A Simple Way to Estimate the Cost of Downtime

Dave Patterson

EECS Department

University of California, Berkeley

http://roc.cs.berkeley.edu/projects/downtime

November 2002

Motivation

- **Our perspective: Dependability and Cost of Ownership are the upcoming challenges**
  - Past challenges: Performance and Cost of Purchase
- **Ideal: compare (purchase cost + outage cost)**
- **But companies claim most customers won't pay much more for more dependable products**
  1. How do you measure product availability?
  2. How much money would greater availability save?
- **Researchers are starting to benchmark dependability: commonplace in 2 to 4 years?**
  - Predict hours of downtime per year per product
- **If customers cannot easily estimate downtime costs, who will pay more for dependability?**
  - **outage cost** = downtime hours X **cost/hour of downtime**

2000 “Downtime Costs” (per Hour)

• Brokerage operations	$6,450,000
• Credit card authorization	$2,600,000
• Ebay (1 outage 22 hours)	$225,000
• Amazon.com	$180,000
• Package shipping services	$150,000
• Home shopping channel	$113,000
• Catalog sales center	$90,000
• Airline reservation center	$89,000
• Cellular service activation	$41,000
• On-line network fees	$25,000
• ATM service fees	$14,000

Sources: InternetWeek 4/3/2000 + *Fibre Channel: A Comprehensive Introduction*, R. Kembel 2000, p.8.
"...based on a survey done by Contingency Planning Research."

One Approach

1. **Estimate on-line income lost during outages**
   - (Online Income / quarter divided by hours / quarter) X hours of downtime

- **Lost off-line sales? Employee productivity?**

2. **Interview employees after an outage, ask how many were idled by the outage, and calculate their salaries and benefits**
   - How many employees would answer (honestly)? (Big Brother data collection?)
   - How many companies would spend the money to collect this information?

- **But want CIO, system administrator to easily estimate costs to evaluate future purchases**

A Simple Estimate

- ***Estimated Average Cost of hour of downtime*** =

  Employee Costs per Hour * Fraction Employees Affected by Outage

  + Average Revenue per Hour * Fraction Revenue Affected by Outage
- ***Employee Costs per Hour***: total salaries and benefits of employees per week divided by the average number of working hours
- ***Average Revenue per Hour**:* total revenue per week divided by average number of open hours
- **“Fraction Employees Affected by Outage”** and **“Fraction Revenue Affected by Outage”** are just educated guesses or plausible ranges
  - **Since evaluating purchases, estimates OK**

Caveats

- **Ignores cost of repair, such as cost of operator overtime or bringing in consultants**
- **Ignores daily, seasonal variations in revenue**
- **Indirect costs of outages can be as important as these more immediate costs**
  - Company morale can suffer, reducing productivity for periods that far exceed the outage
  - Frequent outages can lead to a loss of confidence in the IT team and its skills (IT blamed for everything)
  - Can eventually lead to individual departments hiring their own IT people, which lead to higher direct costs
- **Hence estimate tends to be conservative**

3 Examples (2001)

Institution	Revenue Hours per Week	Employee Hours per Week
EECS Dept. U.C. Berkeley	- -	10 hrs x 5 days
Amazon	24 hrs x 7 days (Online)	10 x 5
Sun Microsystems	24 x 5 (Offline, but Global)	10 x 5

Example 1: EECS Dept. at U.C.B.

- State funds: 68 staff @ $100k / week + 90 faculty @ $200k/week (including benefits)
- External (federal) funds per week:
  - School year: 670 (students, staff) @ $450k
  - Summer: 635 (students, faculty, staff) @ $675k
- ~ $44,000,000 / year in Salaries & Benefits or ~ $850,000 / week
- @ 50 hours / week => ~ $17,000 per hour
- If outage affects 80% employees, ~ $14,000 / hour
- Guess at 2002 outages: ~ 50 hours => ~ $680,000 per year

Example 2: Amazon 2001

- Revenue $3.1B/year, with 7744 employees
- Revenue per hour (24x7): ~ $350,000
- If outage affects 90% revenue: ~ $320,000
- Public quarterly reports do not include salaries and benefits directly
- Assume avg. annual salary is $85,000
  - UC staff ~$70,000; Amazon 20% higher?
- $656M / year, or $12.5M / week for all staff
- @ 50 hours / week => ~ $250,000 per hour
- If outage affects 80% employees: $200,000
- Total: ~ $520,000 / hour
- Note: Employee cost/hour ~ revenue

