?

Who wants to spend their life taking meaningless exams?

What kind of people will this scenario attract to our field?




What kind of work will be common in this environment?

# My Take…

**The problem isn't**

**"Researchers today! They need to be more like the researchers in the good old days!"**

**Rather, it is more**

**"we need to re-think the environment we have created and what it encourages"**

More on this later, first some "old war stories"


…about work I found rewarding and fun

# War Story #1: The Sorting Record

1940S 1950s … 1970s 1980s 2000s … TODAY

DeWitt had a parallel DBMS project (***Gamma***)

- Very fortunately for me, he let me join
- One of the things we worked on:
  - **parallel sorting on an Intel iPSC-2 Hypercube**
  - (Donovan Schneider was key to this project.)

- **Our main idea:**
  - Use parallel sampling to probabilistically pick range partitioning cutoffs
  - Repartition the data
  - Sort locally

# In Those Days…

- It still made (a little) sense to try the original sort benchmark:
  - 1M 100 byte records (now perhaps known as wristwatch sort)
- I think our time was something like 30 seconds
- We mentioned it to Jim Gray when he visited Wisconsin.
  - (it seemed a safe enough thing to do)

# Maybe 6 Months Later…

- Gray called me up and invited me to visit the DEC Bay Area Research Center

- Said they wanted to hear about my work (that alone should have made me suspicious)

# With Some Trepidation I Flew Out

(folks from Wisconsin don't require too much reason to fly to CA in the winter)

# At the Talk…

- Oddly, quite a few people were there
- At the end, they asked
  **"are you finished?"**

- When I said yes,
  - A lot of cameras appeared.
  - Also a large trophy

- They handed it to me and said we had set the **world record for sorting!**

# Continuing With the Story…

- I basked in the glory for about 5 seconds

- Then they announced:
  - **"We have beaten your record."**
  - **"Please hand the trophy to Chris, we want a picture of the hand-over."**

# Would You Do This Today?

- It was perhaps 6 months work for three researchers
- Yielded one paper in a minor conference

**Today it would be zero papers!**

**I can see the reviews:**

**"Reject: Not enough math, can't be deep."**

**"Reject: Superficial performance evaluation."**

**"Reject: Their sorting program cannot converse in English like the Star Trek computer."**

# Today We Probably Couldn't Spare the Time to do This

- A team of halfway decent researchers:
  - **should (must?) write more than one paper** in six months
  - **should do much more reviewer-proof work** than implementing and running a sorting algorithm

# Was it Worth Doing?

**Yes!**

- Taught me a ton about sampling in parallel DBMS
- Convinced me of the importance of processor architecture in performance
  - the BARC guys smoked us with a uniprocessor implementation by making clever use of the cache
- It was really fun and kept me interested and connected with the field

# War Story #2: 007 Benchmark

Carey, DeWitt, and I decided to benchmark object oriented DBMS


- We
  - designed the benchmark
  - negotiated with four vendors to get them involved
  - implemented it on five systems
    - four commercial plus our own
  - started running tests

**This was a huge amount of work!**

# After Months of Work…

- We received a fax from a law firm representing one of the companies

- The fax ordered us
  - to **stop benchmarking** their system
  - and to **destroy all copies** of their system that we might have


**We reacted extremely maturely as usual…**

# The Next Morning…

- I woke up early

- Made a big pot of coffee

- Drank most of it


- Fired off an angry fax

  - **demanding that the company destroy all copies of our benchmark implementation**

