2009 ARS, North America, San Diego

Track 2, Session 5

Begins at 9:10 AM, Wednesday, June 10th

Current Time:
5:22 PM

How Statistical Modeling of Field Reliability Failures Guided Efforts on Problem Resolution

David C. Trindade, Ph.D.
Distinguished Engineer
Sun Microsystems, Inc.

Agenda

• Introduction	5 min
• Addressing Field Failures	5 min
• Data from the Field	5 min
• Reliability Analysis of Field Failures	10 min
• Application of Model	10 min
• Cause Implications	5 min
• Physical Mechanisms/Remediation	5 min
• Summary	5 min
• Questions	10 min

Introduction

- Field Failures in New Servers
  - In 1999, Sun Microsystems began experiencing a number of field failures in new servers
  - The failures were sudden, unexpected, and could cause the system to “panic”.
- Engineers spent considerable efforts to restore systems to operation and prevent recurrence
  - Boards experiencing a failure were replaced and returned to Sun for analysis.
  - Extensive data logging of conditions at the time of the failure were recorded for analysis.

System Boards

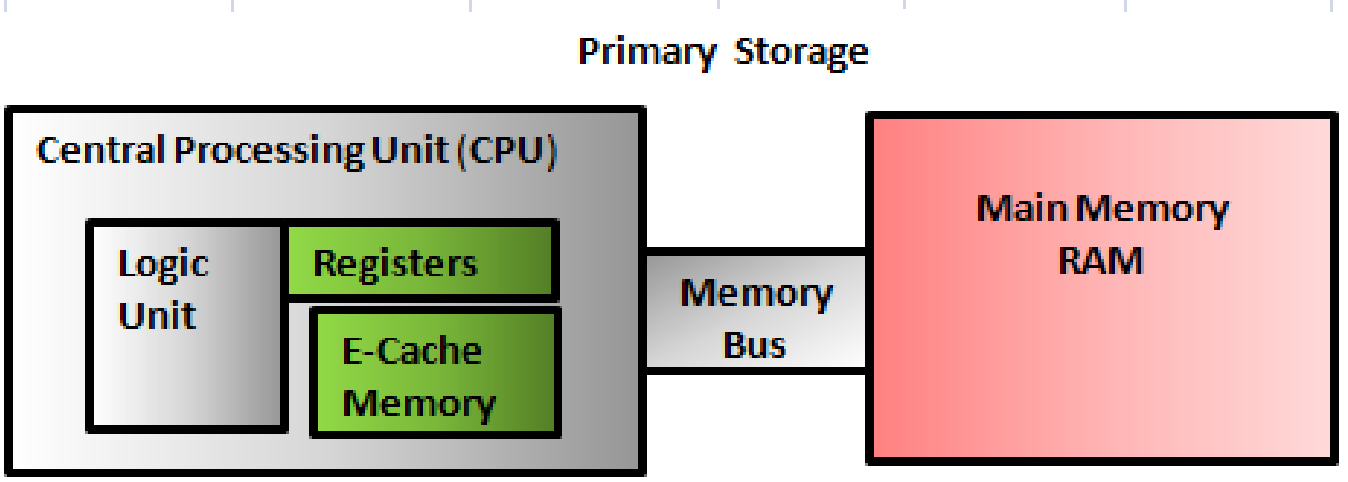
- Typical system board
  - Approximate size is 2'x2' and weight ~30 lbs.
  - Cost ~$100,000 per board
- Boards returned for analysis to factory
  - Damage in transit was not uncommon.
  - After analysis, over 95% of returned boards were classified as no trouble found (NTF). Remaining 5% were often determined to be damaged in transit.

Actions to Identify Cause of Failures

- Extensive stressing and testing of new and returned boards in systems
- Physical failure analysis of returned boards
- Replacement with new boards
- Observational visits to customer sites
- Field environmental measurements
- Data logging activity (Explorer runs)
- Consultation with suppliers
- Frequent review and update meetings of teams of engineers and management

Failure Mode: E-Cache Parity Errors

Months of work identified parity errors in e-cache (external, L2) SRAMS as problem location but determining exact cause was elusive.



Source: Wikipedia: "Computer Data Storage"

Slow Progress in Isolating Causes

- Failures continued in the field
- Service engineers worked diligently to diagnose failures and restore systems
- Costs of field repairs escalated
- Customers demanded prompt resolution

Data Collection Team

- A team was formed to collect data on field failures
- Data from major customers' datacenters were collected
- The importance of acquiring time dependent field data was emphasized

Data: Random Field Behavior?

- Some customers experienced no failures
- Other customers saw high levels of failures for the same systems
- A customer in a concrete vault below ground level saw no failures
- Other customers in high altitude environments (observation stations) had more frequent failures
- Was altitude or barometric pressure a factor?

