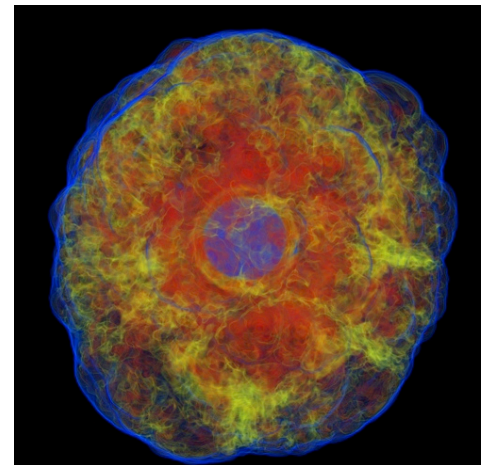
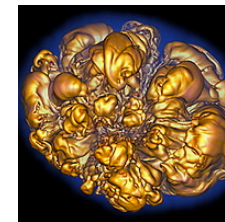
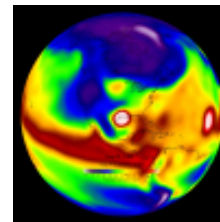
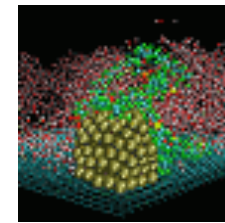
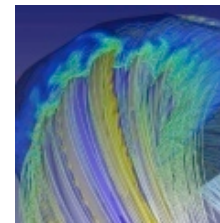
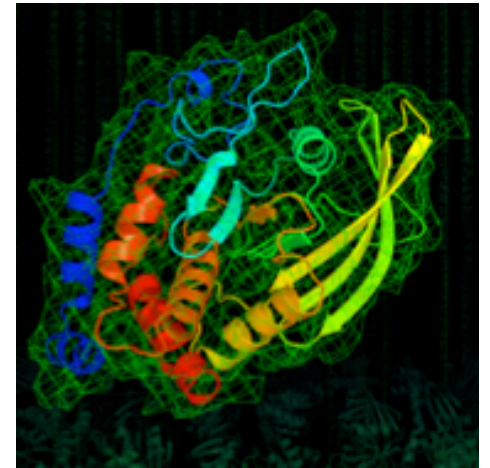
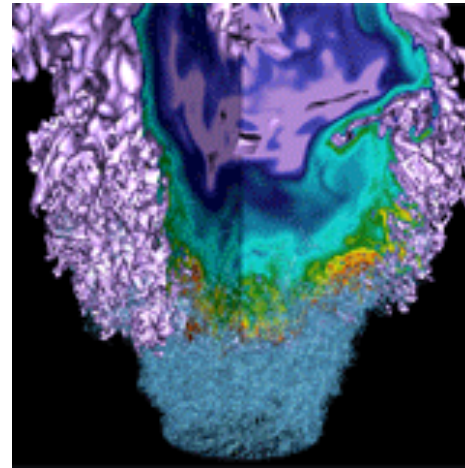


Memory Errors in Modern Systems

The Good,
The Bad,
And The Ugly



Vilas Sridharan,
Nathan DeBardeleben,
John Shalf,
Sean Blanchard,
Kurt B. Ferreira,
Jon Stearley,
Sudhanva Gurumurthi

December 9, 2015

Acknowledgments

THIS WORK WAS SUPPORTED BY DOE ASCR AND ASC

AMD WAS SUPPORTED BY DOE FAST FORWARD AND AMD RESEARCH



• Collaborators

- Vilas Sridharan, AMD Research, Advanced Micro Devices Inc.
- Nathan Debardeleben, Ultrascade Systems Research Center, Los Alamos National Laboratory
- Sean Blanchard, Ultrascade Systems Research Center, Los Alamos National Laboratory
- Sudhanva Gurumurthi, AMD Research, Advanced Micro Devices Inc.
- Jon Stearley, Scalable Architectures, Sandia National Laboratories
- Kurt B. Ferreira, Scalable Architectures, Sandia National Laboratories
- John Shalf, NERSC and CRD, Lawrence Berkeley National Laboratory

• Thanks to

- AMD Inc. who were central to the study
- Many folks at Cray Inc. for dumping error logs
- NERSC Systems staff, including Tina Butler and Jay Srinivasan



Motivation and background

- Reliability is crucial for large-scale systems
- Must confirm reliability models are accurate
- Use data from real systems to correlate to models

Cielo at Los Alamos National Lab



8-core AMD Opteron™ CPUs

8,944 nodes : 1,144,832 DRAM

DDR-3 DRAM, Chipkill-correct ECC

Hopper at NERSC / Lawrence Berkeley National Lab



12-core AMD Opteron™ CPUs

6,384 nodes : 817,152 DRAM

DDR-3 DRAM, Chipkill-detect ECC

Production systems
500M+ CPU socket-hours
40B+ DRAM device-hours

Sources of Failure

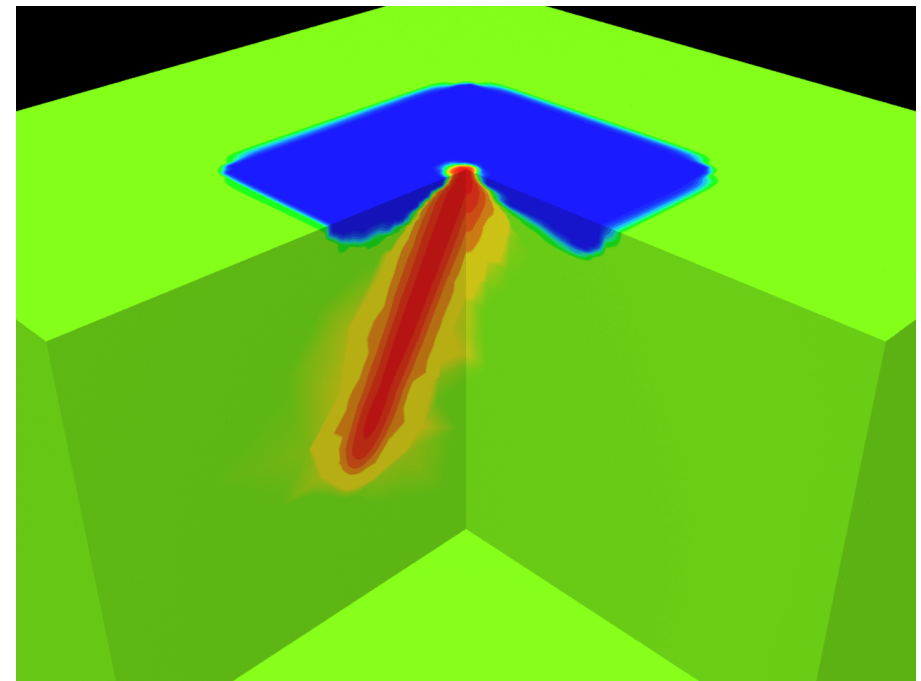
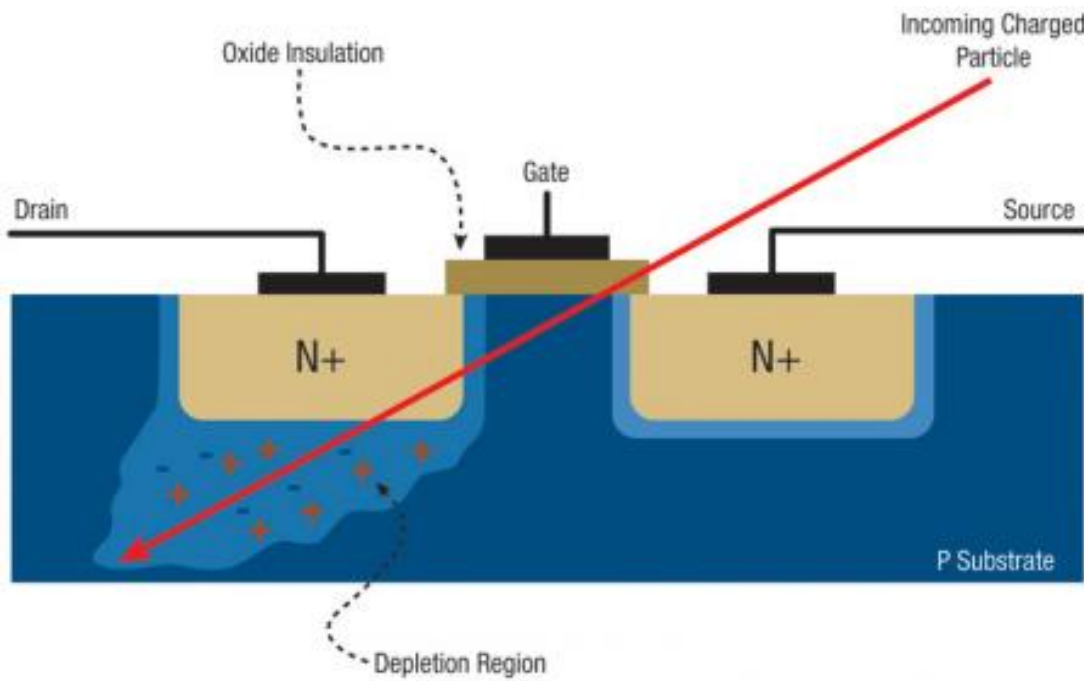
- Solder joints and other connector/mechanical failures
- Ephemeral bit upset is tied to energetic particle strikes
(probability is proportional to surface area exposure)