**We knew they were using it for internal testing.**
**We were pissed.**

# 15 Minutes After Sending the Fax

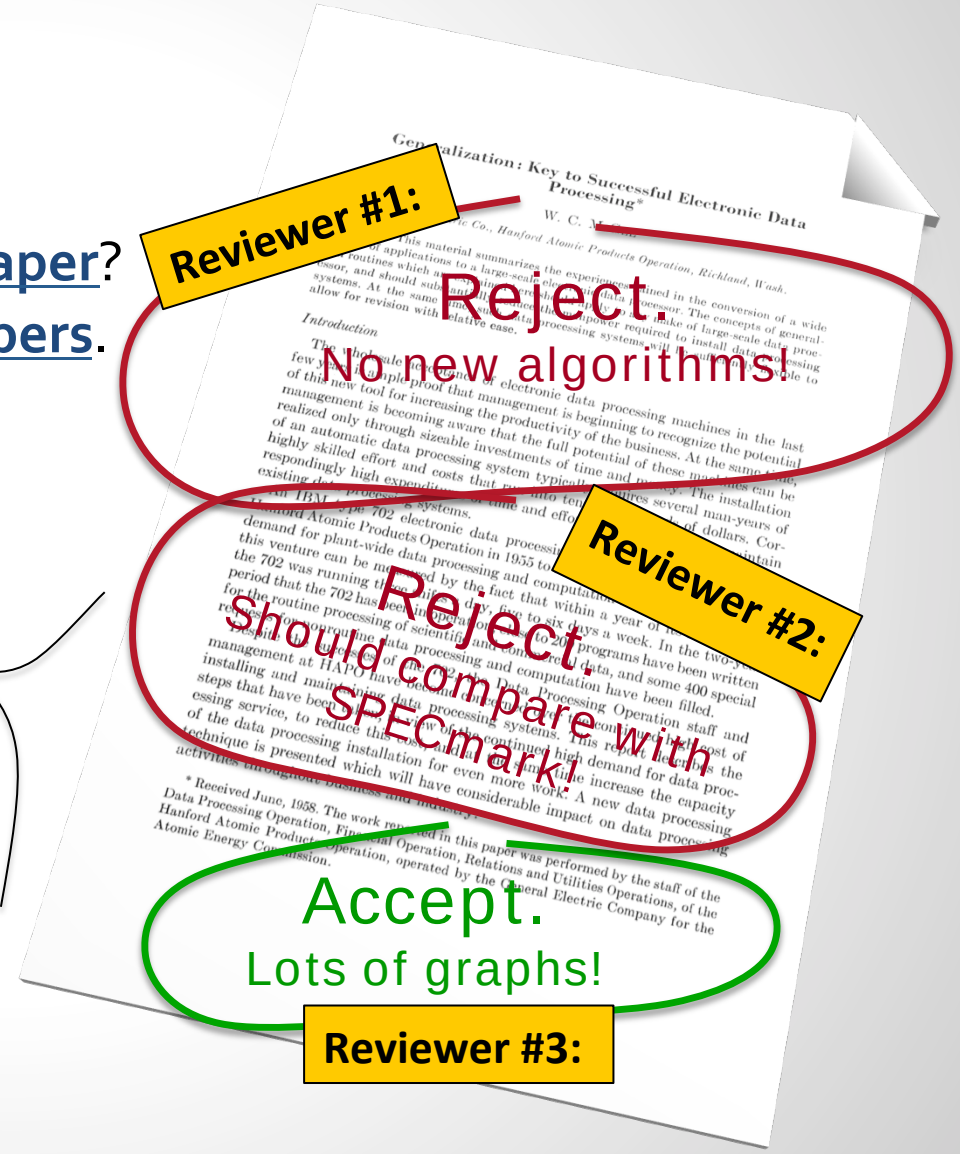
- I received a phone call 

- **The following conversation ensued:**
  - Confused Woman: "Professor Naughton?"
  - Indignant Professor: "Yes?"
  - Confused Woman: "This is First National Bank, we received a fax from you about destroying a benchmark…"
  - Sheepish Professor: "OK, please comply, thanks."

# Would This Project be Smart Now?

- Probably not
  - Three profs, 18+ months, one paper?
  - Again, today would be zero papers.

- Reviews today would be:



# Worth It?

- **We certainly learned a lot about:**
  - OODBMS technology
  - Stresses companies (and researchers) are under with respect to benchmarking
  - Legal issues
  - Pitfalls in designing benchmarks
  - Interaction with popular tech press


**Positive for all of us, but would be discouraged by today's environment**

# Point of the War Stories…

- At least for me, some of **the most rewarding work** did not have a publication as its goal

**Goal** ≠ 

- At least for me, some of **the most rewarding work** did not result in many publications

**Result** ≠ 

- At least for me, some of **the most rewarding work** did not result in any papers that would pass today's program committees

**Result** ≠ 

# These days I fear we are

- **Discouraging work motivated by asking:**
  - Can we learn something?
  - Can we build something?
  - Can we prove something?
  - Can we improve something?