Example 3: Sun Microsystems 2001

- Revenue $12.5B/year, with 43,314 employees
- Revenue per hour (24x5): ~ $2,000k
- If outage affected 10% revenue: ~ $200,000
- Assume avg. annual salary is $100,000
  - More engineers than Amazon, so 20% higher
- $4331M/year, or $83M / week for all staff
- @ 50 hours / week => ~ $1,660,000 per hour
- If outage affects 50% employees: ~$830,000
- Total: ~ $1,030,000 / hour
- Note: Employee costs 80% of outage cost

Purchase Example

- Comparing 2 RAID disk arrays, same capacity;

  Brand X Purchase Price: $200,000

  Brand Y Purchase Price: $250,000
- Dependability benchmarks suggest over 3-year product lifetime 10 hours of downtime for Brand X vs. 1 hour of downtime for Brand Y
- Using EECS downtime costs of $14k per hour:

  Brand X: $200,000+10x$14,000 = $340,000

  Brand Y: $250,000+ 1x$14,000 = $264,000
- Helps organization justify selecting more dependable but more expensive product

Observations

- Data was easy to collect
  - 3 emails inside EECS
  - Quarterly Reports for Amazon and Sun
- Quantifies cost difference of planned vs. unplanned downtime, which is not captured by availability (99.9...9%)
- Employee productivity costs, traditionally ignored in such estimates, are significant
  - Even for Internet companies like Amazon
  - Dominate traditional organizations like Sun
- Outages at universities and government organizations can be expensive, even without a computer-related revenue stream
- Include employee productivity in costs!

Is greater accuracy needed?

- Downtime can have subtle, difficult to measure effects on sales and productivity
  - Are sales simply re-ordered after downtime, Or do customers switch to more dependable company?
  - Do employees just do other work during downtime, Or does downtime result in lost work, psychological impact, so that it takes longer than the downtime to recover?
- Hence spending much more for accuracy may not be worthwhile
  - Also, metric is less likely to see widespread use if it's much more difficult to generate

Conclusion

- **Employee productivity costs are significant**
- **Goal: easy-to-calculate estimate to lays groundwork for buying dependable products**
- **Unclear what precision is needed or possible, so make easy for CIOs, SysAdmins to calculate**

Estimated Average Cost of 1 hour of downtime =

Employee Costs per Hour * Fraction Employees Affected

+ Average Revenue per Hour * Fraction Revenue Affected

- **Web page for downtime calculation for you:**

http://ROC.cs.berkeley.edu/projects/downtime

FYI: ROC Project

- **Recovery Oriented Computing (ROC) Project collecting real failure for Recovery Benchmarks**
  - If interested in helping, let us know, and see

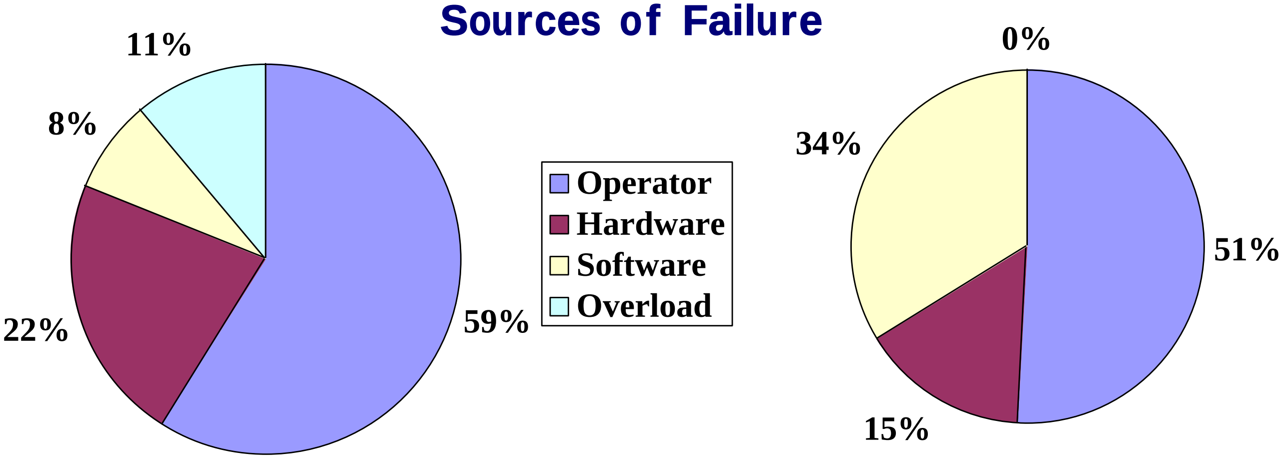
http://ROC.cs.berkeley.edu

BACKUP SLIDES

Human error

- **Human operator error is the leading cause of dependability problems in many domains**