Software

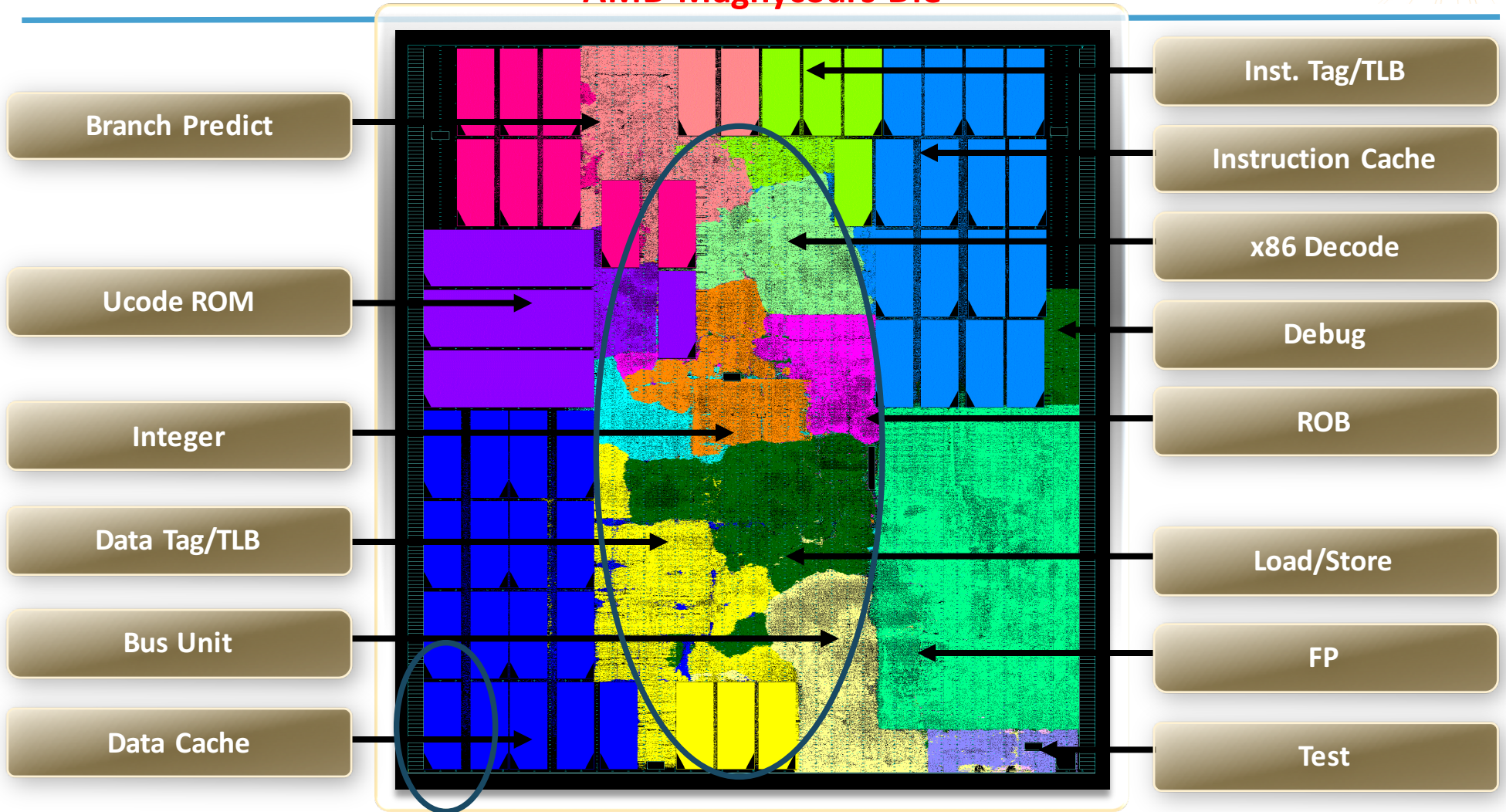
Why Focus on Memory

- HPC has an overwhelming obsession with compute
- But most of your computer is in fact memory
- And the probability of a bit upset is proportional to silicon surface area



SRAM Structures Consume a LOT of Area on Modern CPUs

AMD Magnycours Die

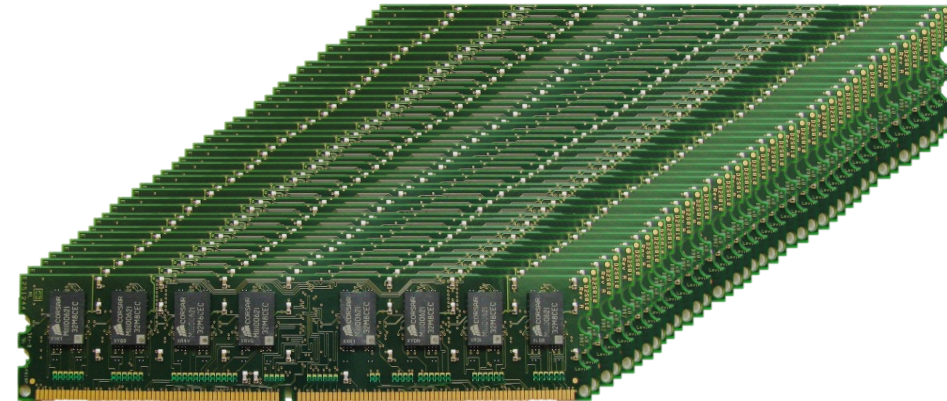


Each rectangle contains 16k bits of SRAM

100k+ flip flops and latches

DRAM Involves a lot of Discrete Components and even MORE Silicon Surface Area

- **Dynamic random-access memory (DRAM)**
 - Used for almost all computer main memory
 - Single-capacitor memory
 - Reads are destructive – must rewrite data after read (“precharge”)
 - Capacitors lose charge over time – must periodically rewrite data (“refresh”)
- **DRAM reliability is important today**
 - Laptop: O(1-10 GB) of DRAM
 - Petascale supercomputer: O(10-100 TB) of DRAM
- **DRAM reliability will be critical in the future**
 - Exascale: O(1-100 PB) of DRAM
 - In-package (die-stacked) DRAM

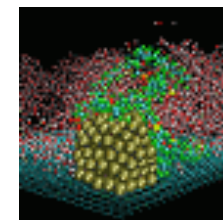
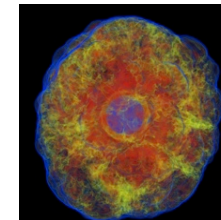
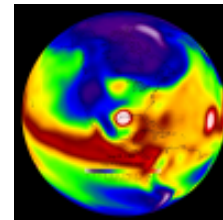
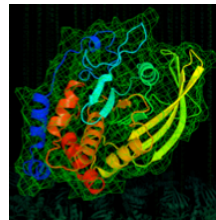
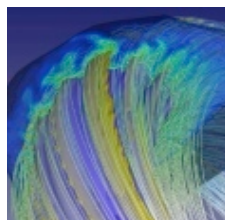
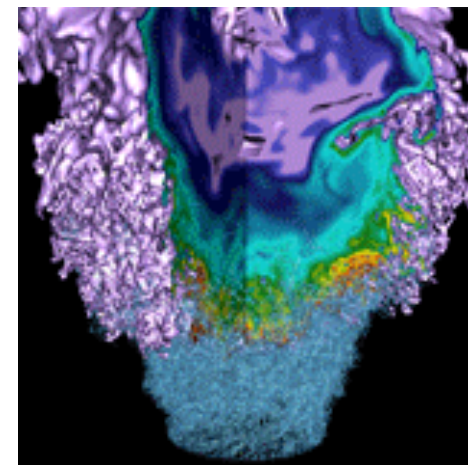


Motivation



- **Architectural & micro-architectural approaches to reliability**
- **To get it right, you must know the faults to expect**
- **This talk looks at faults collected in production systems in the field (validating the fault model)**

Terminology and Methodology



Failure Rates in Context

What is a FIT?



- A FIT is **ONE** failure per **Billion** hours of operation
- A FIT rate of **1** corresponds to....
 - 1 Billion hours of operation Failure every **115,000** years
 - For 8,944 nodes (Cielo): Failure every **12.8** years
 - For 71,552 DIMMs: Failure every **1.6** years
 - For 1,144,832 DRAM chips: Failure every **36** days
- **Real FIT rates (FIT rates for components on Cielo)**
 - Target socket FIT rate of **1000**: failure every **2.3** days
 - Target **DRAM** chip FIT rate of **35**: failure every **1** days

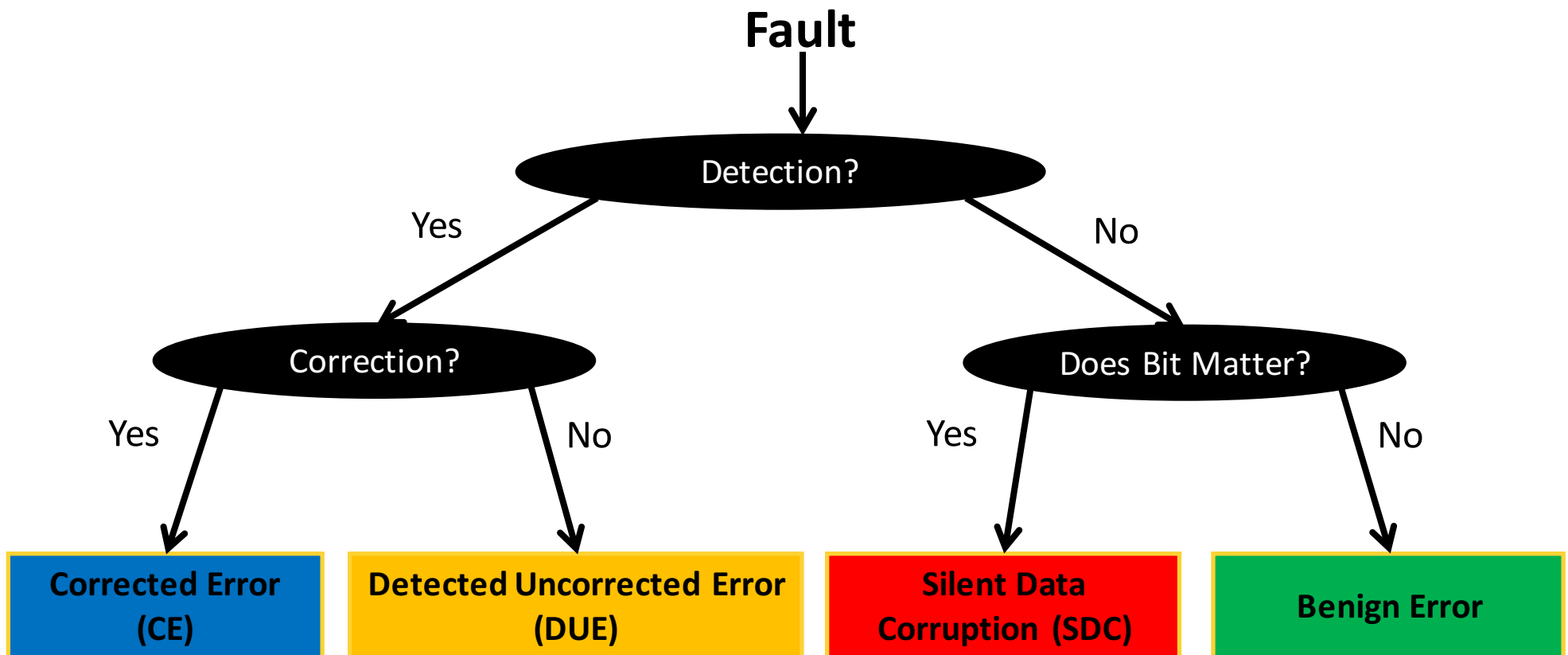
Terminology

- **Fault**
 - The underlying cause of an error, such as a stuck-at bit or high-energy particle strike
- **Transient fault**
 - Return incorrect data until overwritten
 - Random and not indicative of device damage
- **Hard fault**
 - **Consistently** return an incorrect value
 - Repair by disabling or by replacing the faulty device
- **Intermittent fault**
 - **Sometimes** return an incorrect value
 - Under specific conditions such as elevated temperature
 - Indicative of device damage or malfunction