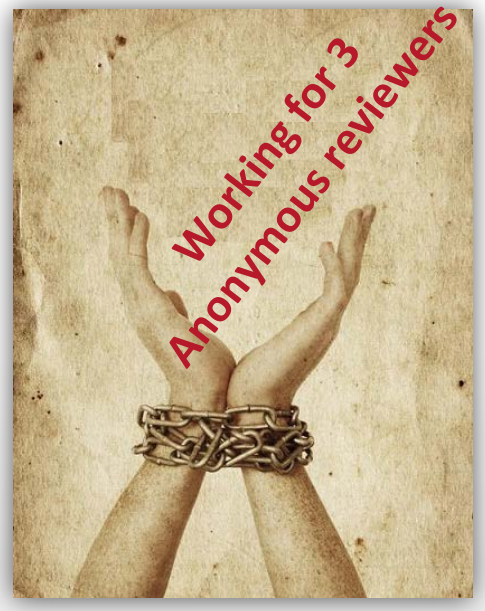
- **Encouraging work motivated by asking:**
  - Can I write a paper before the deadline?

# Today…

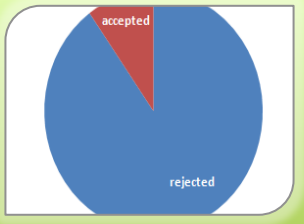
- We are so **frenetically pursuing the next conference deadline**

…under the watchful glare of bad reviewing,

…that the freedom required to do exploratory work is disappearing.



# Three Factors Causing Problems




**Low acceptance rates**


**Emphasis on paper count**


**Bad reviewing**

Let's consider them in order

# The Problem With Low Acceptance Rates

- Very discouraging

- Reduces tolerance for errors in reviewing
- Enforces evaluation of work by **three randomly chosen reviewers rather than the community**
- Makes major event in the course of research acceptance/rejection, not scientific or engineering progress

# How to Fix Low Acceptance Rates

- Increase acceptance rates (duh!)
- Mandate a target
- **Hoped for effects:**
  - Fewer rejections
  - Fewer papers sloshing from conference to conference
  - Reduce the damage done by bad reviewing

# Turning to Paper Count

- **Paper count inflation**
  - To be a success, people feel one has to publish a lot of papers
  - To publish a lot of papers, one has to get past a lot of program committees


- To do this is a more than full time job
- Leaves too little room for activities not focused on generating publications

**[No Time to Explore]**

# Tough to Fix, but…

Let people in on a secret:

- In general paper count is much less important in evaluations than you might think.
- ***Stars are never evaluated by paper count***

It is OK to write a lot of papers.

- Just don't make it the primary metric motivating researchers.
- Don't let it block those who have a different "research style."

# Acceptance Rate Again

- **Hypothesis**: emphasis on paper count can be somewhat ameliorated by increasing acceptance rate
  - If it is easier to publish papers, publishing lots of them will be perceived as less impressive
  - Shift the focus from paper count to paper quality

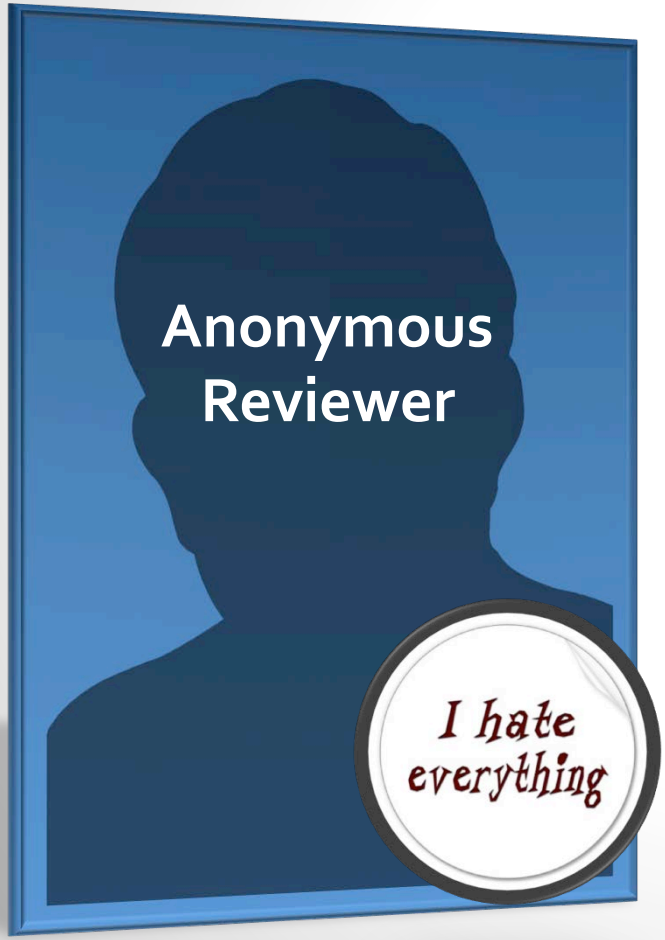
# Third Issue: Bad Reviewing

- Very hard to fix
- Very important to fix
- Extremely important to discuss <u>because it gives me a chance to vent</u>

# Caveat!

- I have received extremely helpful while ultimately negative reviews.
- These reviews have dramatically helped me and my co-authors with the presentation and content of the work.
- I am very grateful to those reviewers.
- These “helpful but negative” reviews are not the same as the “bad” reviews I will discuss next!