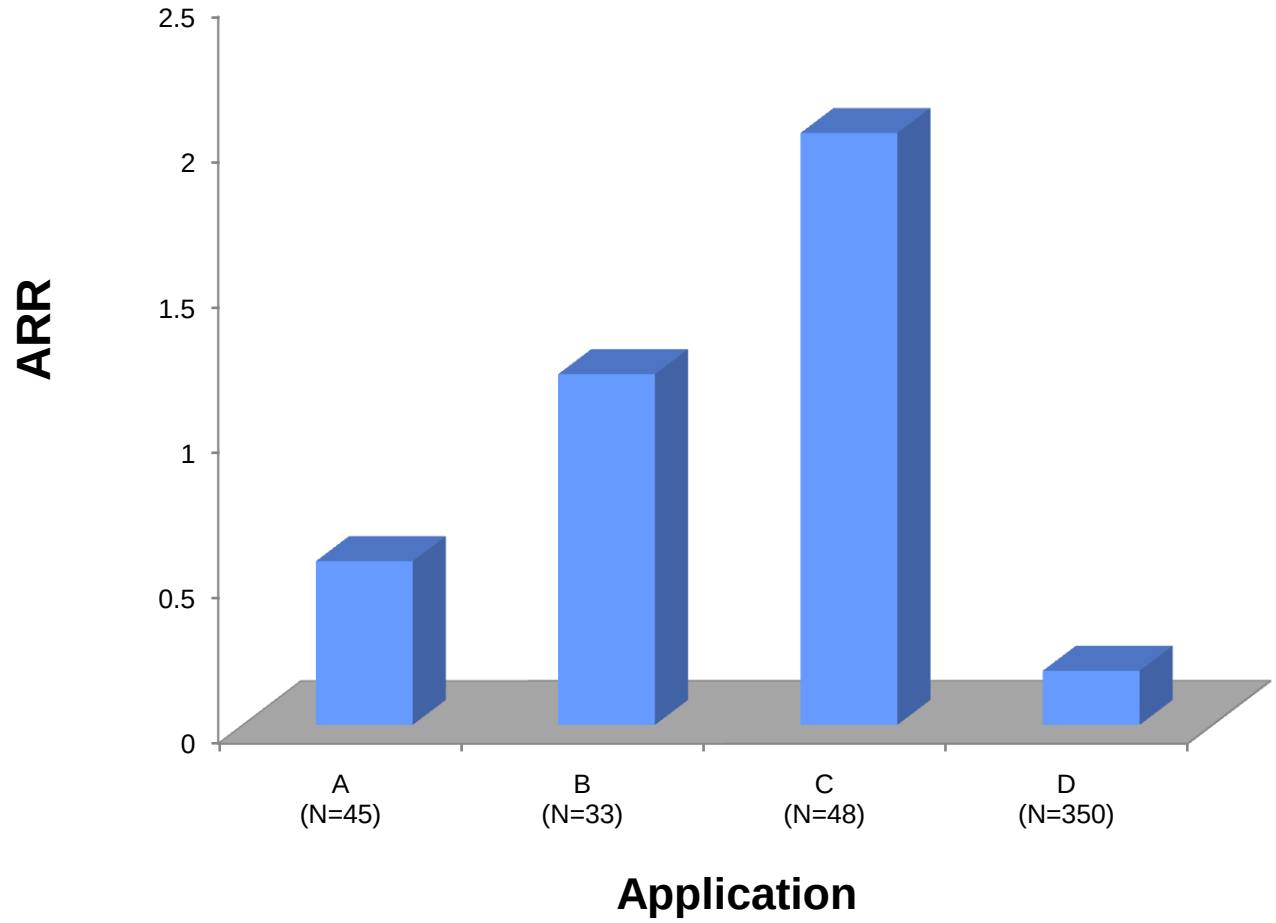
Datacenter Field Failures

In the **same** datacenter, customers running **different applications** on **identical** systems experienced widely different failure rates, that is, rate of occurrence of failures (*ROCOF*).

Example of Application Dependence

Single Datacenter, 476 Identical Systems

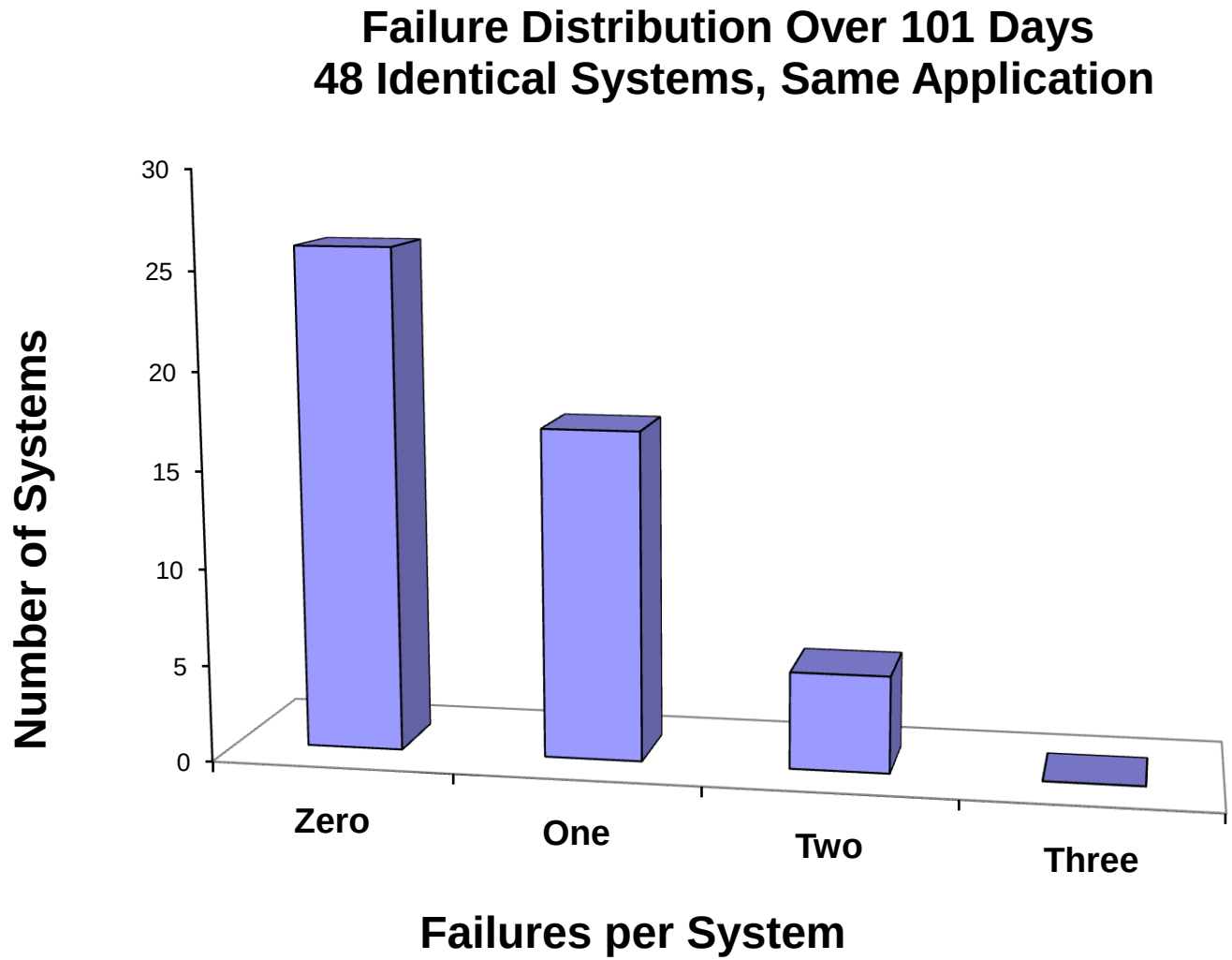
Annualized Failure Rates Versus Application



Distribution of Failures Across Systems

In the **same** datacenter, for **identical** systems running the **same applications** over the **same time period**, there could be systems with **no** failures, some with **single** failures, and some with **multiple** failures.

