# CS 4440 – APRIL 7, 2025
# DATABASES IN THE WILD

NICK DOWNEY – GUEST LECTURER

(FOUNDER AND CEO OF MERCH LOGIX, INC., GA TECH CMPE '02)

# POP QUIZ

- HOW MANY SECONDS ARE IN A DAY?
  - 86,400
- WHY ARE MANHOLES ROUND?
  - END-USER PROOF
- WHAT IS THE PRIMARY DIFFERENCE BETWEEN SCIENCE AND ENGINEERING?
  - MONEY
- WHAT IS THE DEFINITION OF LEGACY SOFTWARE?
  - SOFTWARE THAT WORKS
- WHAT IS THE BEST DBMS?
  - TRICK QUESTION

# FASHIONABLE IS NOT A DESIGN CRITERIA

- ORACLE IS REALLY FAST AND HAS TRUSTWORTHY REPLICATION
  - IT IS USUALLY THE MOST EXPENSIVE CHOICE, SOMETIMES PROHIBITIVELY
- IMB'S DB2 IS TRUSTED BY ENTERPRISE
  - PREMIUM SUPPORT IS ONLY AVAILABLE ON IBM HARDWARE
- MICROSOFT SQL SERVER IS EASY TO USE AND A "SAFE" CHOICE
  - FRIENDS DON'T LET FRIENDS RUN MICROSOFT
- POSTGRESQL IS FREE, AND ALMOST AS FAST AS ORACLE
  - YOU HAVE TO MAKE A PLAN FOR SUPPORT YOURSELF

# FAMILIARITY IS NOT A DESIGN CRITERIA

- MONGO DB SCALES EASY AND HAS TONS OF SUPPORT
  - IT IS A DOCUMENT STORE NOT A KEY-VALUE STORE, NOR CAN IT DO GRAPHS
- COUCHBASE DOES BOTH DOCUMENTS AND KEY-VALUE
  - SUPPORT IS NOT AS MATURE AS MONGO DB
- REDIS IS FAST AND GREAT FOR REAL-TIME APPLICATION
  - HOW MUCH RAM YOU GOT THERE, PADRE?
- CASSANDRA IS PRODUCED BY APACHE, AND SUPPORTS WIDE COLUMNS
  - BEWARE OF THE SLIPPERY SLOPE, RDBMS'S STILL HAVE A ROLE TO PLAY

# PIPE WRENCHES MAKE POOR HAMMERS

- UNDERSTAND THE PHYSICAL CONSTRAINTS
  - VOLUME, VARIETY AND VELOCITY OF THE DATA
- UNDERSTAND THE TEMPORAL CONSTRAINTS
  - TIME TO INGEST, TIME TO SYNTHESIZE, TIME TO REPORT, TIME TO RECOVER
- UNDERSTAND THE LEGAL CONSTRAINTS
  - OWNERSHIP, CUSTODY AND COMPLIANCE
- UNDERSTAND THE FINANCIAL CONSTRAINTS
  - COST TO CREATE, COST TO RUN, COST TO MAINTAIN

# GUIDING PRINCIPLES FOR DBA'S

- THINK OF YOURSELF AS A "CURATOR OF THOUGHT"
  - TABLE AND COLUMN NAMES CAN LIVE FOR DECADES, CHOOSE CAREFULLY. E.G. DENOTATIONS VS CONNOTATIONS
- THE CLAIM OF "OVER-NORMALIZATION" IS OFTEN A SMOKE SCREEN FOR APATHY OR LAZINESS.
  - YOU CAN'T QUERY OR JOIN ON COLUMNS AND TABLES THAT DON'T EXIST.
- DO NOT SUCCUMB TO VENDOR LOCK-IN, TRY TO KEEP IT "AGNOSTIC"
  - FANCY IS YOUR ENEMY IN A PRODUCTION FAILURE

# BACKGROUND ON CASE STUDY (1 OF 3)

- RETAIL STORES HOMOGENEITY REALITY CHECK
  - LOCALIZATION
- WHAT IS A STOCK KEEPING UNIT (SKU)
  - SOME UPC'S ARE SKU'S, NOT ALL SKU'S ARE UPC'S
- WHAT IS GROSS MARGIN (NOT A TRICK QUESTION)
- WHAT IS PERFORMANCE DATA
  - TRANSACTION LOG "TLOG" – ONE LINE ITEM FROM ONE RECEIPT
- WHAT IS A PLANOGRAM – LIVE DEMO!