# One problem: Reviewers Hate EVERYTHING!




- One anecdote:

  **SIGMOD 2010**

- **350 submissions**
  - Number of papers with all reviews "accept" or higher: **1**
  - Number of papers with average "accept" or higher: **4**

**Either we all suck or something is broken!**

# PC Reviewing is Important

- This is the primary place most researchers get most of their outside feedback

- **This feedback trains most researchers in:**
  - What to do for their next submission
  - How to evaluate others


**Receiving dysfunctional reviews begets writing dysfunctional reviews**

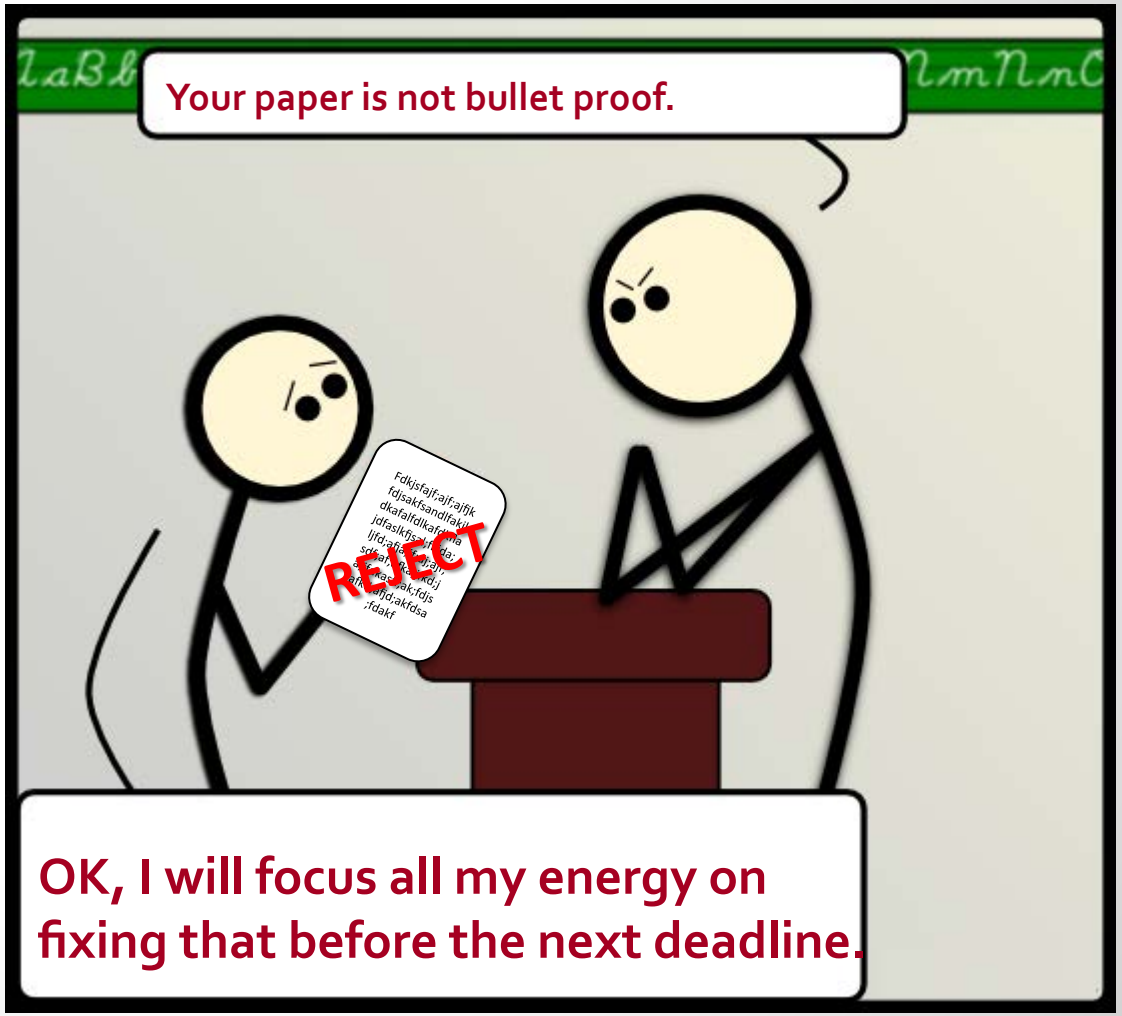
# Why is This Bad?