Permanent faults

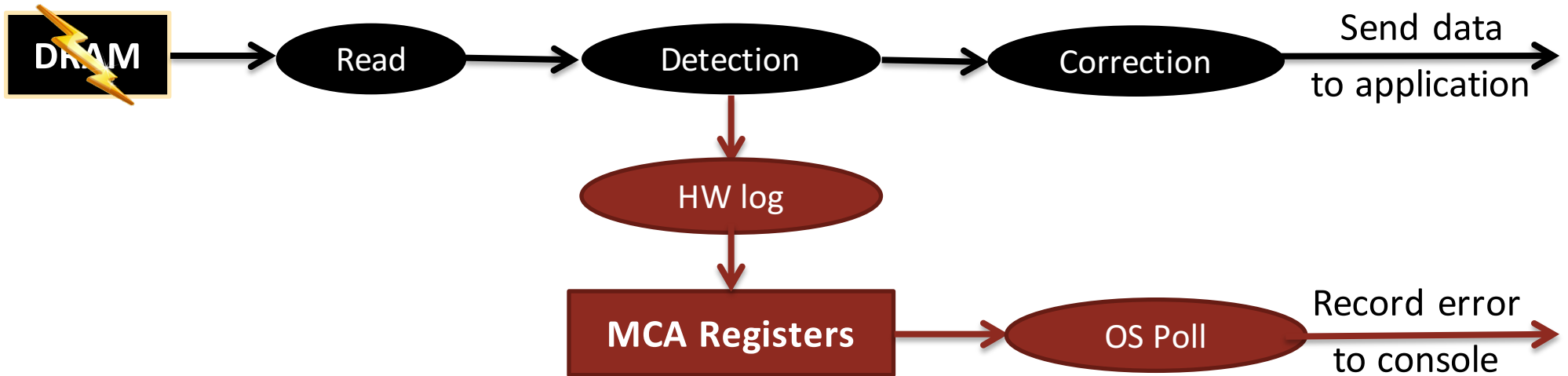
Terminology

- **Error:** *An incorrect state resulting from an active Fault, such as an incorrect value in memory*

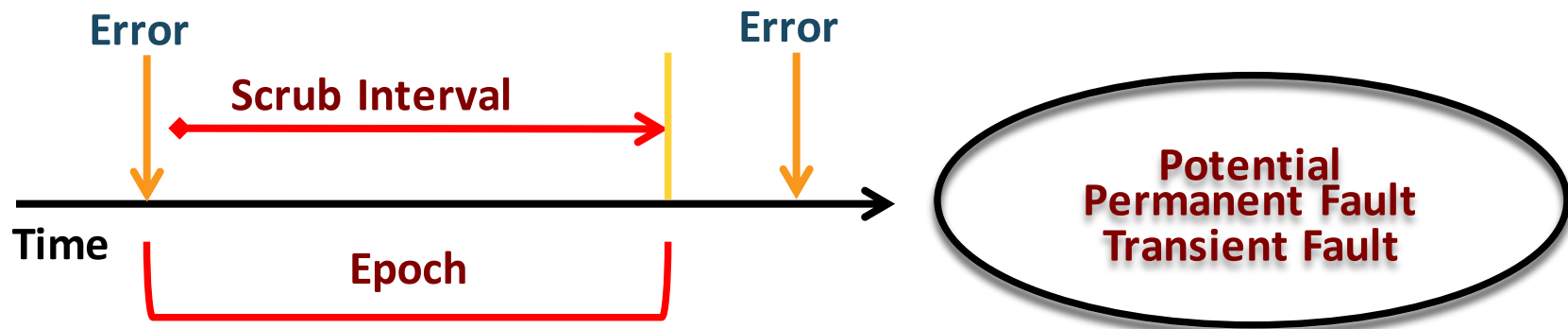


Methodology

- Data collection

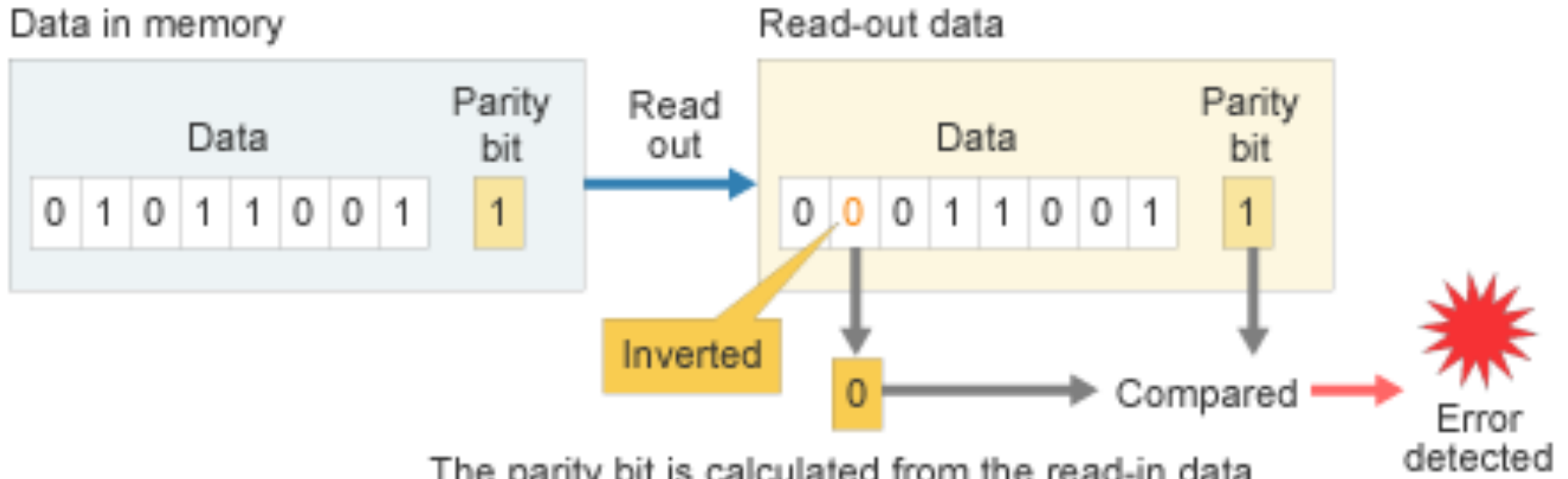


- Use presence of scrubber to coalesce errors into faults



Parity Protection

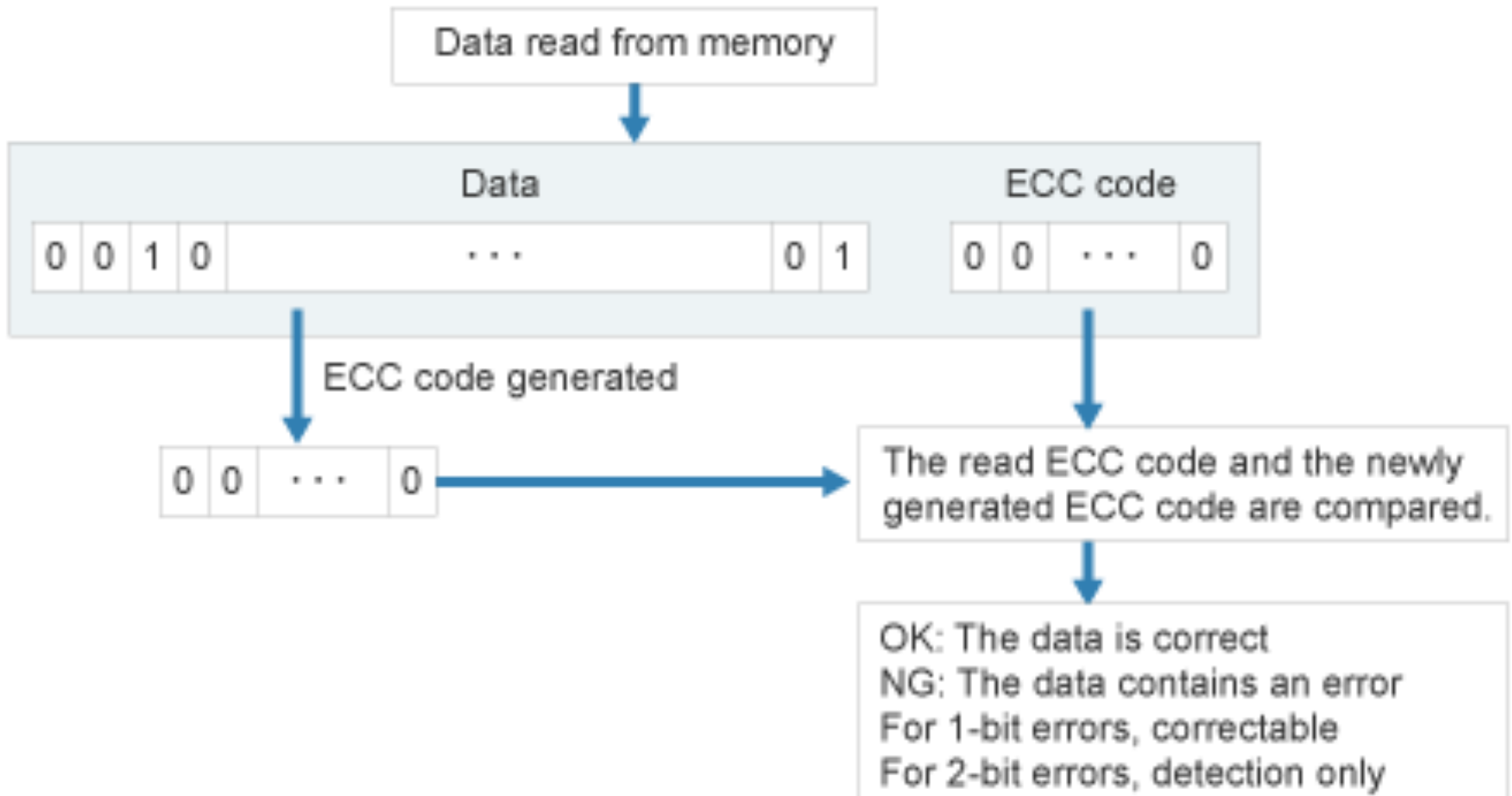
just detects errors



The parity bit is calculated from the read-in data.
Since the number of 1s is an odd number, the parity bit is "0".