Sources of Failure



Public Switched Telephone Network

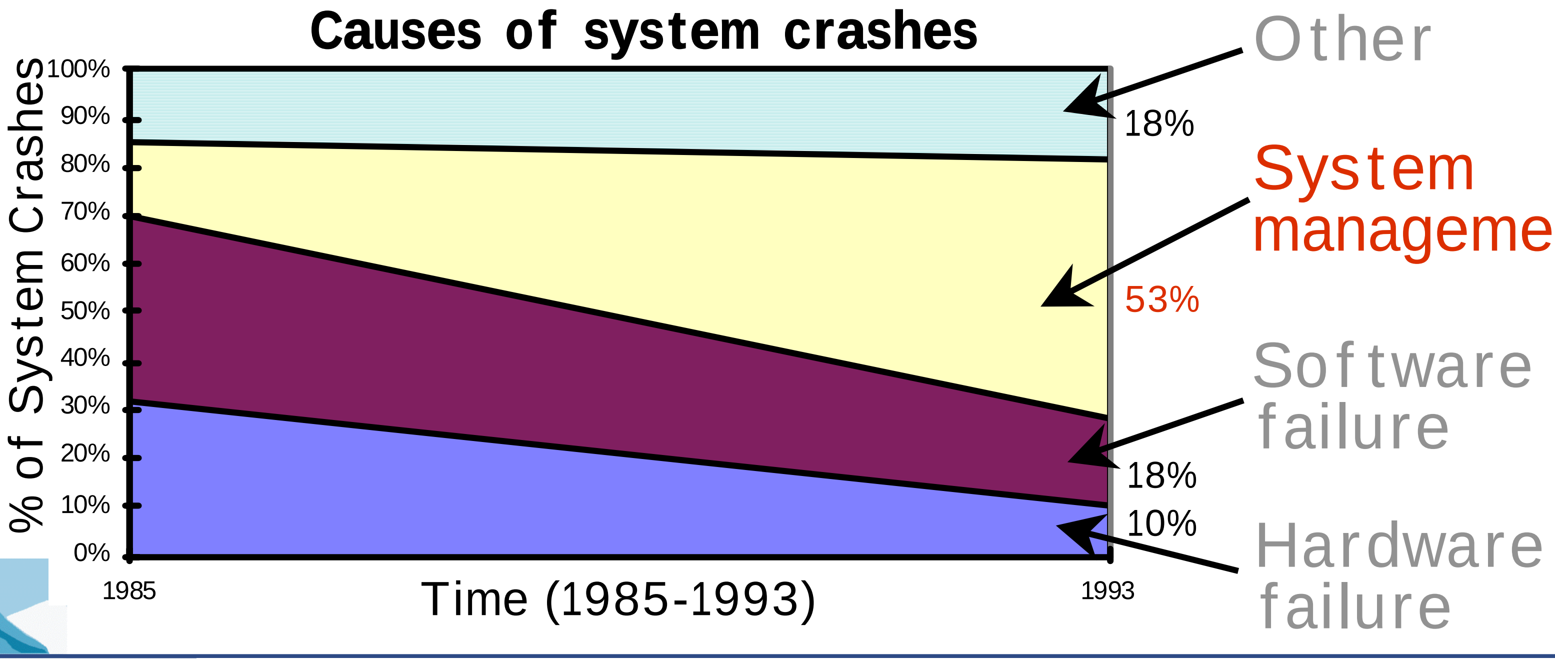
Average of 3 Internet Sites

- **Operator error cannot be eliminated**
  - humans inevitably make mistakes: “to err is human”
  - *automation irony* tells us we can’t eliminate the human