# BACKGROUND ON CASE STUDY (2 OF 3)

- TICKET – A STORE LEVEL SALES RECORD OF ONE OR MORE TLOG'S, NAMELY:
  - THE STORE ID, THE STORE LOCAL DATE, THE STORE LOCAL TIME
  - THE DISTINCT SKU'S YOU PURCHASED, AND THE QUANTITY OF EACH
  - THE UNIT PRICE YOU PAID FOR EACH SKU, AND THE EXTENDED PRICE
  - THE DISCOUNTS AND COUPONS YOU USED
  - AND THE TOTAL SALE AMOUNT
- DEPARTMENT -> CATEGORY -> SUBCATEGORY
  - ORAL CARE -> TOOTH BRUSHES -> ELECTRIC
- DIVISION -> REGION -> DISTRICT -> METRO -> STORE
  - EAST -> SOUTHEAST -> GREATER ATLANTA -> SMYRNA -> CUMB. #121

# BACKGROUND ON CASE STUDY (3 OF 3)

- MERCHANT – PERFORMANCE BASED COMPENSATION
- STORE MANAGER – PERFORMANCE BASED COMPENSATION
- SUPPLIERS – OBVIOUSLY WANT TO SELL AS MUCH AS POSSIBLE
- WALL STREET BENCHMARKS
  - SAME STORE SALES – USUALLY EXPRESSED AS A PERCENTAGE, IN RELATION TO HISTORICAL E.G. 10% INCREASE FROM Q1 TO Q2, OR 18% YEAR-OVER-YEAR.
  - GROSS MARGIN PER SQUARE FOOT, BY CATEGORY – EXP. IN DOLLARS/SQ. FT.
  - AVERAGE TICKET SIZE – EXPRESSED IN DOLLARS

# CASE STUDY – CLIENT PROFILE / OBJECTIVE

- NATIONAL GROCERY CHAIN
- $13,000,000,000.00 (THIRTEEN BILLION) ANNUAL GROSS SALES
- 534 STORES IN THE UNITED STATES
- AVERAGE TICKET SIZE IS $65, WITH 7 DISTINCT SKU'S.
- CLIENT GOAL – COMPUTE GROSS MARGIN PER SQUARE FOOT, PER CATEGORY, OVER ANY ARBITRARY TIME WINDOW ($T_0$ – 104 WKS.), OVER ANY STORE, DISTRICT, REGION OR DIVISION, AND/OR BY PLANOGRAM.

# CASE STUDY – CURRENT STATE

- AWS / LINUX / NGINX / PHP / NODEJS / POSTGRESL
- ALL FLOORPLANS ARE IN-SYSTEM (TB_LOCATION_FLOORPLAN*)
- ALL PLANOGRAMS ARE IN-SYSTEM (TB_PLANOGRAM*)
- ALL PRODUCT DATA ARE IN-SYSTEM (TB_PRODUCT*)
- PERFORMANCE (SALES) DATA (TLOG'S) ARE NOT IN-SYSTEM (TB_???)
- SCHEMA IS PURELY RELATIONAL (~1,200 TABLES)
  - LARGEST CUSTOMERS ARE IN THE 10'S OF GB'S

# WHERE DO WE START!? (HINT: SEE SLIDE 5)

- SIZE OF THE DATA (PHYSICAL CONSTRAINTS)
  - $\frac{\frac{\$13B}{yr}}{\frac{\$65}{ticket}} = 200M\ tickets/yr$
  - $200M\ tickets/yr\ * \left(7\ \frac{tlog's}{ticket}\right) = 1.4B\ tlog's/yr$
  - $104\ wks\ * 52wk/yr\ * 1.4B\ \frac{tlog's}{yr} = 2.8B\ tlog's$
- GREAT! HOW BIG IS ONE (1) TLOG?
  - NEXT SLIDE!