Example of Failure Distribution



Statistical Analysis and Modeling

- Could statistical analysis and modeling of the data provide any insights into the cause?
- How could the application dependence be explained?
- Could the model agree with field behavior and predict future failures?
- Could we explain the distribution of failures across systems in a datacenter?

Reliability Measures for Repairable Systems

- Key measures
  - Times between repairs (interarrival times)
  - Number of repairs over time
- Reliability is a function of many factors:

Basic system design	Types of repairs
Operating conditions	Quality of repairs
Environment	Materials used
Applications	Suppliers
Software robustness	Human behavior

System Age

- System age is the total running hours, that is, the elapsed time, on a system starting at installation turn-on. Also called power-on hours (**POH**) or operating hours.
- Often called the **uptime**
- Distinguish from times **between** failures (interarrival times) and device-hours or unit-hours.

Sequence of Failure Times

Key property of repairable systems:

- **Failures occur sequentially in time**.
- If the times between successive failures are getting **longer**, then the system reliability is **improving**.
- Conversely, if the times between failures are becoming **shorter**, the reliability of the system is **degrading**.
- Thus, the ***sequence*** of system failure times can be very important.
- If the times show no trend (relatively **stable)**, the system is neither improving or degrading, a characteristic of what is called a **renewal process**.

Renewal Process for a System

Critical question:

- For a renewal process, the times between failures are **independent and identically distributed (*i.i.d.*)** observations from a single population. How can we verify such an assumption?
- In a renewal process, there is no trend.
- For a system, restoration to "**like new**," such as replacement of a failed component with one from same population, implies a **renewal process** (*i.i.d.*).
- The assumption of a renewal process must be checked for validity.

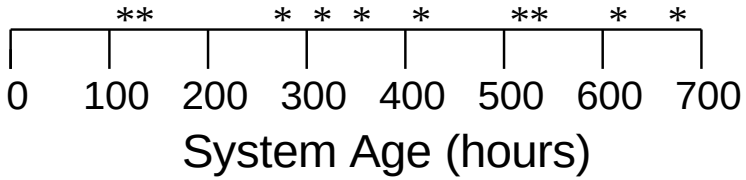
Analysis of a Renewal Process

Consider a **single** system for which the **times to make repairs are ignored.**

Ten failures are reported at the system ages (in hours):

106, 132, 289, 309, 352, 407, 523, 544, 611, 660.

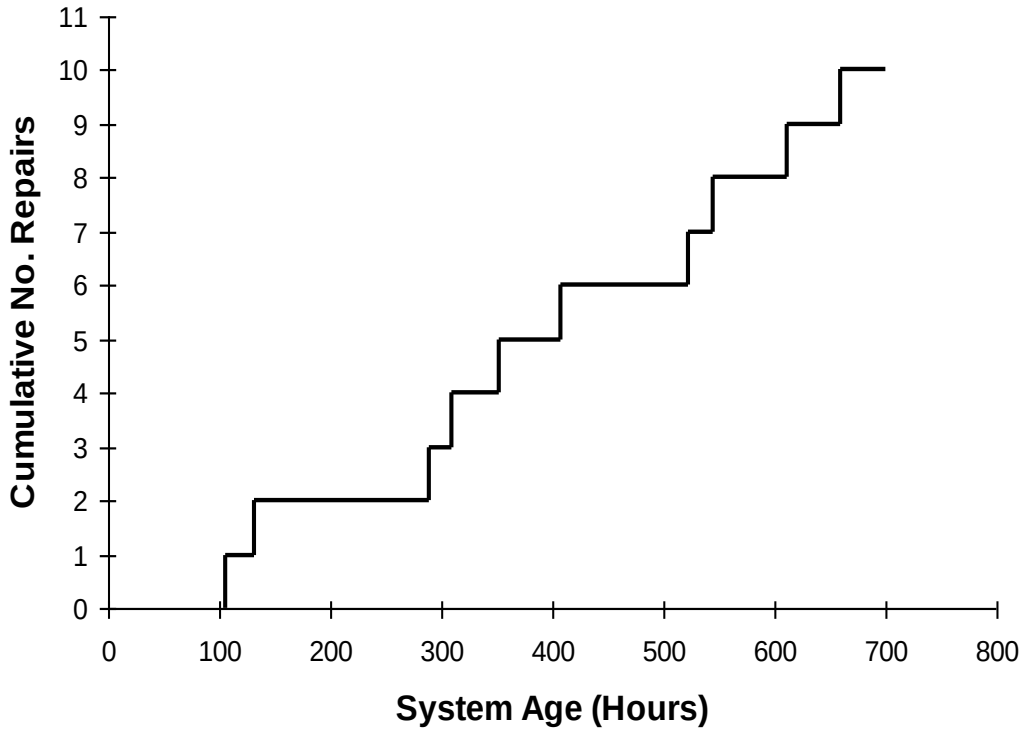
The pattern of repairs is



Analysis of a Renewal Process

A very revealing and useful data graph is called the **cumulative plot**: the **cumulative number** of repairs, ***N(t)***, is plotted against the system **age**, ***t***, at repair.