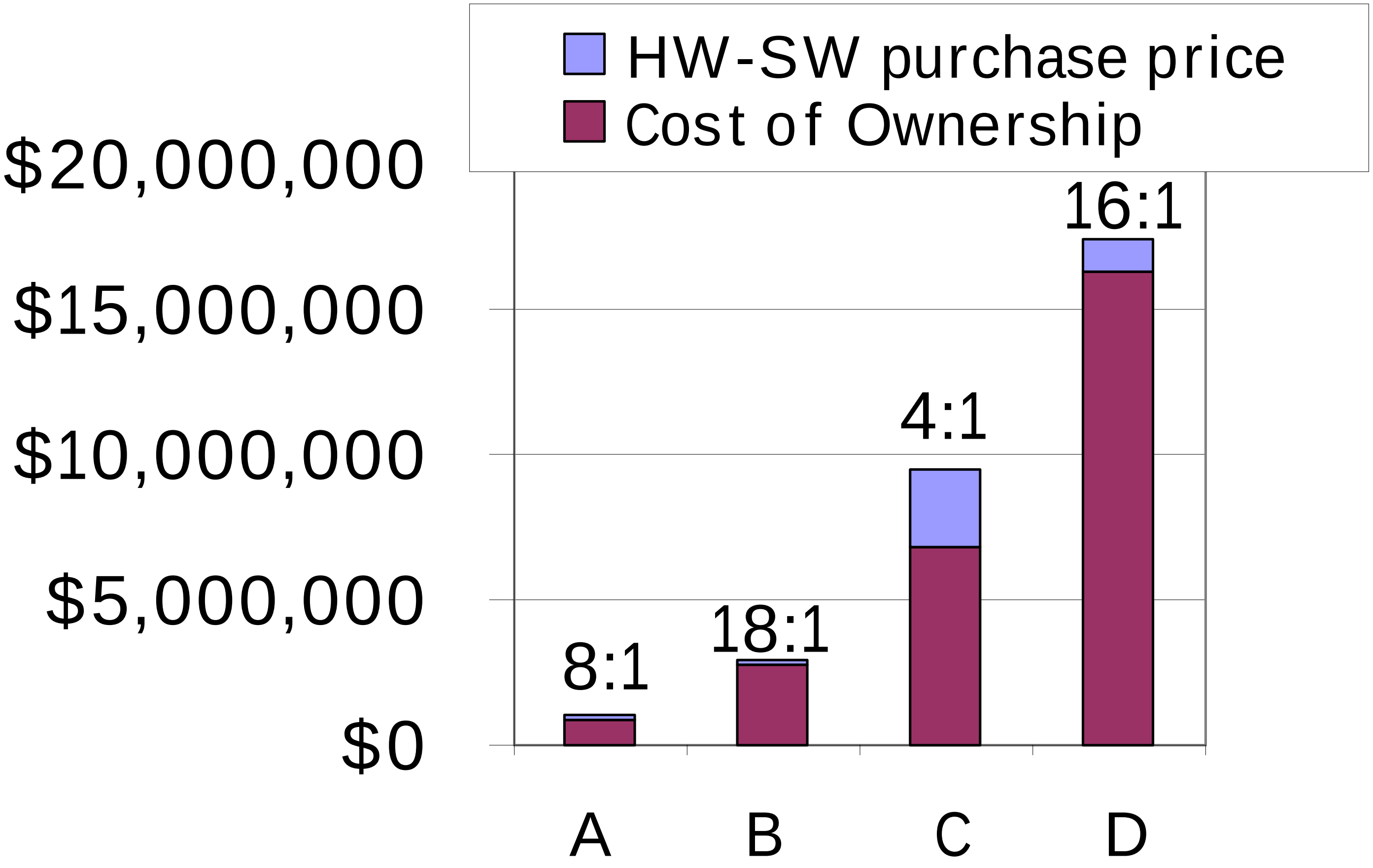
Source: D. Patterson et al. Recovery Oriented Computing (ROC): Motivation, Definition, Techniques, and Case Studies, UC Berkeley Technical Report UCB//CSD-02-1175, March 2002.