- Discouraging for authors
- Devastating when it occurs in grant proposal reviewing
  - Funding agencies believe us when we say we suck
- The absolute score can be fixed by enforced scaling


**More fundamental problem:** papers are being rejected for the wrong reasons

# What is Modern Reviewing Most Like?

- Today reviewing is like grading
- Perhaps because so many reviewers are professsors or students or former students?



# But Reviewing is Not Grading

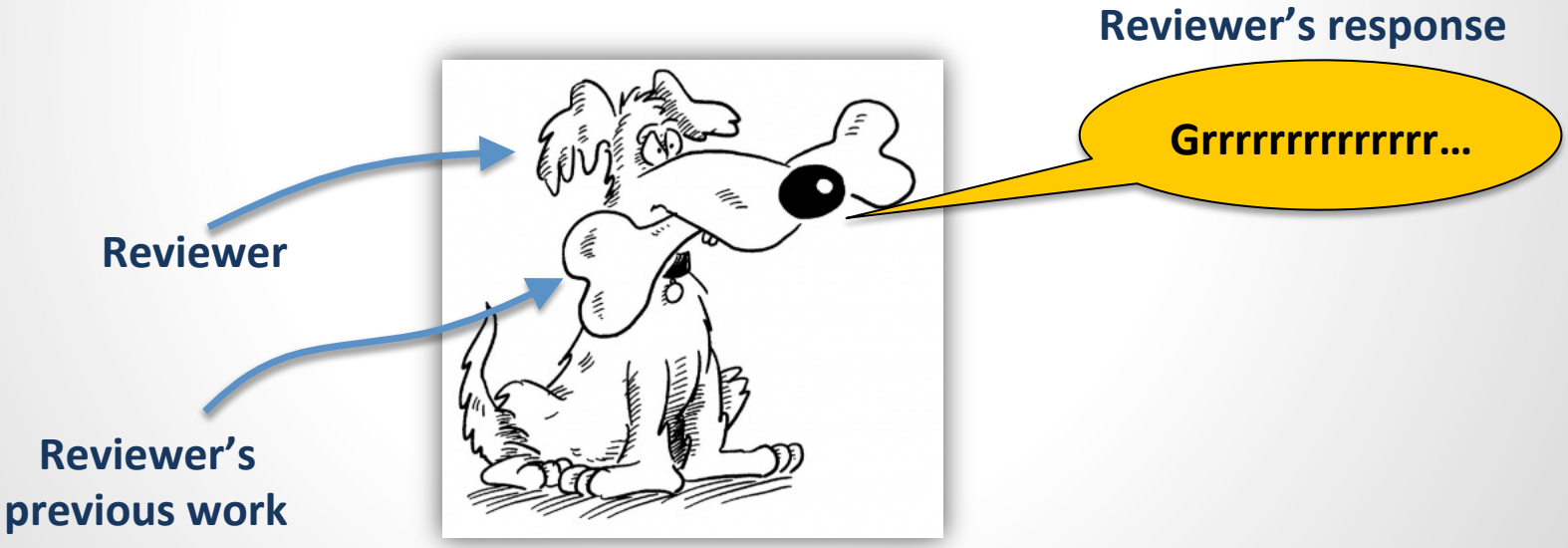
- When grading exams, zero credit goes for thinking of the question
  - **(good) reviewing**: acknowledges that the question can be the major contribution
- When grading exams, zero credit goes for a novel approach to solution.
  - **(good) reviewing**: acknowledges that a novel approach can be more important than the existence of the solution

# Bad Reviewing: Rejection Checklist Reviewing

- **Rejection checklist**:
  - **Is it "difficult"?**
  - **Is it "complete"?**
  - **Can I find any flaw?**
  - **Can I kill it quickly?**
    - **Writing negative reviews** → you are intelligent and have high standards
    - **Finding the positive in a flawed paper** → you are soft, stupid, and have low standards