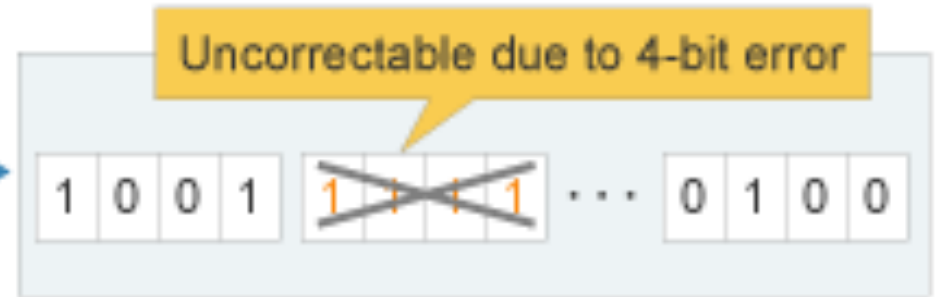
ECC (Error Correcting Code) Protection

(SEC-DED) Single Error Correct, Double Error Detect

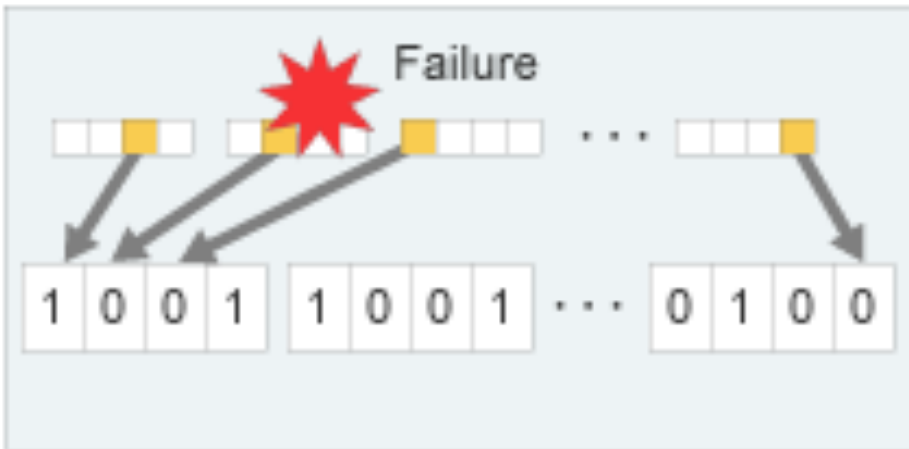


Interleaving Data for Protection (chip-kill)

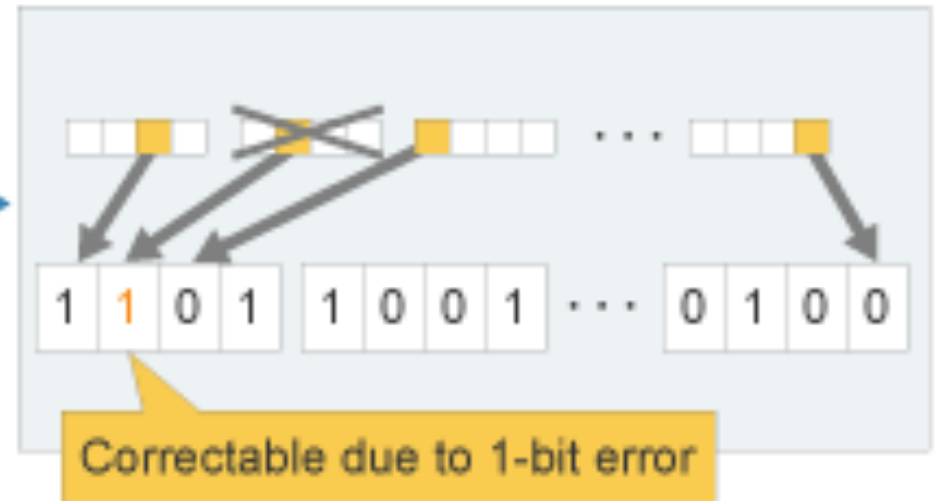
When data is stored in contiguous locations



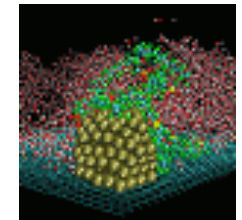
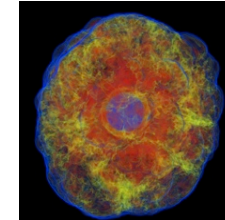
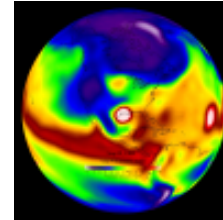
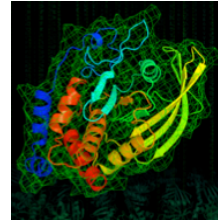
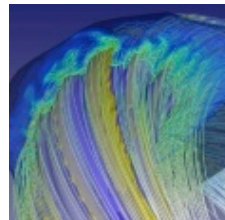
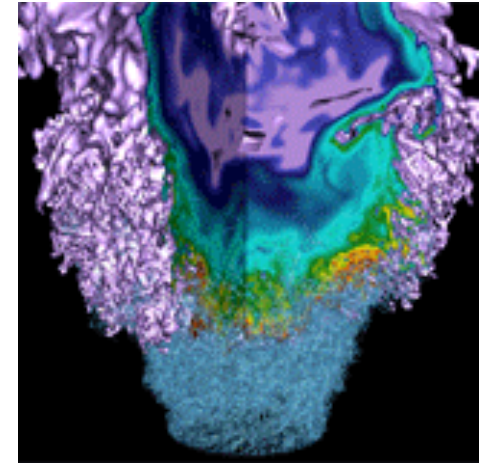
When data is stored in noncontiguous locations



When data is stored in noncontiguous locations



The Good



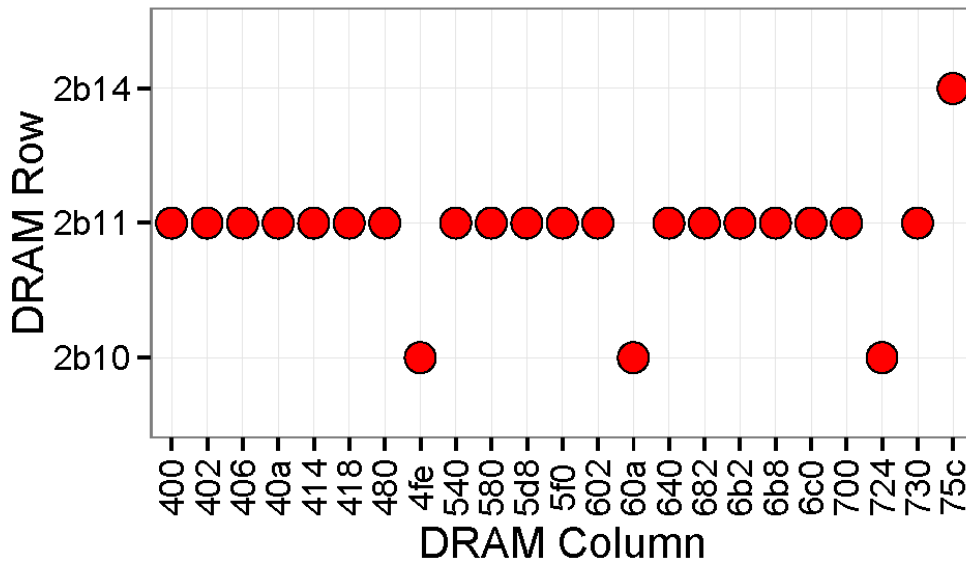
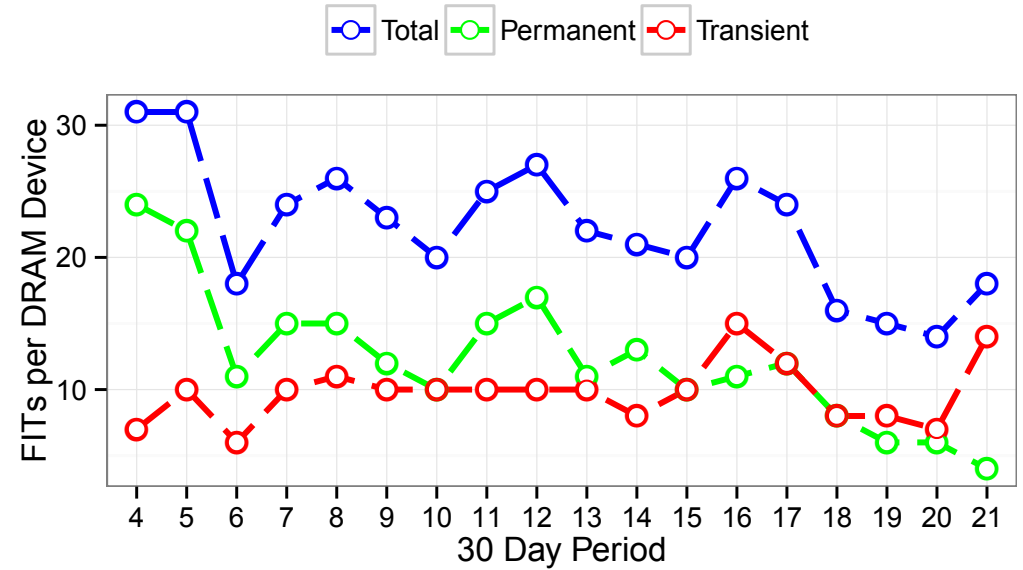
DRAM Fault Rates and MODES

▲ Fault rates

- Constant rate of transient faults
- Declining rate of permanent faults
- >50% permanent faults

▲ Fault modes

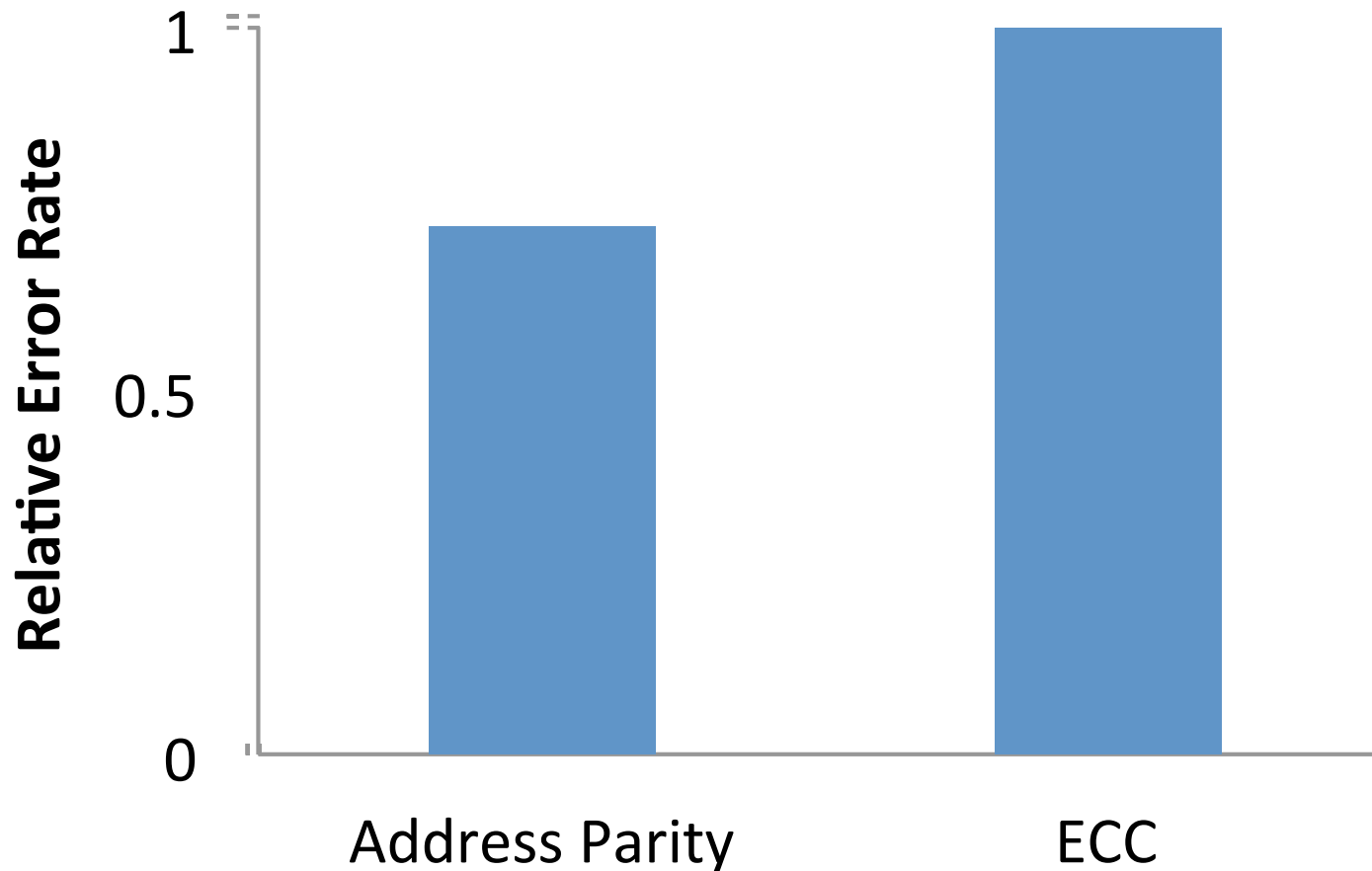
- Often affect multiple rows/columns



Single-bit		
Single-word		
Single-column		
Single-row		
Single-bank		
Multi-bank		
Multi-rank	0.2%	0.8%