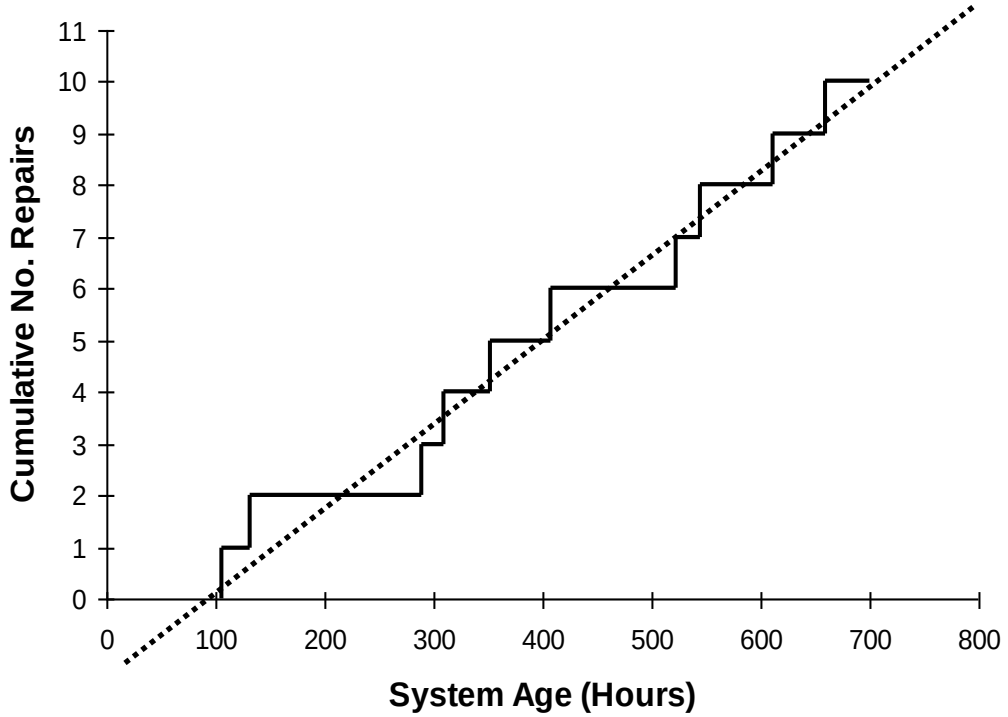
For the renewal data, the cumulative plot is:



Analysis of a Renewal Process

Under a renewal process, the times between failures are i.i.d., that is, from a single population having a ***constant*** mean time between repairs (average or ***MTBF***).

Consequently, the cumulative plot should appear to follow a **straight line**.



Data Limitations

- Unfortunately, age related data is typically not available for systems.
- Field reliability data is often presented in terms of a **mean time between failure**, ***MTBF***.
- It is much easier to count the numbers of failures in a given time period (e.g., one month) for a group of systems operating for that time period than it is to obtain the system installation dates to measure age and the time dependent history of the ages of failures.
- Are there other ways to model the field behavior?

Enhancing Graphical Analysis

Also, the cumulative plot alone does not tell us all we'd like to know.

How **precise** is the estimate of $N(t)$?

What is the **distribution** of the **number of repairs** for systems at time ***t***?

What is the average number of repairs $M(t)$ at time t. Called the ***mean cumulative function, MCF***.

What is the **distribution** of the ***time*** to a specific number of repairs?

Graphical analysis is important, but we need additional analytical and modeling tools.

Importance of a Model

Helps us to understand current results.
Allows for prediction of future behavior.
May prevent reaction to noise.
Helps identify potential failure mechanisms.

George Box:

“All models are wrong. Some are useful.”

What models are useful for repairable system?

Key Analysis Variables

Two variables are of key interest:

M(*t*) the **mean number of repairs** by time *t,* that is, the ***MCF***

T(*k*) the **time to reach the *k*th failure**

For a renewal process, *M*(*t*), the ***MCF***, is also called the **renewal function**, which is the **expected (or average) value** of *N*(*t*), the number of repairs by time ***t*** for a single system.

Renewal Process: Single System

For a renewal process, the **single distribution** of **failure times between repairs** defines the expected pattern of repairs.

Let X_i denote the **interarrival time** between the *i*th and the (*i*-1) repair.

The **time to the *k*th repair** can be written as the sum of *k* interarrival times

$$T(k) = \sum_{i=1}^{k} X_i$$

For example, if the first three interarrival times are 100, 150, and 75 hours, then the time to the third repair is 100+150+75 = 325 hours.

Knowing the **probability distribution** (***pdf***) of X_i, we can theoretically find distributions for ***N*(*t*)** and ***T*(*k*)** along with ***M*(*t*)** and the renewal or recurrence rate (ROCOF) ***m*(*t*)** = $dM(t)/dt$

Poisson Model for Renewal Process

Suppose the interarrival times X_i are *i.i.d.* with **exponential** probability density function (***pdf***) having **constant** failure rate intensity λ, that is,

$$f(x) = \lambda e^{-\lambda x}$$

Then, we can show that $N(t)$ has a **Poisson distribution** with **constant renewal rate** intensity λ. The **expected number** of repairs in time t is λt.

Note that λ is a **rate** (i.e., repairs/time) that is multiplied by time ***t*** to give the **number** of repairs by time ***t***.

Homogeneous Poisson Process Model (*HPP*)

Consequently, the probability of observing **exactly** $N(t) = k$ failures in the **interval** $(0,t)$ is given by the Poisson distribution

$$P\left[N(t)=k\right]=\frac{\left(\lambda t\right)^{k} e^{-\lambda t}}{k!}$$

We call this renewal process for which the interarrival times are exponentially distributed a ***homogeneous Poisson process*** (***HPP***).

MTBF for *HPP*

For a *HPP*, the **mean time between failures** (***MTBF***) is constant and

$$MTBF = \theta = 1/\lambda$$

The **expected number** of repairs in time ***t*** is

$$M(t) = \lambda t = t/\theta.$$

The **mean time to the *k*th** repair is

$$k/\lambda = k\theta.$$

HPP in Terms of *MTBF*

We can rewrite the Poisson distribution for the HPP in terms of the *MTBF*, θ:

$$P\left[N(t)=k\right]=\frac{\left(t/\theta\right)^{k}e^{-t/\theta}}{k!}$$

Example: The *MTBF* is 10,000 hours. What's the probability of one failure in 3 months?

The expected number λt is

t/θ = (91days x24hrs/day)/10,000 hrs = 0.218

The probability of exactly one failure is

$$P\left[N(t)=1\right]=\frac{(0.218)e^{-0.218}}{1!}=0.0878$$

HPP for Multiple Systems

By **multiplying** the calculated *HPP* Poisson distribution **probabilities** for a given failure rate or *MTBF* by the **number** of systems, we can estimate the expected **distribution of failures** across many similar *HPP* systems.

Examples of Questions on Repair Distributions for *HPP*

The *MTBF* is 10,000 hours. Assume a *HPP*. There are 100 servers in use. After 60 days:

What's the expected number of systems with no repairs?