# Bad Reviewing 2

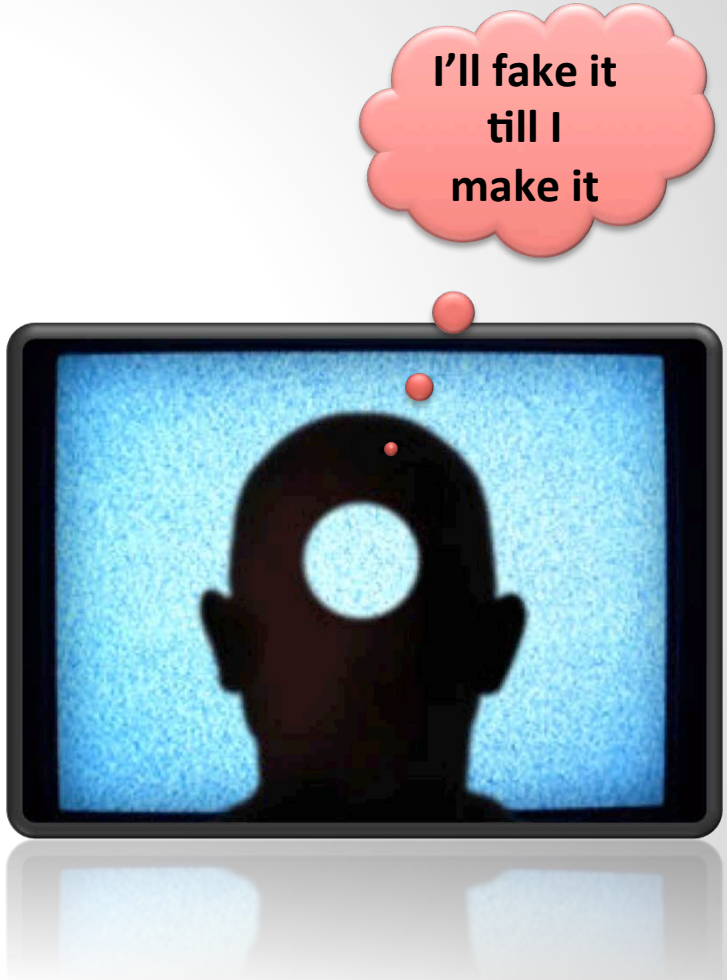
- The "dog and bone" model of reviewing
  - The reviewer is a dog
  - The reviewer's previous work is a bone
  - The author of the new paper is another dog trying to steal the bone
  - Response by the reviewer: **growl like hell**




Systems folks seem much much worse about this than theoreticians

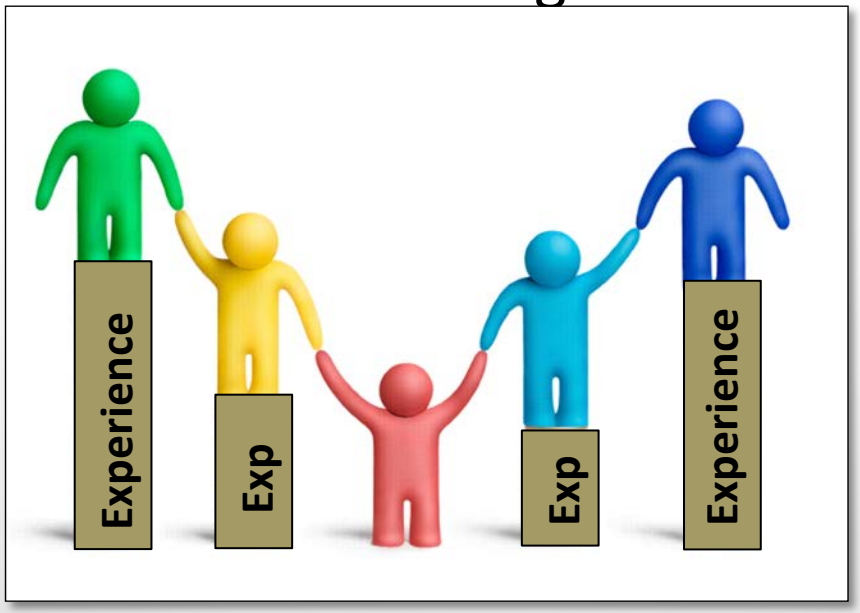
# Bad Reviewing 3

- The **ignorant but confident reviewer**
- Doesn't know what is going on but would rather fake it than admit it
- Seems to feel admission of lack of knowledge would reveal a weakness
  - The exact opposite is true



# Why Are Things Bad?

- **One explanation**: **reviewers are bad people**
- **More likely**: **reviewers are being poorly trained**
- In prehistoric times
  - We had face-to-face PC meetings
  - There was a lot of accountability pressure
  - There was a lot of coaching and mentoring



# What Is the Training Today?

- Reviewers are trained by receiving bad reviews from other reviewers who have received bad reviews in the past