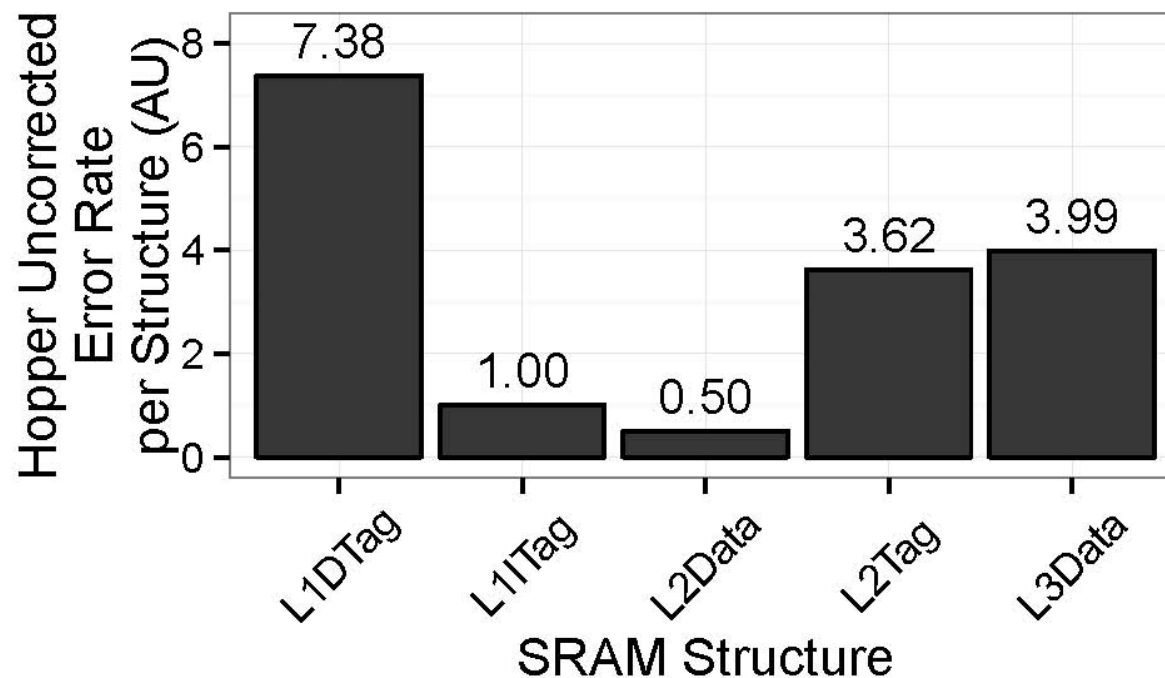
DDR Address/Command Parity

DDR-3 Address/command parity



Address/command parity is a valuable addition to the DDR spec

SRAM: Case Study



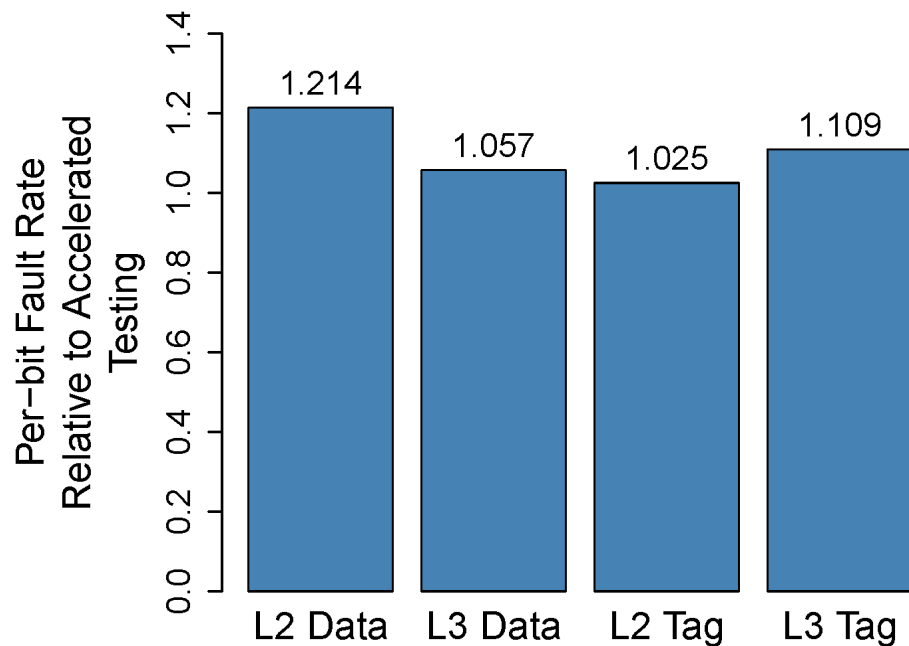
- **Uncorrected errors dominated by L1DTag**
 - Small structure, ~50% of all errors
- **L2Tag has ECC: why so many errors?**
 - One bit per entry is covered by parity

Details matter: seemingly small decisions can have large impact on system reliability