ROC Part I: Failure Data
Lessons about human operators

- **Human error is largest single failure source**
  - – HP HA labs: human error is #1 cause of failures (2001)
  - – Oracle: half of DB failures due to human error (1999)
  - – Gray/Tandem: 42% of failures from human administrator errors (1986)
  - – Murphy/Gent study of VAX systems (1993):


Causes of system crashes

Total Cost of Ownership: Ownership vs. Purchase

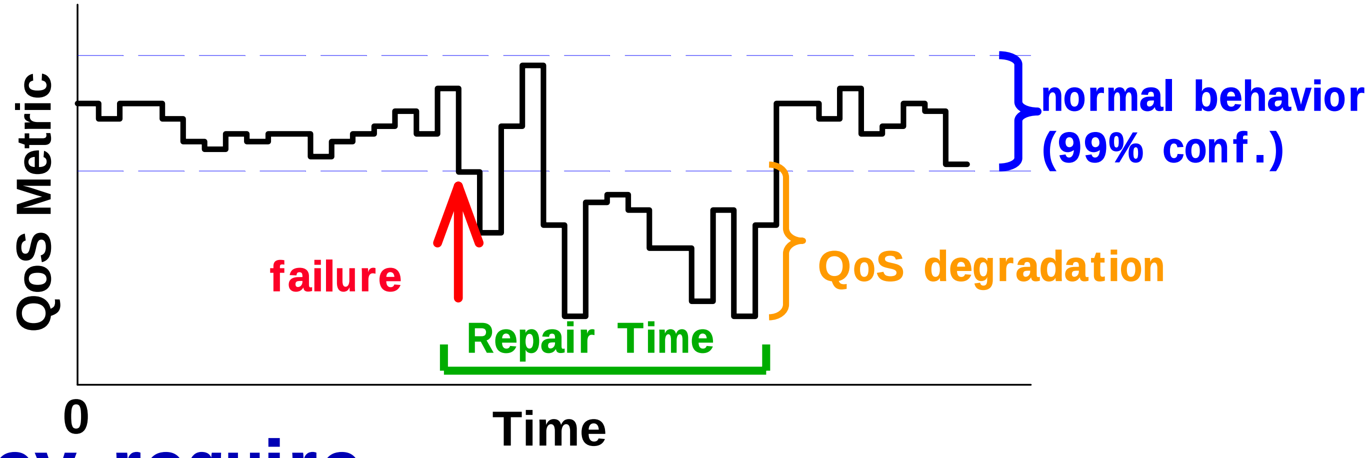


- **HW/SW decrease vs. Salary Increase**
  - 142 sites, 1200-7600 users/site, $2B/yr sales

Source: "The Role of Linux in Reducing the Cost of Enterprise Computing", IDC white paper, sponsored by Red Hat, by Al Gillen, Dan Kusnetzky, and Scott McLaron, Jan. 2002, available at www.redhat.com