# Can We Change This?

- Here are some ideas, with the goal:
  - **Encourage discussion**
  - **Argue that change is possible**
  - **Agitate for change**
  - **Some are deliberately far fetched!**

# Idea #1: Disarm the Killers

- A large part of "success" in conference submissions is getting a lucky assignment of reviewers
  - If you get a "killer" reviewer, you are dead
  - So you resubmit until you have three non-killers
- **Possible solution:**
  - Mandate a certain percentage of "accepts" from each reviewer

# Idea #2: Shine Light on the Process

- **Publish reviews**
  - Might already encourage more care in reviewing
  - At the very least it would be cathartic
- **Allow voting for best reviews**
  - Somehow reward best reviewers?
- **Allow voting for worst reviews**
  - And?

# Idea #3: Return to the Past

- Require face-to-face PC meetings
- Perhaps have partitioned committees to make this tractable
- Restore accountability for reviews
- Create opportunities for mentorship

# Proposal #4: Single-Blind Reviewing

- **But reverse it:**
  - Reviewer doesn't know author
  - Author knows reviewer

- **So**
  - You know who is criticizing you, but not who you are criticizing
  - Would certainly encourage more thoughtful reviewing

# Proposal #5: Eliminate the Problem

- No reviewing!
- Accept everything.
- Let the community sort things out over time.
- Why are we still behaving as if our proceedings are printed by overworked monks on a Gutenberg press?

# Wrapping Up the Talk

- **There is a lot of angst in the field**
  - Where are the big new ideas?
  - What defines us as a community?
  - How did we miss the web?
  - How long is this guy going to talk in this keynote?

- These are all great questions, worthy of discussion

**But I don't think our success in the next 50 years depends on answering them**

# Looking Forward

- My crystal ball cannot see specific technical trends 50 years out
- It does predict that the three drivers
  - **commercial interest**
  - **common data management challenges**
  - **attractive problems**

will exist in for another 50 years

# So what do we need?

- **We will be OK if we:**
  - Periodically reconnect to these three drivers
  - Create an environment that attracts good people and gives them the freedom and incentive to do good work
- Our success depends on both of these.

# This is Important

- As a research community, despite commercial interest and great problems to work on, **we will not thrive if we create a stifling, depressing environment that discourages a diversity of work.**

# Who is Going to Fix This?

- This is not **"us vs. them"**
- **There is only us.**
- Don't wait for "them" to change things.
- We are "them."

# Are Things Really So Bad?

- Not entirely, not yet.
  - Somehow good work still gets done.
  - Somehow great papers still get written.
  - Somehow great new people still join the community.
  - The community is beginning to respond with great initiatives at various conferences.
- But the overall trend is not good.
- If we don't address this, innovative data management research will get done, but probably not by us.

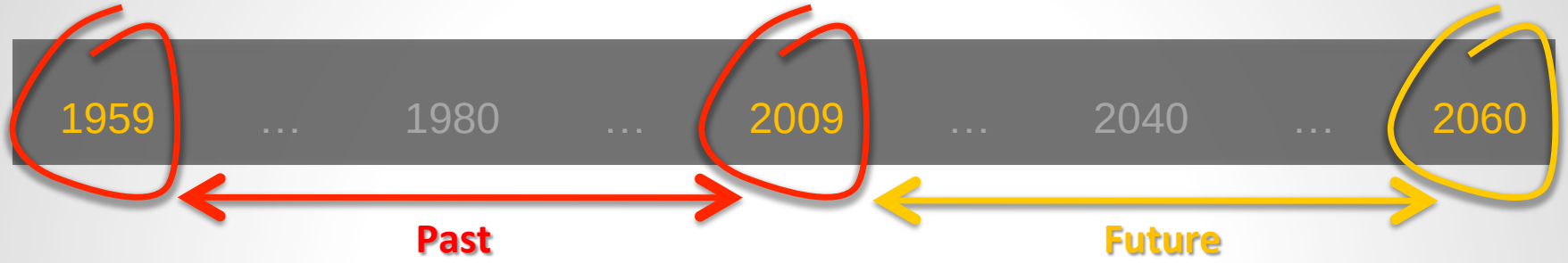
# The Next Big Research Idea

- Maybe the next big idea is not a new data model.
- Maybe it is a new community model.
  - How we create, disseminate, and evaluate research.
  - How we attract, evaluate, and motivate researchers.
- The ideas in this keynote are incremental.
- Can some brilliant person come up with a paradigm-changing idea for the community?