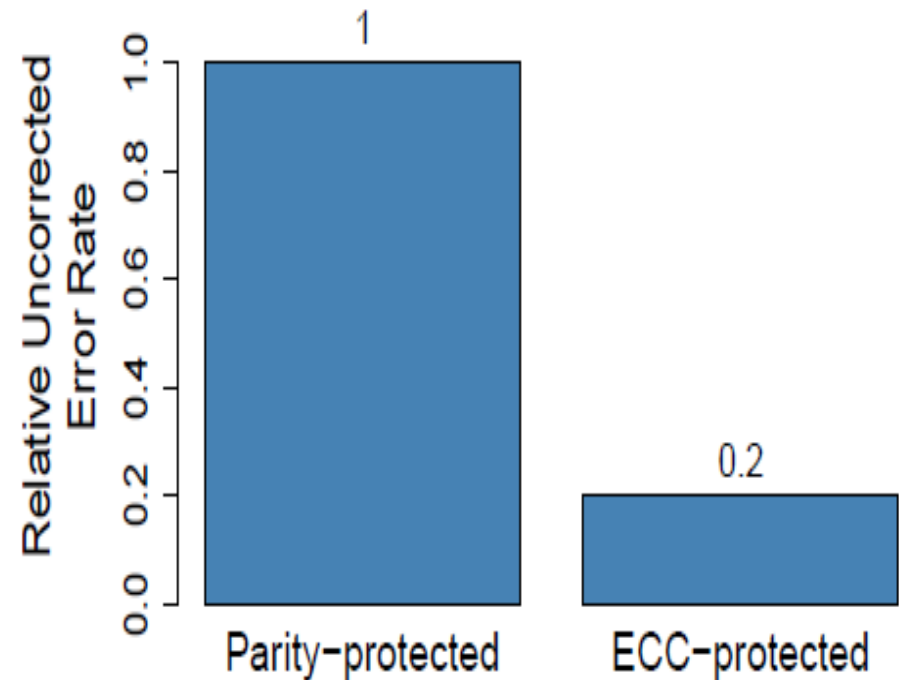
SRAM Faults

ACCELERATED TESTING CORRECTLY PREDICTS ERROR RATES IN THE FIELD

SRAM faults are well-understood



Most errors are from parity-protected structures

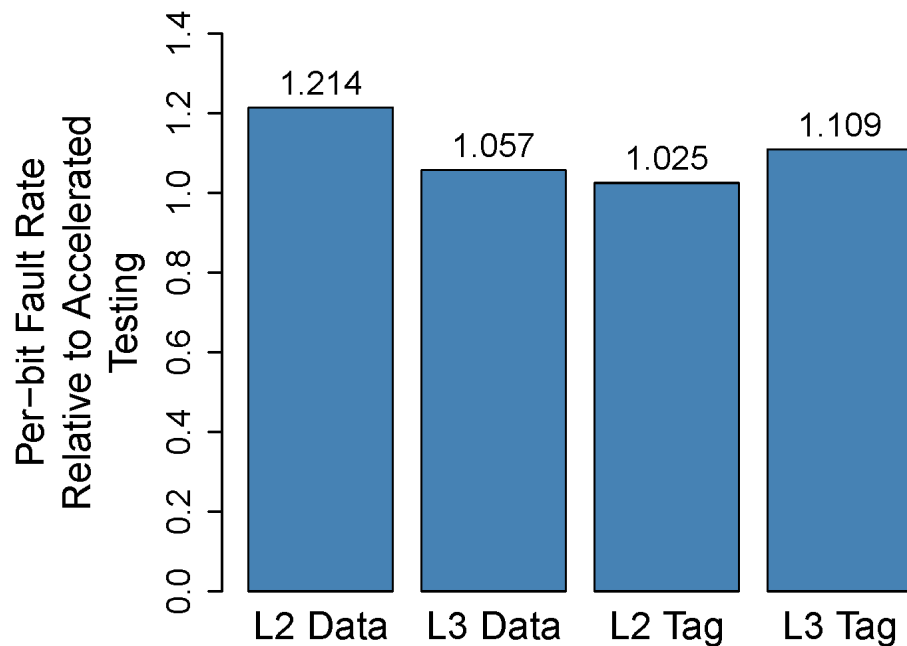


SRAM faults and mitigation techniques are well-understood

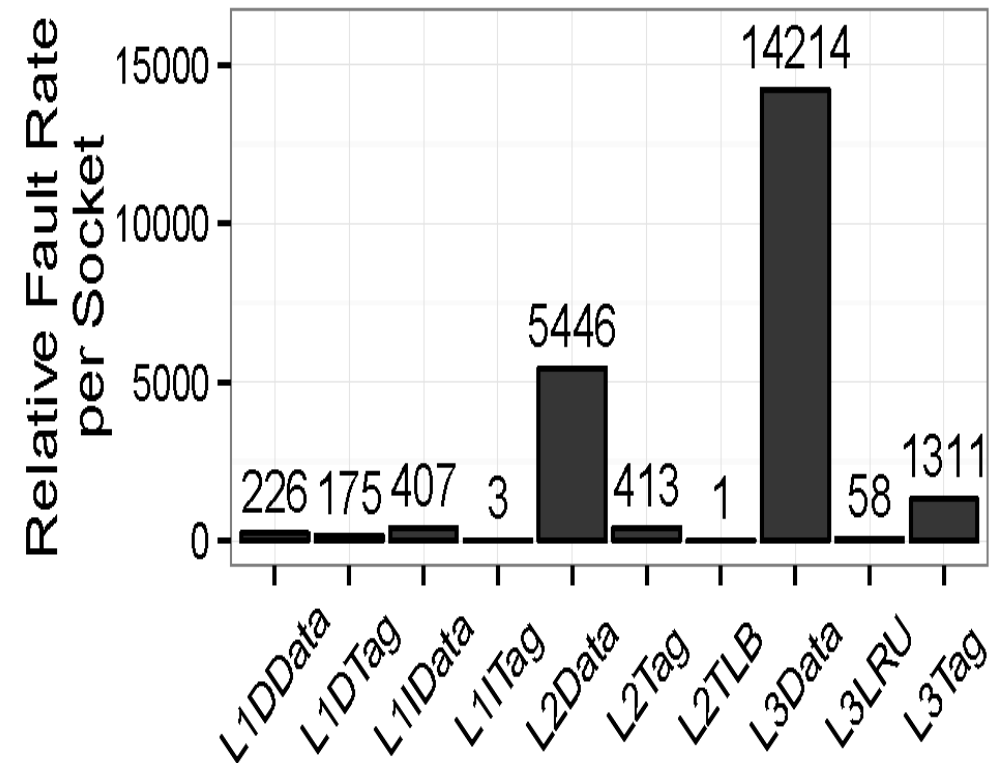
SRAM Faults

FAULT MODEL IS VALIDATED BY ACCELERATED TESTING AND FIELD DATA

SRAM faults are well-understood

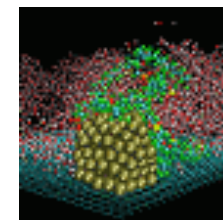
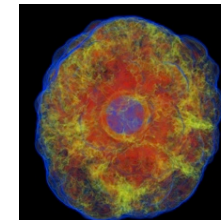
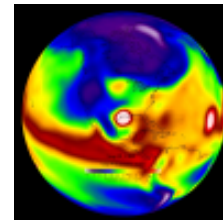
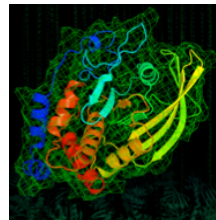
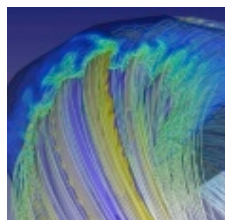
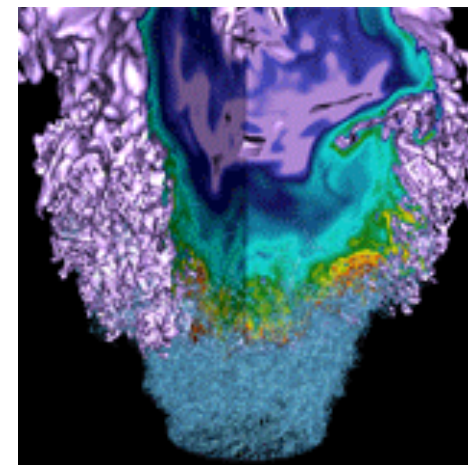


At scale, even small structures see faults



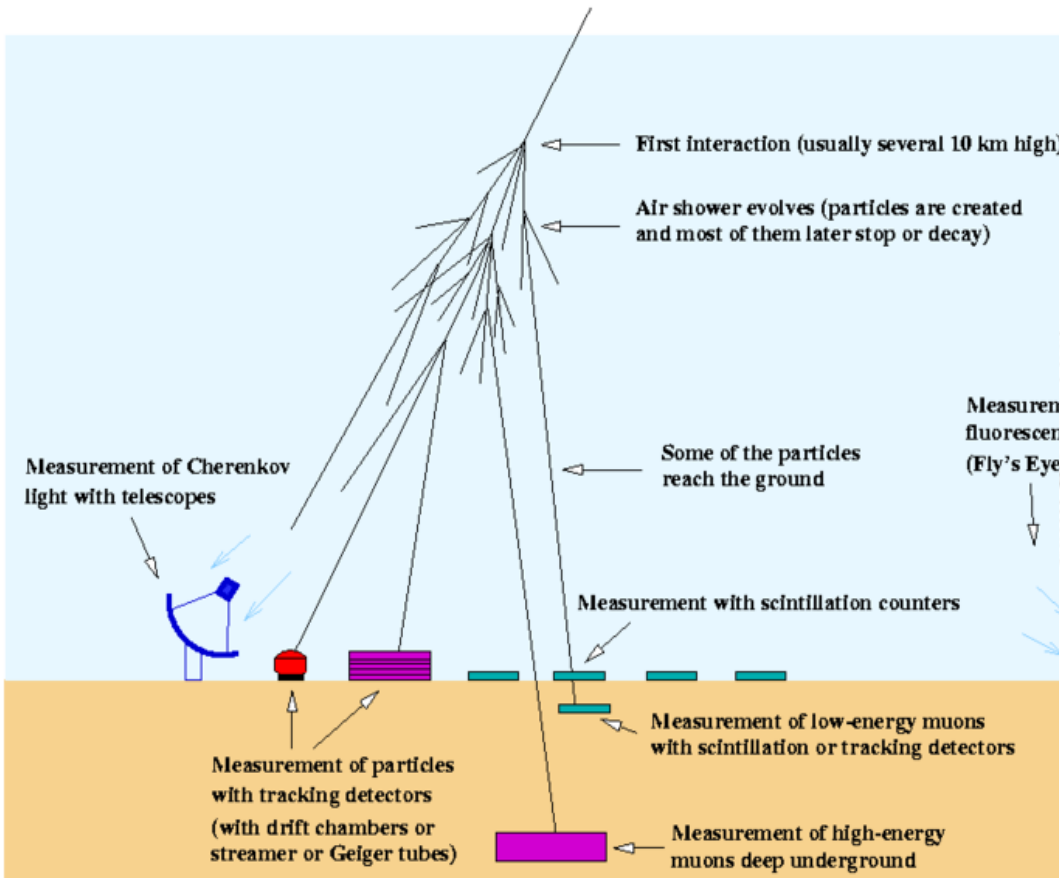
Chip architects must pay attention to reliable design (and they do)

The Bad



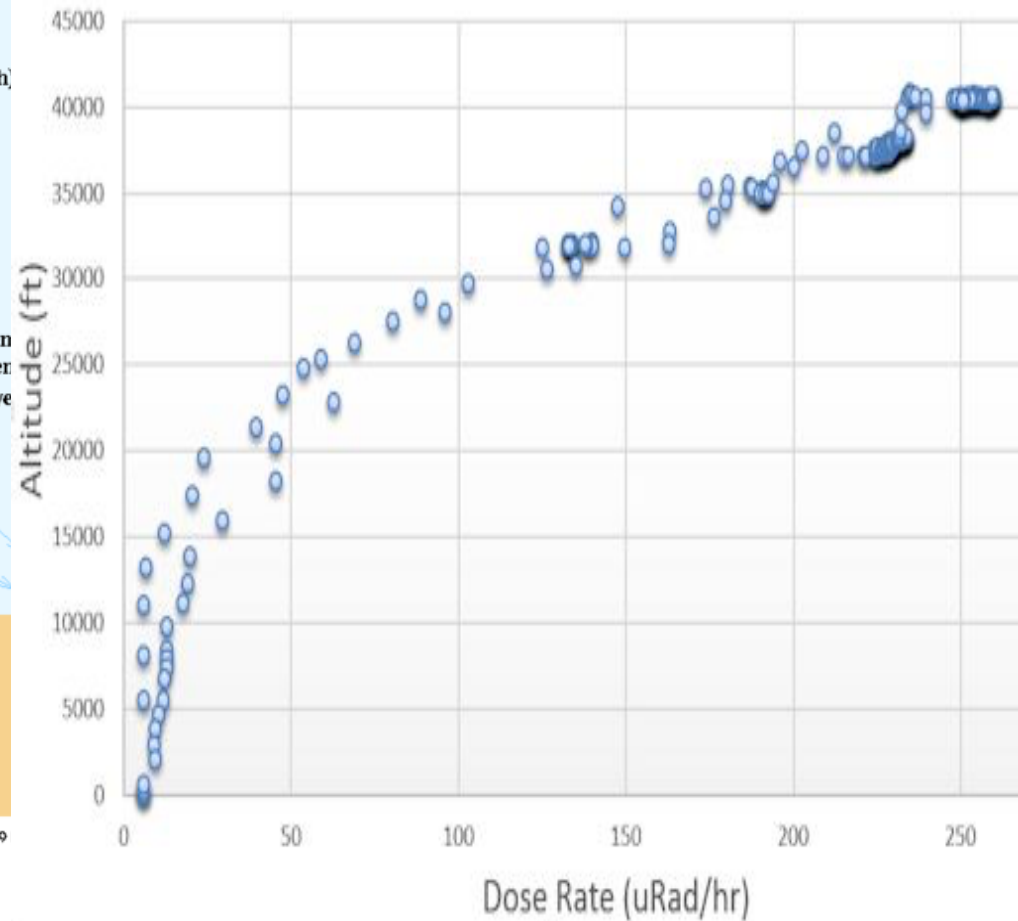
Altitude Effects

Measuring cosmic-ray and gamma-ray air showers

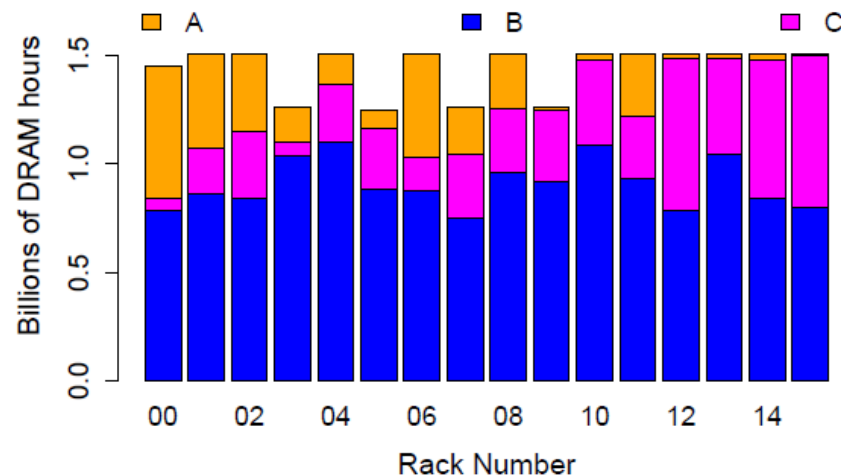
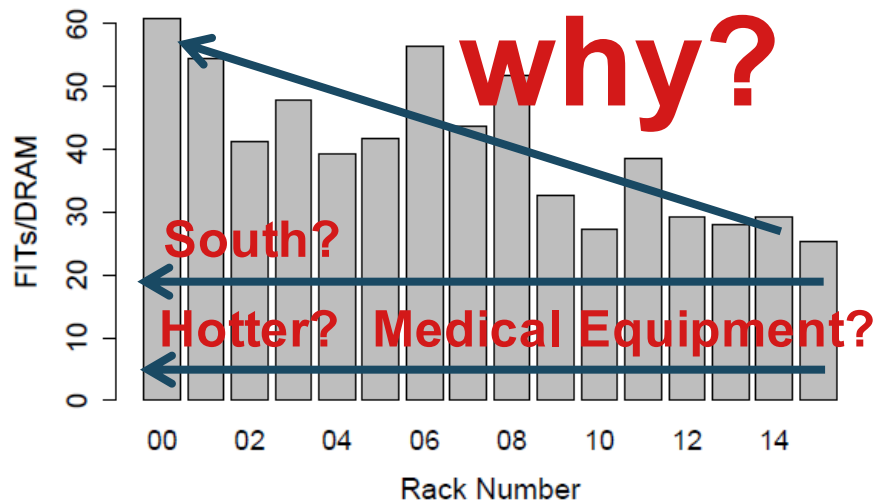