What's the expected number of systems with exactly one repair?

What's the expected number of systems with exactly two repairs?

What's the expected number of systems with more than two repairs?

Case Study *HPP*

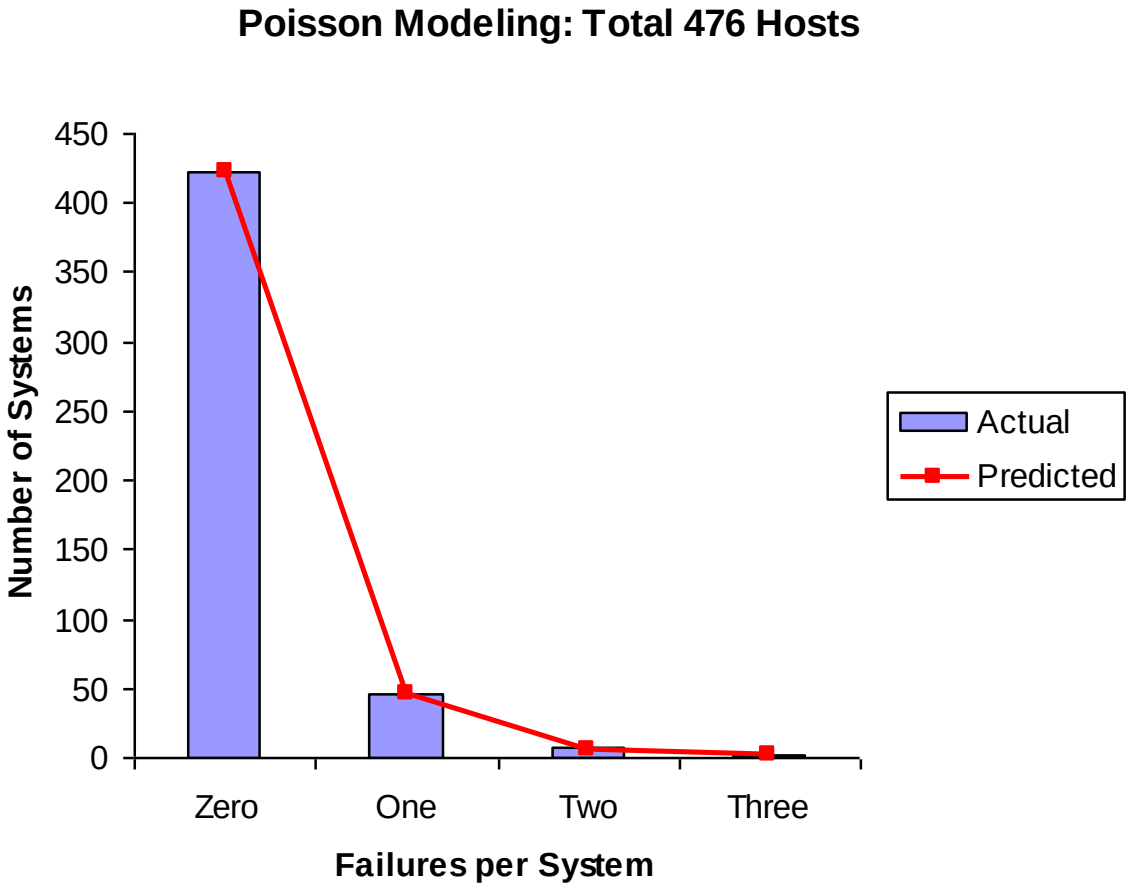
There were a total of 476 hosts in a large datacenter.
For confidentiality, the specific customer, type of system (large), and applications are not identified.

By determining an overall failure rate or *MTBF* over the previous few months, we checked for the suitability of an *HPP* model that could predict over the next 101 days how many of the 476 systems would have no failures, one failure, two failures, and so on. This prediction was then compared against actual failure counts across all systems.

The model was in excellent agreement with observed results, confirming the *HPP*.

Model Confirmation

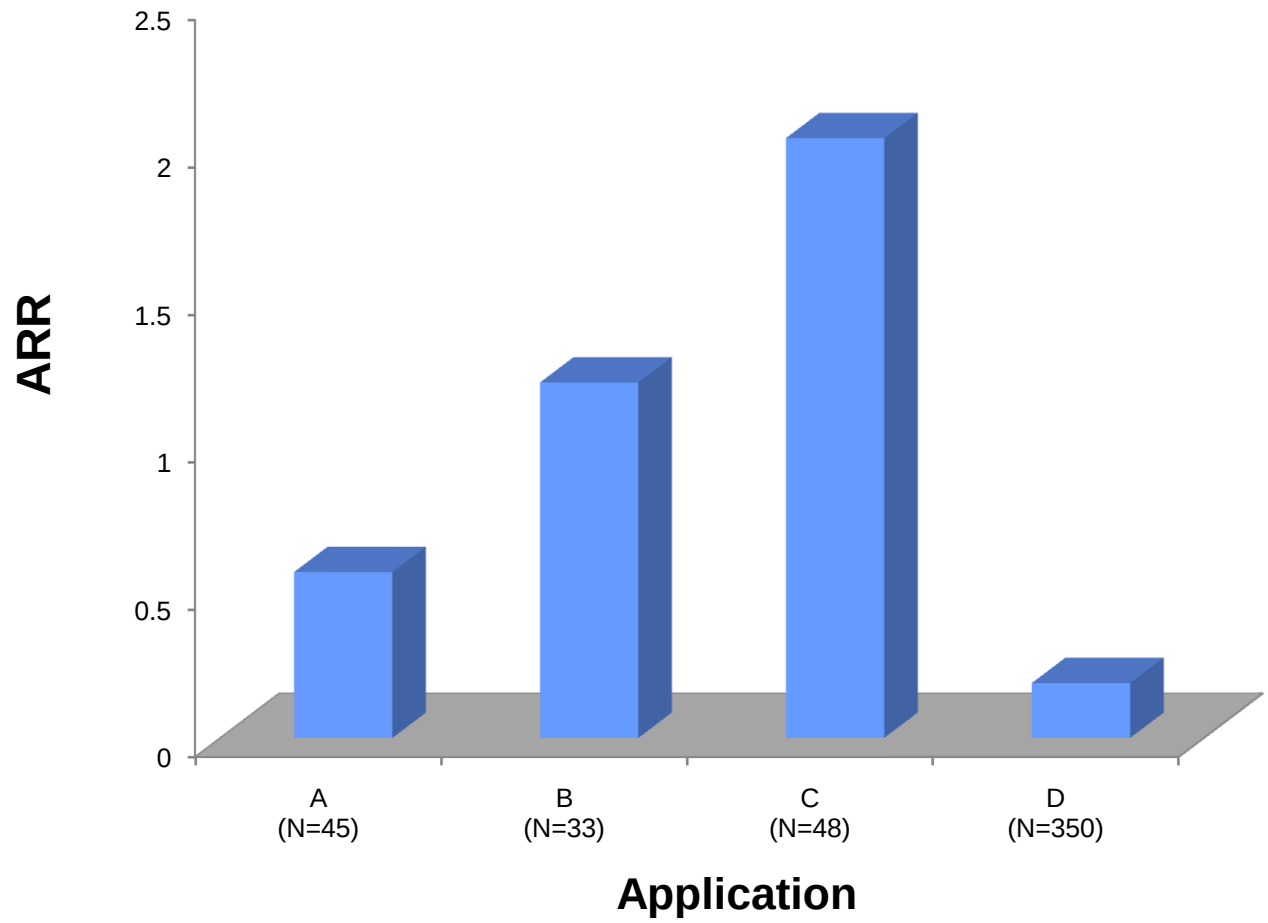
Comparison of Poisson Distribution Predictions Versus Actual Failures for a 101 Day Period