# ANATOMY OF A TRANSACTION / RECEIPT

```
{
     "tid": numeric(32,8), //external surrogate key
  "store" : numeric(6),    //store number/surrogate key
   "date" : date_stamp,    //date of sale local to store
   "time" : time_stamp,    //time of sale local to store
  "tlog" : [               //array of items purchased in this sale
    {
              "SKU" : numeric(32,0),  //unique sku number of this item
        "quantity" : int,             //quantity purchased
      "unit price" : numeric(9,2),    //per unit price
      "ext. price" : numeric(9,2),    //quantity * unit price
       "unit cost" : numeric(9,2),    //per unit cost
         "coupons" : [],              //array of coupons used for this item
       "discounts" : []               //array of discounts used for this item
    }
  ]
  "total list price" : numeric(9,2),
   "total discounts" : numeric(9,2),
     "total coupons" : numeric(9,2),
        "total sale" : numeric(9,2)
}
```

# APPROXIMATE SIZE ON DISK

- ONE TLOG CAN BE SAFELY APPROXIMATED AS "MANY HUNDREDS OF BYTES"
  - WE CAN SANELY ROUND UP TO 1 KILOBYTE (1KB)
- FROM EARLIER SLIDE
  - $104\ wks\ * \frac{52wks}{yr}\ * 1.4B\ \frac{tlog's}{yr} = 2.8B\ tlog's$
  - $2.8B\ tlog's\ * 1{,}024\ \frac{bytes}{tlog} \cong 2.8TB$
- HOW MANY BYTES PER DAY COMING IN?
  - $\frac{2.8TB}{2yr} = \frac{1.4TB}{365days} = 3.8GB/day$

# TEMPORAL CONSIDERATIONS (INGESTION)

- HOW MANY TLOGS COMING IN AND AT WHAT FREQUENCY?
  - $\frac{1.4B\frac{tlog's}{yr}}{365\frac{day's}{yr}} = \frac{3.8M\ tlog's}{day} \cong \frac{160k\ tlog's}{hr} \cong \frac{2.6k\ tlog's}{min} \cong 45\ \frac{tlog's}{second}$
- HOW LONG DOES THE VALIDATION ROUTINE FOR ONE (1) TLOG TAKE?
  - ARRAY OF TEXT (BYTES) IS ACTUALLY VALID JSON (UNICODE/ASCII, ESCAPED?, ETC)
  - REGULAR EXPRESSION ON TOTAL PAYLOAD AND/OR EACH FIELD
  - VALIDATE SURROGATE KEYS (STORE, SKU) AND SANITY (ALL $ >= 0, ALL QTY > 0)
  - THE CORRECT ANSWER IS: MICROSECONDS
- WHAT IS THE WRITE SPEED OF THE DATABASE?
  - POSTGRES CAN KEEP UP – IF PROPERTY CONFIGURED

# TEMPORAL CONSIDERATIONS (ANALYSIS / REPORTING)

- WE'RE BEING PAID FOR SYNTHESIS AND REPORTING
- WHAT, IF ANY, INDEXES ARE ON THESE DATA?
  - HOW LONG DOES VACUUMING TAKE ON BILLIONS OF ROWS?
  - HOW MUCH SPACE DO THE INDEXES OCCUPY ON DISK?
- WHAT IS THE READ SPEED, WHILE WRITING, OF THE DB AND DISK?
  - WHAT IS ROUND-TRIP TIME FROM USER TO DISK AND BACK?
  - MOST OF THE TIME "TOO SLOW" - SPLIT READ/WRITE?
- DO WE PAY A PENALTY FOR MASS DELETE?
  - TABLE LOCKING IS THE BANE OF EVERYONE'S DIGITAL LIVES

# TEMPORAL CONSIDERATIONS (ANALYSIS)

- THE BASIC USE CASE: HOW MUCH MONEY DID 1 SKU MAKE, ON 1 DAY, AT 1 STORE. FORMALLY: "WHAT IS THE GROSS MARGIN PER SKU, PER STORE, PER DAY."
  - E.G. HOW MUCH MONEY DID I EARN ON AVOCADOS AT STORE #456, YESTERDAY.
- THEY WANT STORE-ORG-LEVEL ROLL-UPS, AND SLIDING WINDOWS!
  - E.G. HOW MUCH MONEY DID I EARN ON AVOCADOS AT ALL THE STORES IN DISTRICT #12, LAST WEEK. DID BANANAS DO ANY BETTER?
- THEY ALSO WANT CATEGORY-LEVEL ROLL-UPS LAYERED ON-TOP
  - E.G. HOW MUCH MONEY DID WE EARN ON PRODUCE AT ALL THE STORES IN THE METRO-PHOENIX REGION, OVER THE TRAILING 4-WEEKS.