(C) 1999

Radiation Dose Rate vs. Altitude

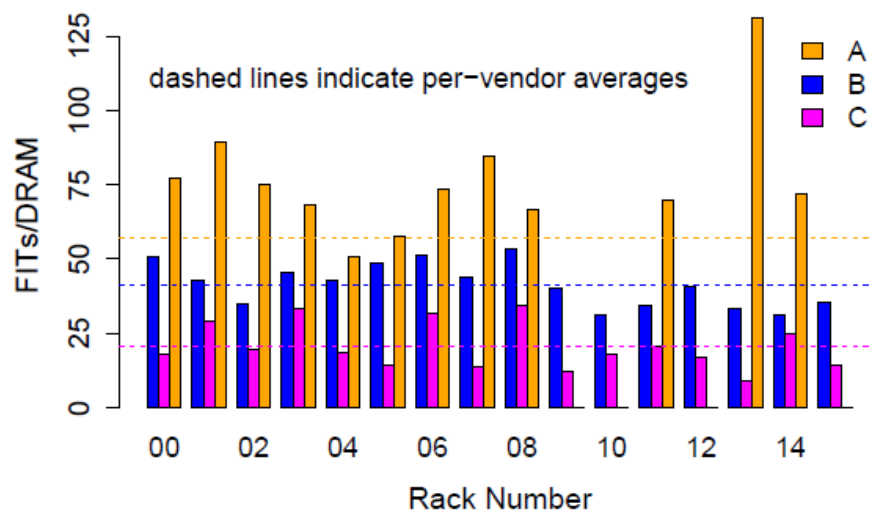


Location Dependence for DRAM Errors?



▶ A correlation to physical location...

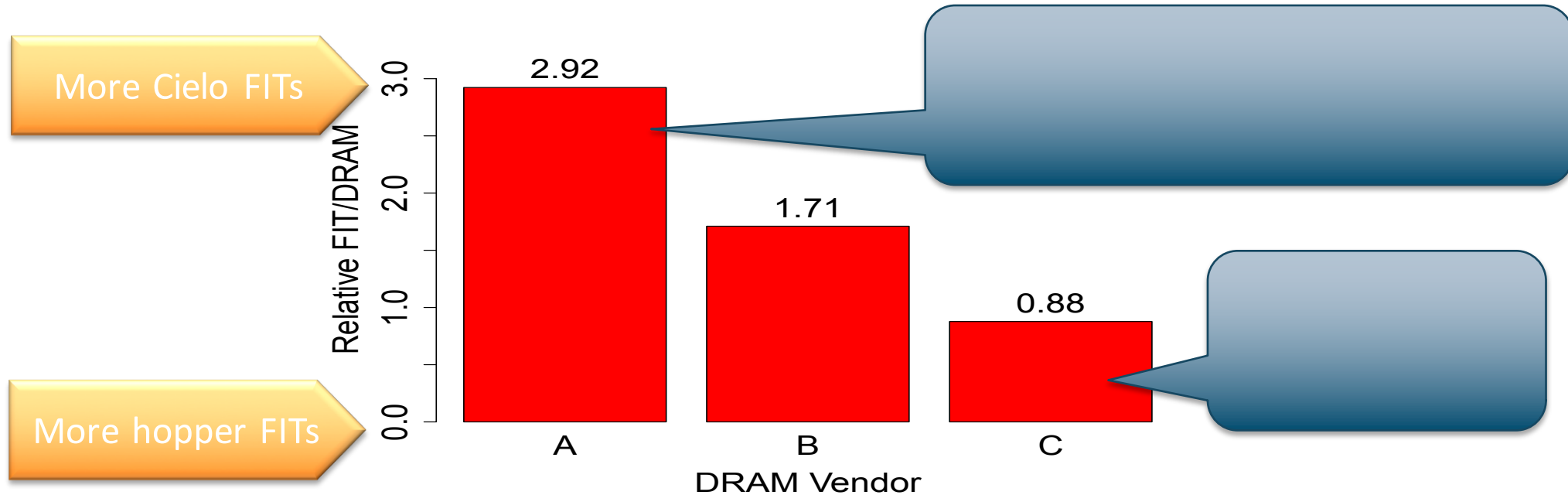
▶ ...is due to non-uniform distribution of vendor...



▶ ...and disappears when examined by vendor.

DRAM reliability studies must account for DRAM vendor or risk inaccurate conclusions

Altitude Effects in DRAM?

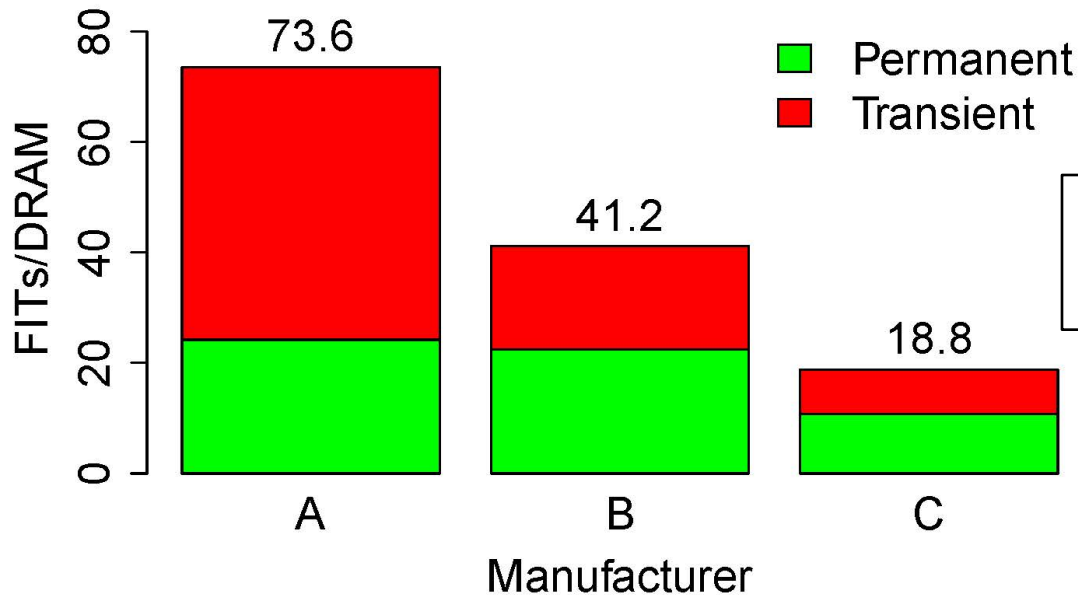


- **Difference in DRAM fault rate on Cielo vs. Hopper**
 - Effect differs per vendor
 - Almost entirely due to a subset of fault modes (single-bit, single-column transient)
- **Primary difference between the two systems is altitude**
 - Cielo at 7000+ ft., Hopper at 43 ft.

Some DRAM devices show a potential altitude effect

Vendor Effects

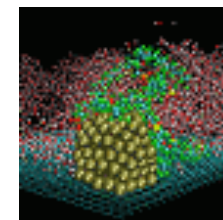
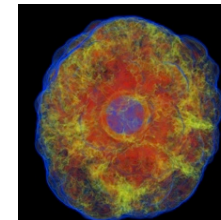
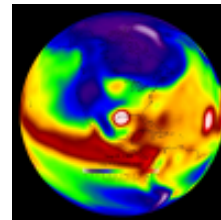
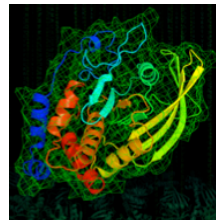
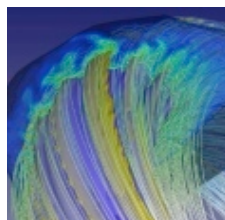
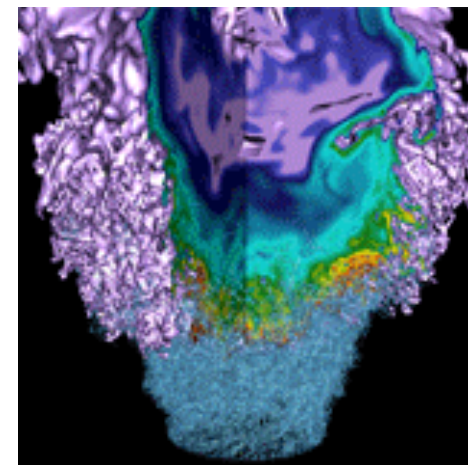
Fault Mode	Vendor A	Vendor B	Vendor C
Single-bit	64.6%	69.5%	58.4%
Single-word	0%	0.3%	0%
Single-column	8.7%	8.8%	11.9%
Single-row	12.2%	10.6%	14.9%
Single-bank	13.5%	7.8%	9.9%
Multiple-bank	1.3%	0.7%	2.0%
Multiple-rank	1.3%	3.0%	3.0%



▶ Fault modes are present across vendors
 ▶ Fault rates differ significantly by vendor