Application Dependency

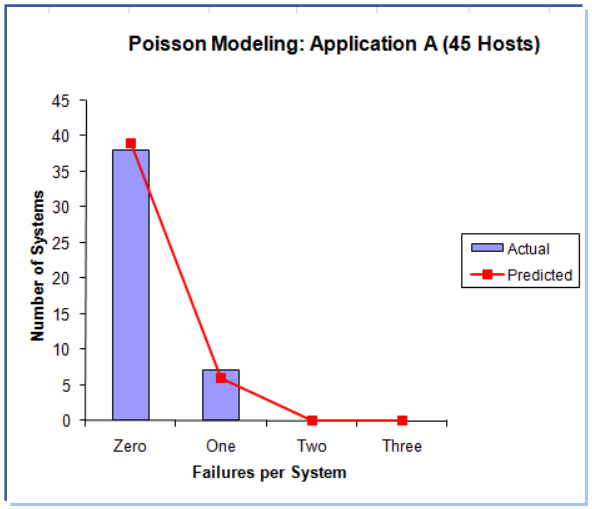
Single Datacenter, 476 Identical Systems

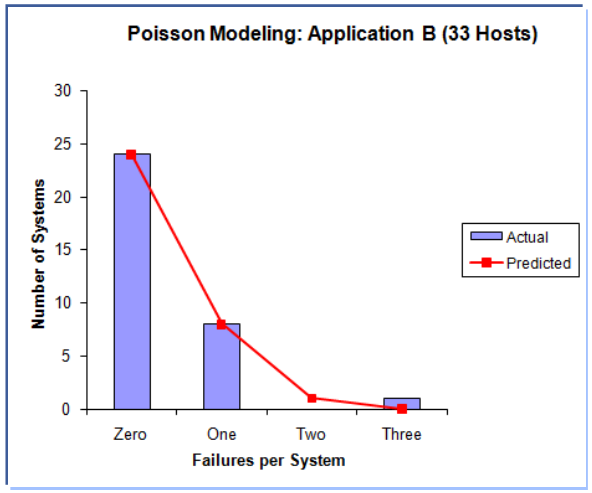
Annualized Failure Rates Versus Application

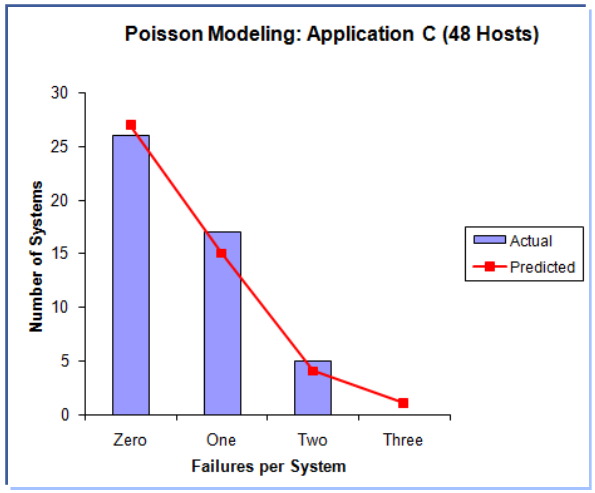


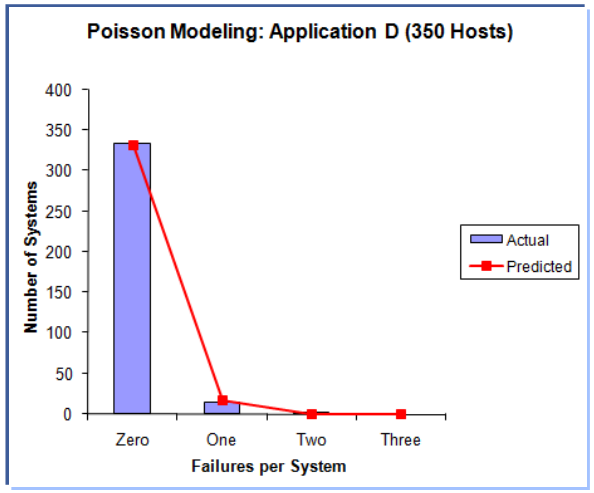
Application Modeling to Poisson Process

Each application was checked against Poisson distribution predictions. Agreement was excellent.