# Acknowledgements

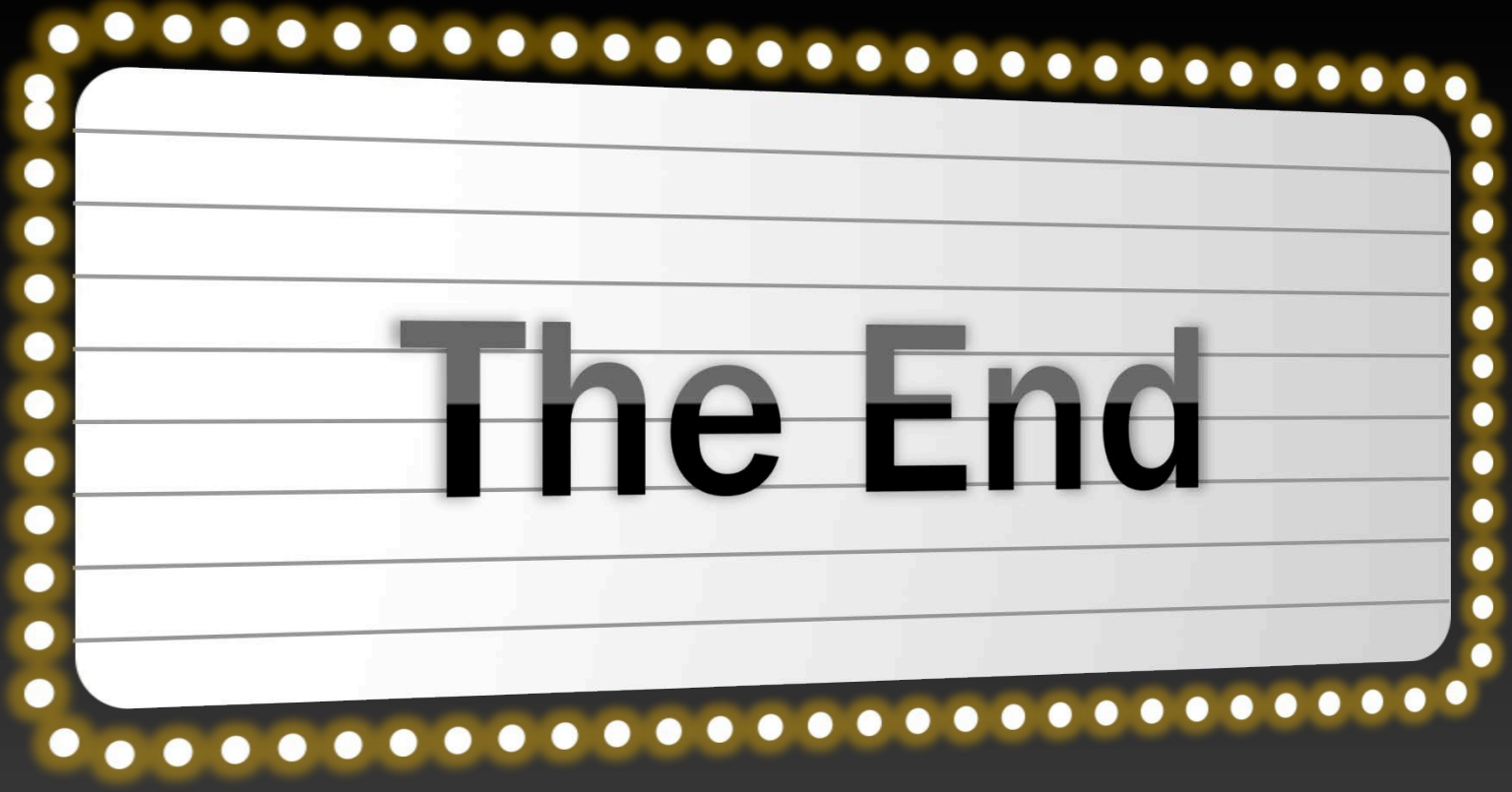

- I'd like to thank the many colleagues near and far whose ideas I have used in this talk.
- I'd like to apologize to the many colleagues near and far whose ideas I should have used in this talk but didn't.
- I'd like to thank an anonymous colleague for heroic work making these slides much less visually boring than is typical for a Naughton talk.

# Closing Thought: We Are In It for the Long Haul




- The McGee paper I discussed was from **1959**
- His most recent publication was in **2009**
- So note to youngsters first publishing in this conference: you should still be publishing in **2060**!

**So it is REALLY in your interest to decide how you would like to spend the next 50 years…**


The End