Recovery benchmarking 101

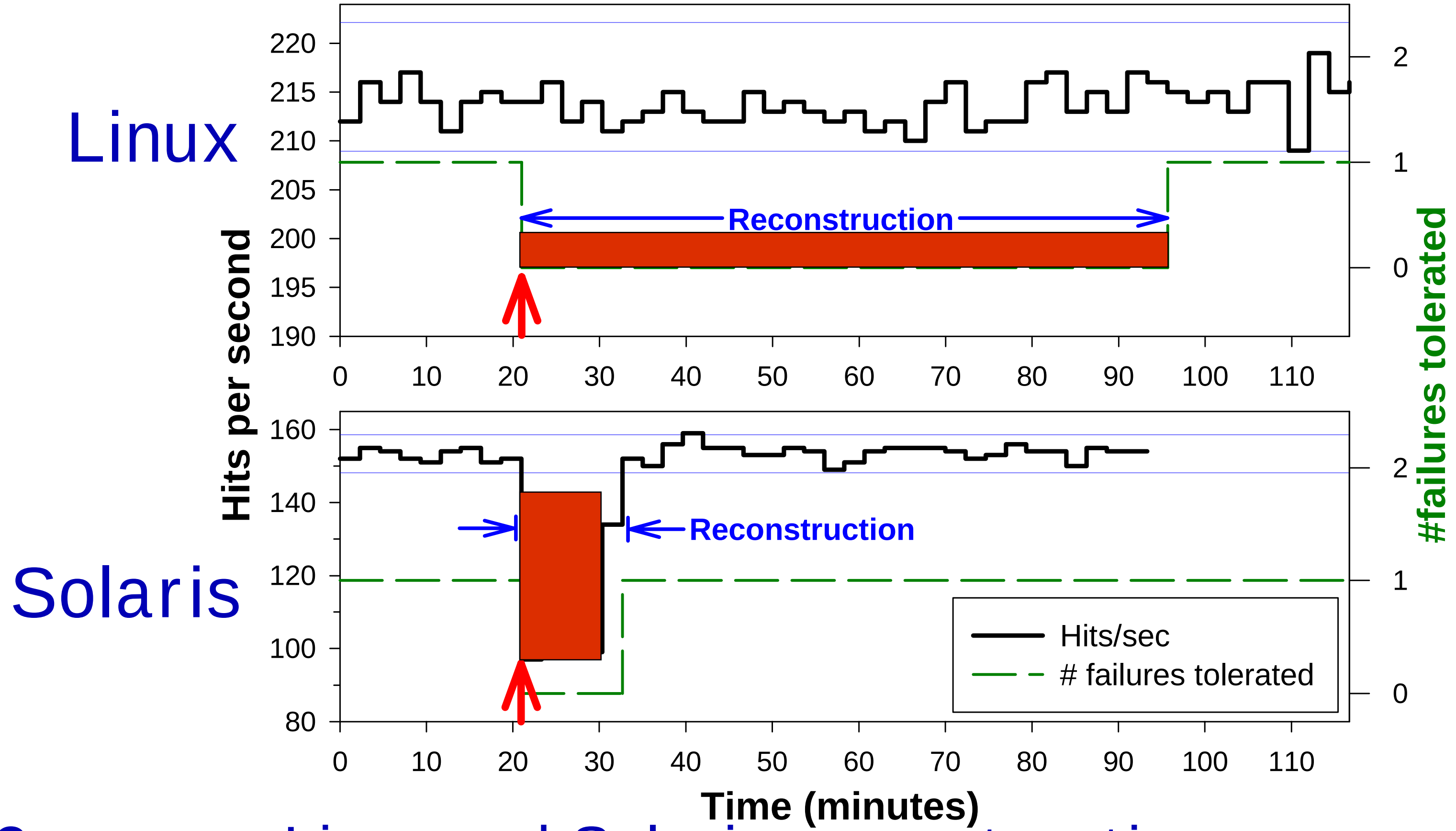
- **Recovery benchmarks quantify system behavior under failures, maintenance, recovery**



- **They require**
  - A realistic workload for the system
  - Quality of service metrics and tools to measure them
  - Fault-injection to simulate failures
  - Human operators to perform repairs

Source: A. Brown, and D. Patterson, "Towards availability benchmarks: a case study of software RAID systems," *Proc. USENIX*, 18-23 June 2000

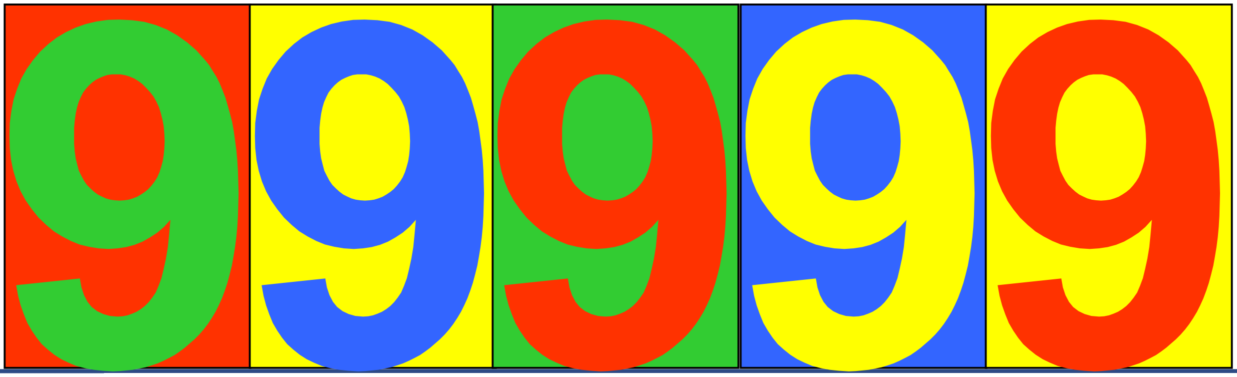
Example: 1 fault in SW RAID



- Compares Linux and Solaris reconstruction
  - **Linux:** Small impact but longer vulnerability to 2nd fault
  - **Solaris:** large perf. impact but restores redundancy fast
  - **Windows**: did not auto-reconstruct!