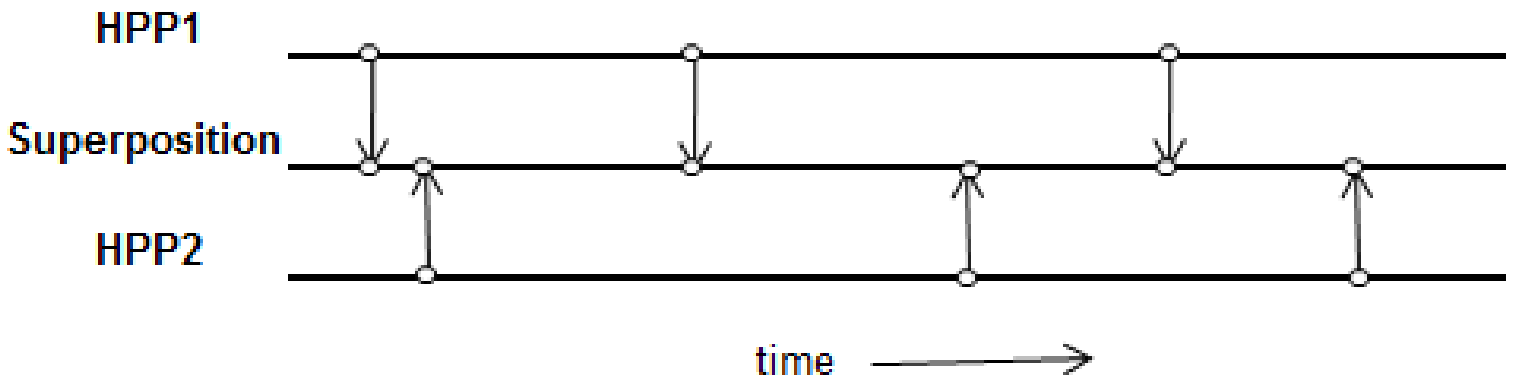






Superposition of Poisson Processes

Consider two independent Poisson streams with separate rates λ_1 and λ_2. If an event occurs whenever an event in either of the two Poisson process occurs, we have a superposition process with rate $\lambda = \lambda_1 + \lambda_2$.



The probability of the next event coming from stream 1 rather than stream 2 is $\lambda_1 / (\lambda_1 + \lambda_2)$.

Superposition of Poisson Processes

The superposition of *N* Poisson processes with intensities $\lambda_1, \lambda_2, \cdots, \lambda_N$ is a Poisson process with intensity $\lambda = \lambda_1 + \lambda_2 + \cdots + \lambda_N$.

Failure Rate Estimates for Poisson Processes

Over a period of 101 days, there were a total of 63 failures among the 476 systems in the datacenter. The overall annualized recurrence rate (***ARR***) is estimated as

$$\lambda = \frac{63}{476}\left(\frac{365}{101}\right) = 0.48 \text{ per system}$$

Similarly, we can estimate the *ARR* separately for each application.

Application	#Hosts	Observation Days	Observation Hours	Total Fails	Device Hours	ARR
A	45	101	2424	7	109080	0.56
B	33	101	2424	11	79992	1.20
C	48	101	2424	27	116352	2.03
D	350	101	2424	18	848400	0.19
Total	476	101	2424	63	1153824	0.48

Superposition *ARR* Estimate

We can also estimate the overall *ARR* by using the weighted superposition formula for a *HPP*

$$\lambda = \frac{\sum_i \lambda_i N_i}{\sum_i N_i} = \frac{0.56 \times 45 + 1.20 \times 33 + 2.03 \times 48 + 0.051 \times 350}{476} = 0.48$$

This result matches the previous estimate for the overall *ARR* for the 476 servers, in agreement with the *HPP* superposition model.

Two Sample Test of *MTBF*s

- Tests for the statistical significance of the different *MTBF*s were performed*. All differences were significant.
- Test shown is for the smallest *MTBF* differences.

Enter the first MTBF:	7,272
Enter the number of failures:	11
Enter the second MTBF:	4,309
Enter the number of failures:	27
Enter desired confidence level as a fraction (e.g., 0.90 for 90%):	0.878
The ratio of the first MTBF to the second is:	1.687630541
The LCL is:	1.001380266
The UCL is:	2.844171129
Conclusion (significant difference or insignificant):	significant

* "An EXCEL Add-In for Comparing Two Exponential Distributions", D. Trindade, *Proceedings of the Joint Statistical Meetings* (2000)

Consequences and Implications

Since the results were consistent with a *HPP*, the implication was that the failure behavior for any system in the datacenter derived from a **renewal process** with a **constant failure rate**.

Constant failure rates result from a **constant source**.

There was **no physical damage** to the SRAM by the cause. The “good as new” assumption for a renewal process seemed valid.

Failure rates were also determined to vary with altitude.

This confirmed that only plausible source was **radiation from cosmic rays** causing single bit parity errors in the e-cache memory. Without corrective actions, failures would occur and panic the systems.

Efforts to Mitigate the Problem

E-Cache scrubbing via software

- Checking e-cache for parity errors and invalidating e-cache entry with error
- Scrubbing was done periodically, took cycles, and could not always catch error before load on system resulted in panic

Physical Mechanisms

The radiation environment

- **Alpha** particles
- **High energy** cosmic rays
- **Low energy** cosmic rays and ^{10}B fission in boron-doped phosphosilicate glass (BPSG) dielectric layers of ICs

Factors impacting SER

- Complexity
- Density
- Lower voltage
- Higher speeds
- Lower cell capacitance

The susceptibility to soft error rates for DRAM and SRAM has increased with reduced dimensions (higher densities) and lowered operating voltages of advancing technology.

Read-Write Activity

The server writes to e-cache memory. Memory in e-cache can be saved to permanent memory. If a cosmic ray causes a parity error to occur in e-cache and an attempt is made to read data in e-cache or to write it to main memory, the parity error will be detected and the system will panic to prevent data corruption.

Scrubbing the e-cache to correct single bit errors in e-cache before the errors are written to memory was a band-aid approach. A much more effective solution was to incorporate **mirroring**, where every byte is duplicated and stored in two locations in *SRAM* along with a parity checker built into the *SRAM*.

(Note: The equally effective alternative of replacing parity protection with single-error correction, double-error detection error correction code, "***SECDED ECC***", was rejected as it would have required a change to the processor's pipeline.)

Explaining Application Dependence

If an application writes often to memory but does not read frequently, an e-cache error can be overwritten before a read cycle sees the error. Imagine an application updating minutes used by a cell phone user. Consequently, the failure rates will be low.

If an application reads frequently, then e-cached errors will be detected quickly and cause failures. The failure rates will be high.