# TEMPORAL CONSIDERATIONS (ANALYSIS CONT.)

- A FANCY USE CASE: HOW MUCH GROSS MARGIN DID THIS PLANOGRAM GENERATE IN THE LAST FISCAL QUARTER.
  - E.G. HOW MUCH MONEY DID I EARN ON EACH SKU AT ON THIS PLANOGRAM AT ONE OR MORE STORES OVER QX. PRESUMES SAME PLANOGRAM AT MULTIPLE STORES.
- A FANCIER USE CASE: HOW DID THE GROSS MARGIN FOR PLANOGRAM A COMPARE TO PLANOGRAM B, OVER ALL OF LAST YEAR
  - E.G. WHICH OF THESE TWO SIMILAR PLANOGRAMS HAS BETTER MARGIN PERFORMANCE.
- A YET FANCIER USE CASE: HOW DID THE GROSS MARGIN FOR A SET OF PLANOGRAMS COMPARE TO ANOTHER SET OF PLANOGRAMS FOR A GIVEN REGION.
  - PLEASE HOLD FOR A BRIEF STORY ABOUT COKE-ZERO

# EXISTENTIAL DECISION MAKING TIME

- DEAR DBA – WHAT SAY YOU? HOW SHOULD WE GO ABOUT THIS? PLEASE LET US KNOW YOUR RECOMMENDATIONS FOR SYSTEMS, ARCHITECTURE, SCHEMA, AND/OR CRON.
  - EXISTING SYSTEM IS POSTGRESQL, AND PURELY RELATIONAL
  - INBOUND DATA IS LIKELY TO IMPOSE SIGNIFICANT PHYSICAL CONSTRAINTS
  - USE CASES ARE LIKELY TO IMPOSE SIGNIFICANT TEMPORAL CONSTRAINTS
  - SERVICE LEVEL AGREEMENT (SLA) MANDATES:
    - 99.99% UPTIME (~52 MIN/YR. OF *UNSCHEDULED* DOWNTIME)
    - PAGE LOAD TIME LESS THAN 5 SECONDS

# HOW DID IT SHAKE OUT?

- TLOG DATA IS VALIDATED BY AN AWS LAMDA FUNCTION - .000015 SEC.
- VALIDATED TLOG DATA IS STORED IN MONGO DB (NOSQL)
- CRON JOBS COMPUTE ROLL-UP DATA (FROM NOSQL) EVERY 15 MINUTES
  - DEPENDENT/DERIVED ROLL-UPS ARE ON TRIGGERS
- ROLL-UP DATA IS STORED IN POSTGRESQL RELATIONAL TABLES
  - MEMCACHED + CUSTOM EXTENSION KEEP THINGS FRESH
- USER-INTERFACES ARE PUSHED REFRESH NOTIFICATION(S)

# KEY TAKE AWAYS

- THINK ABOUT YOUR PROBLEMS HOLISTICALLY AND CRITICALLY BEFORE YOU START MAKING DECISIONS THAT ARE EXPENSIVE TO REVERSE.
  - THERE'S NEVER ENOUGH TIME AND MONEY TO DO IT RIGHT THE FIRST TIME, BUT THERE IS ALWAYS ENOUGH TIME AND MONEY TO DO IT TWICE.
- 90% OF GAINFUL SOFTWARE EMPLOYMENT IS MAINTENANCE, NET-NEW CODE IS THE EXCEPTION, NOT THE RULE.
  - DESIGN AND IMPLEMENT YOUR SYSTEMS AS IF YOU PERSONALLY HAVE TO MAINTAIN THEM FOR THE NEXT 10 YEARS.
- "I DON'T KNOW" IS A LEGITIMATE ANSWER, AND WHEN USED APPROPRIATELY SHOULD BE HEARD OFTEN.
  - DUNNING KRUGER IS WORSE FOR YOUR LONG-TERM FUTURE THAN MOST MEDICAL CONDITIONS. LOOK IT UP.

# QUESTIONS AND ANSWERS

- GOT ANY BRAIN CELLS LEFT?
  - I WOULD LOVE TO TAKE YOUR QUESTIONS!

LAST SLIDE!

WHO WANTS A JOB?!

COME GET A BUSINESS CARD!