Dependability: Claims of 5 9s?

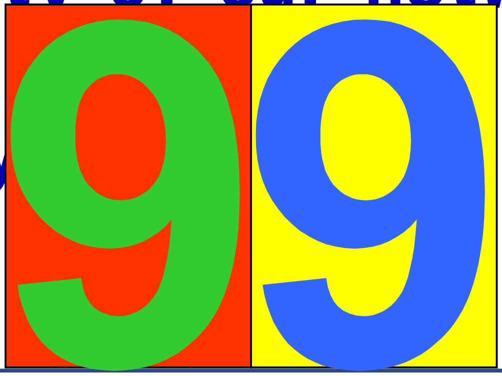
- 99.999% availability from telephone company?
  - **AT&T switches < 2 hours of failure in 40 years**
- Cisco, HP, Microsoft, Sun … claim 99.999% availability claims (5 minutes down / year) in marketing/advertising
  - **HP-9000 server HW and HP-UX OS can deliver 99.999% availability guarantee "in certain pre-defined, pre-tested customer environments"**
  - **Environmental? Application? Operator?**



5 9s from Jim Gray's talk:
"Dependability
in the Internet Era"

“Microsoft fingers technicians for crippling site outages”

By Robert Lemos and Melanie Austria Farmer, ZDNet News, January 25, 2001

- Microsoft blamed its own technicians for a crucial error that crippled the software giant's connection to the Internet, almost completely blocking access to its major Web sites for nearly 24 hours… a "router configuration error" had caused requests for access to the company’s Web sites to go unanswered…
- "This was an operational error and not the result of any issue with Microsoft or third-party products, nor with the security of our networks," a Microsoft spokesman said.
  - (5 9s possible if site stay  rs!)

The ironies of automation

- **Automation doesn't remove human influence**
  - shifts the burden from operator to designer
    - » designers are human too, and make mistakes
    - » unless designer is perfect, human operator still needed
- **Automation can make operator's job harder**
  - reduces operator's understanding of the system
    - » automation increases complexity, decreases visibility
    - » no opportunity to learn without day-to-day interaction
  - uninformed operator still has to solve exceptional scenarios missed by (imperfect) designers
    - » exceptional situations are already the most error-prone
- **Need tools to help, not replace, operator**

Source: J. Reason, Human Error, Cambridge University Press, 1990.

Recovery-Oriented Computing Philosophy

"If a problem has no solution, it may not be a problem, but a fact, not to be solved, but to be coped with over time"

— Shimon Peres ("Peres's Law")

- **People/HW/SW failures are facts, not problems**
- **Recovery/repair is how we cope with them**
- **Improving recovery/repair improves availability**
  - UnAvailability = $\frac{MTTR}{MTTF}$ *(assuming MTTR much less than MTTF)*
  - 1/10th MTTR just as valuable as 10X MTBF
- **ROC also helps with maintenance/TCO**
  - since major Sys Admin job is recovery after failure
- **Since TCO is 5-10X HW/SW $, if necessary spend disk/DRAM/CPU resources for recovery**

Five “ROC Solid” Principles

1. **Given errors occur, design to recover rapidly**
2. **Extensive sanity checks during operation**
   - To discover failures quickly (and to help debug)
   - Report to operator (and remotely to developers)
3. **Tools to help operator find, fix problems**
   - Since operator part of recovery; e.g., hot swap; undo; graceful, gradual SW upgrade/degrade
4. **Any error message in HW or SW can be routinely invoked, scripted for regression test**
   - To test emergency routines during development
   - To validate emergency routines in field
   - To train operators in field
5. **Recovery benchmarks to measure progress**
   - Recreate performance benchmark competition

Recovery Benchmarks (so far)

- **Race to recover vs. race to finish line**
- **Recovery benchmarks involve people, but so do most research by social scientists**
  - “Macro” benchmarks for competition, must be fair, hard to game, representative; use ~ 10 operators in routine maintenance and observe errors; insert realistic HW, SW errors stochastically
  - “Micro” benchmarks for development, must be cheap; inject typical human, HW, SW errors
- **Many opportunities to compare commercial products and claims, measure value of research ideas, … with recovery benchmarks**
  - initial recovery benchmarks find peculiarities
  - Lots of low hanging fruit (~ early RAID days)