Best Practices

Instead of removing a failed board, the simplest action was simply to **reboot the system**. No physical damage had occurred and the probability of a hit by a cosmic ray was purely random.

In addition, the costs of replacing boards and subsequent damage to the boards or systems (e.g., bent pins) could be avoided.

Spreadsheets were sent to the field for the service engineers to do the model fitting for any customer and illustrate the model consistency.

Best Practices: Spreadsheet to Field

Distribution of Number of Failures Per Machine for Poisson Process for Specified Time Period (Repairable Systems)

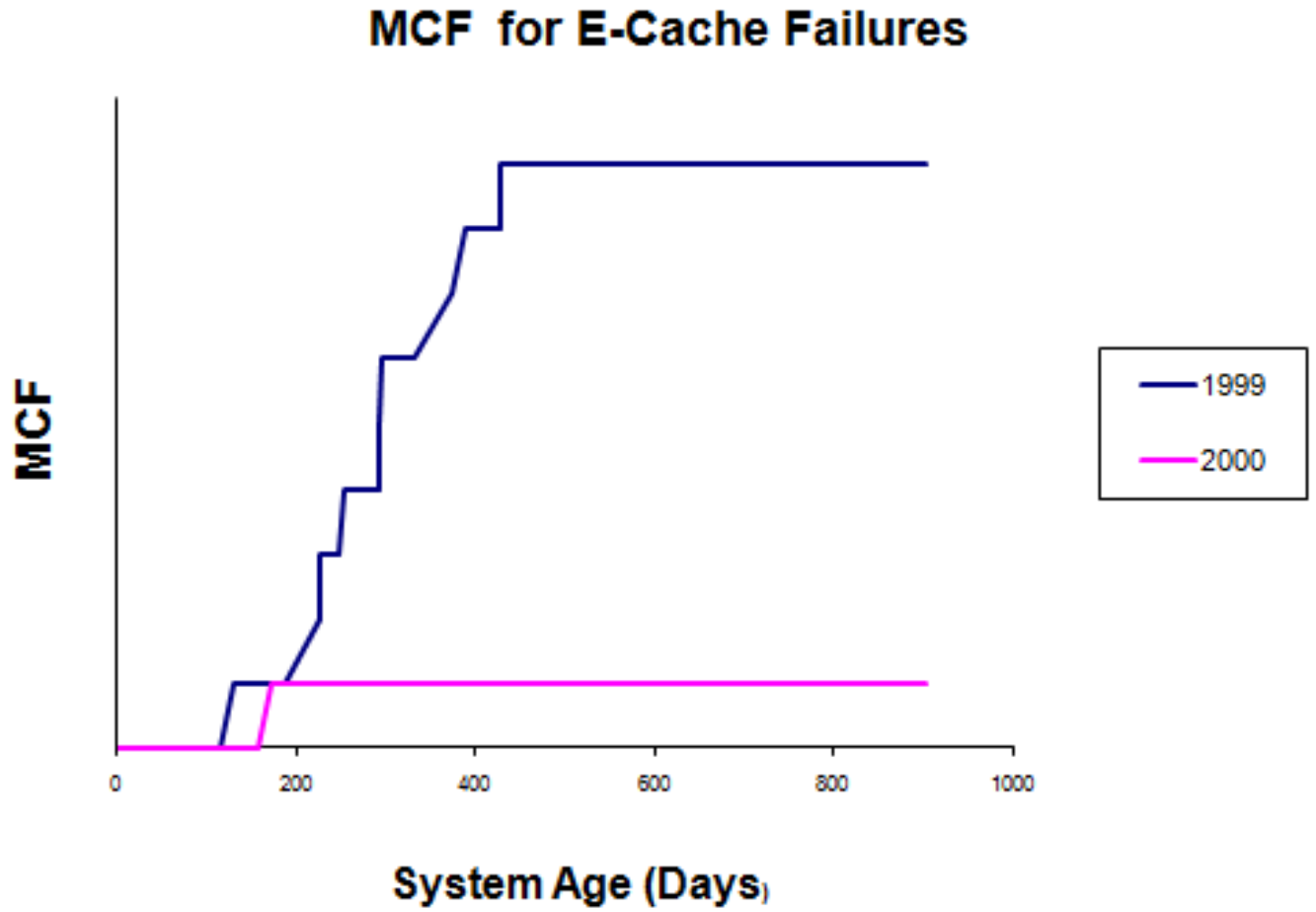
Entry Description	*Enter Value*
MTBF (hrs)	10,000
Time in Days	30
Number of Machines	1,000

Time (hrs)	720
Time/MTBF	0.072

Poisson Distribution Calculations		
X fails per machine	Probability of X	No. Machines with X Fails
0	93.1%	931
1	6.7%	67
2	0.2%	2
3	0.0%	0
4	0.0%	0
5	0.0%	0

Confirmation

Introducing mirrored *SRAMs* into systems stopped the failures.



Summary

- Field failures represent significant inconvenience to customers.
- Field failures remediation efforts are costly to system manufacturers.
- Complex systems make identification of causes difficult and challenging.
- Statistical analysis and modeling can provide valuable insights into causes.
- Undetected and uncorrected soft errors are a significant factor in system reliability, but there are approaches to alleviate the problem.

Where to Get More Information

- Google “soft error reliability” for a wealth of information on the topic.
- Search Wikipedia under “soft error”, “CPU cache”, “cosmic rays”.
- *SER-History, Trends, and Challenges* by J. Ziegler and H. Puchner, Cypress Semiconductor Corporation (2004)
- “Radiation-Induced Soft Errors in Advanced Semiconductor Technologies”, R. Baumann, *IEEE Trans. On Device and Materials Reliability*, Vol. 5, No. 3, September 2005
- For statistical analysis and modeling of reliability data see *Applied Reliability*, 2nd ed. by P. Tobias and D. Trindade, Chapman & Hall/CRC (1995)
- Additional references on modeling and data analysis at www.trindade.com/publications.html

David C. Trindade, Ph.D.

- Author's Information:
  - Distinguished Principal Engineer
  - Sun Microsystems, Inc.
  - Email: david.trindade@sun.com
- For biography, see www.trindade.com

Questions

Thank you for your attention.

Do you have any questions?