▶ Overall fault rate per vendor

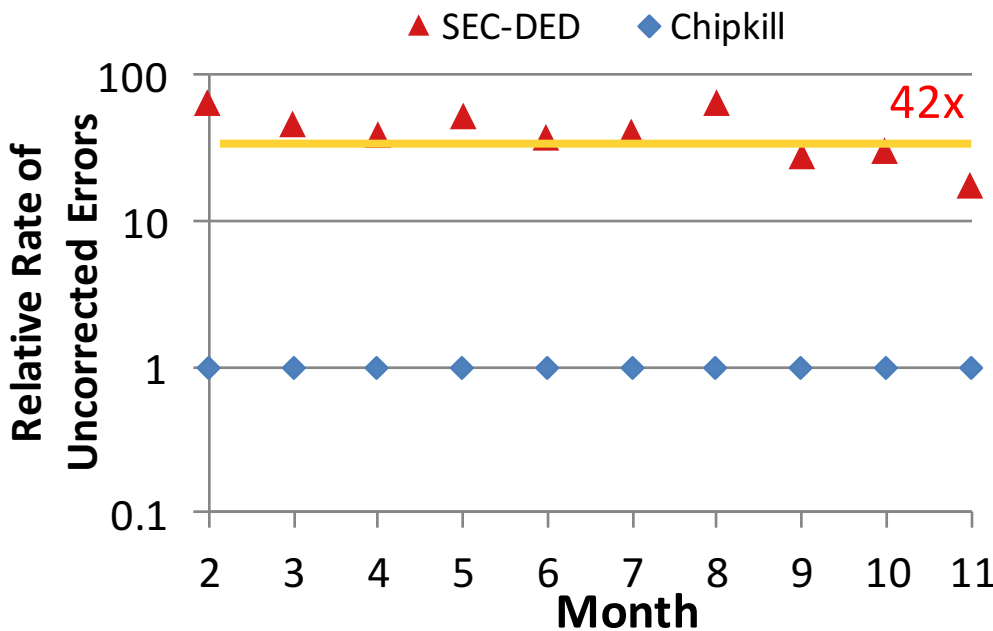
The Ugly



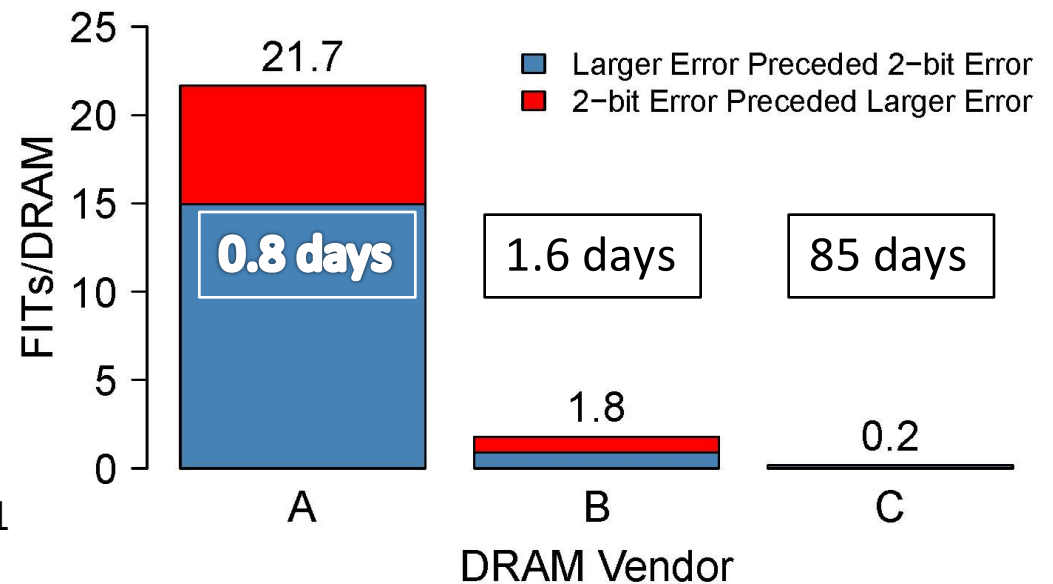
DRAM: Chipkill vs. SEC-DED ECC

- **Chipkill ECC**

- The ability to correct any error from a single DRAM device
- Requires more overhead than SEC-DED ECC (12.5% instead of 7%)
- 30% multibit errors detectable by SEC-DED, but 70% were not



SEC-DED: Rate of Faults Causing Undetected Errors



SEC-DED ECC is poorly suited to modern DRAM technology

Counting Faults vs. Counting Errors

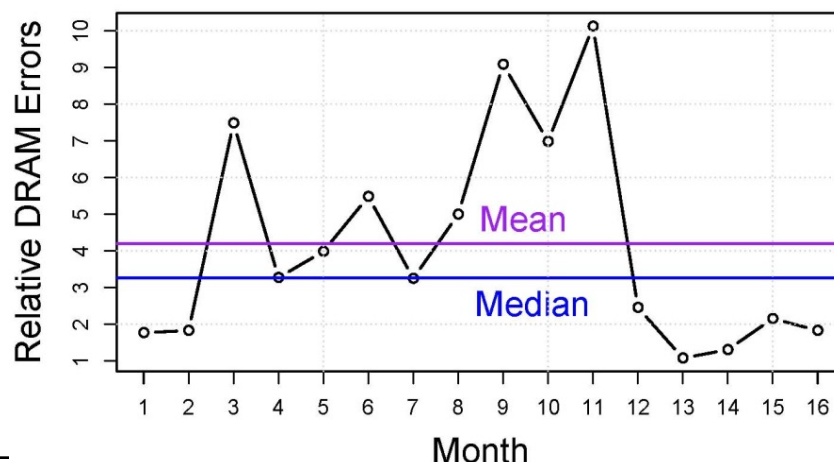
- **Counting logged errors overemphasizes the impact of permanent faults**
 - Error events are not independent
 - A single fault generates an arbitrary number of errors (0 -> infinite)
 - Permanent faults tend to cause more errors than transient faults
- ▲ **The logged corrected error count is *meaningless* for system health**
 - Operating system polls for corrected errors (e.g., once every 10 seconds)
 - But a modern system can experience millions of errors per second
 - Console log contains a (small) sample of corrected errors
- ▲ **The logged uncorrected error count is *meaningful* for system health**
 - Every uncorrected error is reported to the operating system via interrupt
 - Console log contains an exact count of uncorrected errors

Incorrect methodology can lead to incorrect conclusions about system reliability

Hopper has a memory error rate **4x** that of Cielo,
but a memory fault rate **0.625x** that of Cielo.

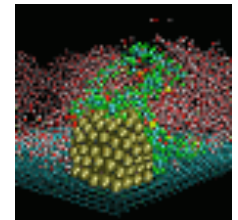
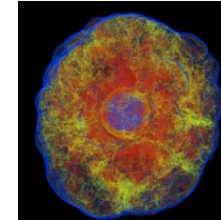
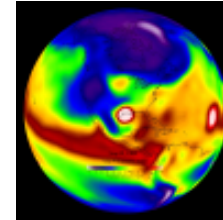
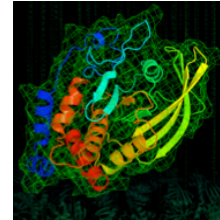
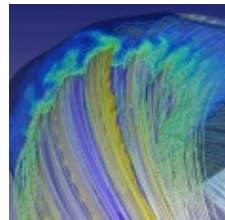
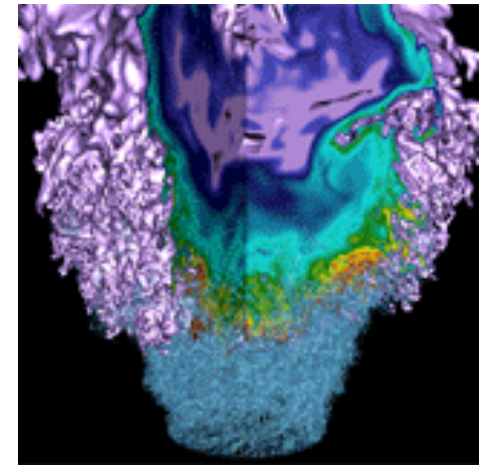
*Error counts are confounded by other factors
such as workload behavior, they are not an
accurate measure of system health.*

- Hopper's DRAM **error rate** was 4x greater than Cielo's ← ~~Cielo is more reliable~~
- **Reality:** Hopper's DRAM **fault rate** was 37% lower than Cielo's



Incorrect methodology can lead to incorrect conclusions about system reliability

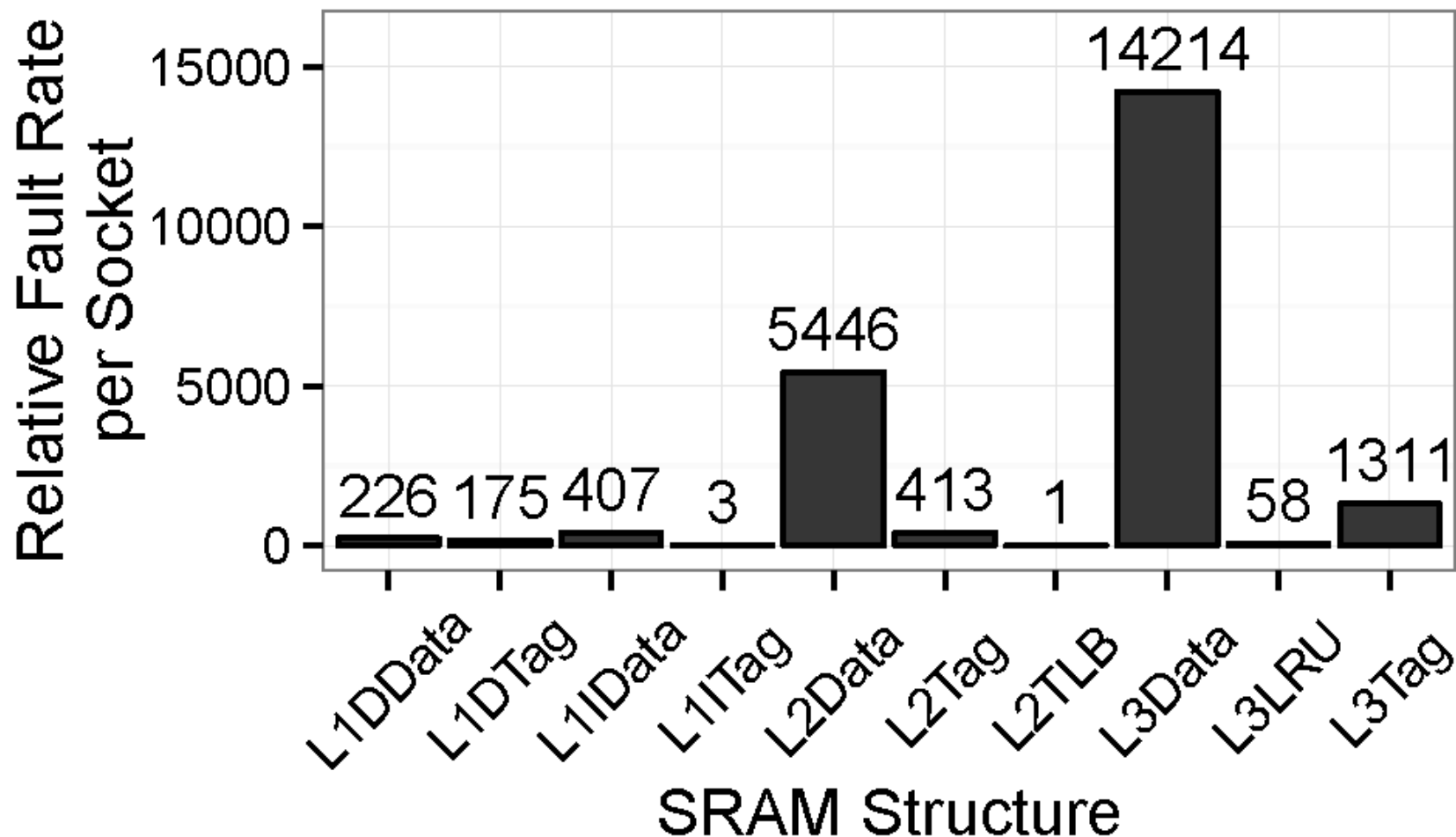
Projecting to Exascale



NERSC **40** YEARS
at the
FOREFRONT
1974-2014

SRAM: Projecting to Exascale

AT SCALE, EVEN SMALL STRUCTURES SEE FAULTS



Vendors must pay attention to reliable design

SRAM: Projecting to Exascale

SRAM UNCORRECTED ERROR RATE RELATIVE TO CIELO

▲ Two potential systems

- Small: 10k nodes
- Large: 100k nodes

▲ Same fault rate as 45nm

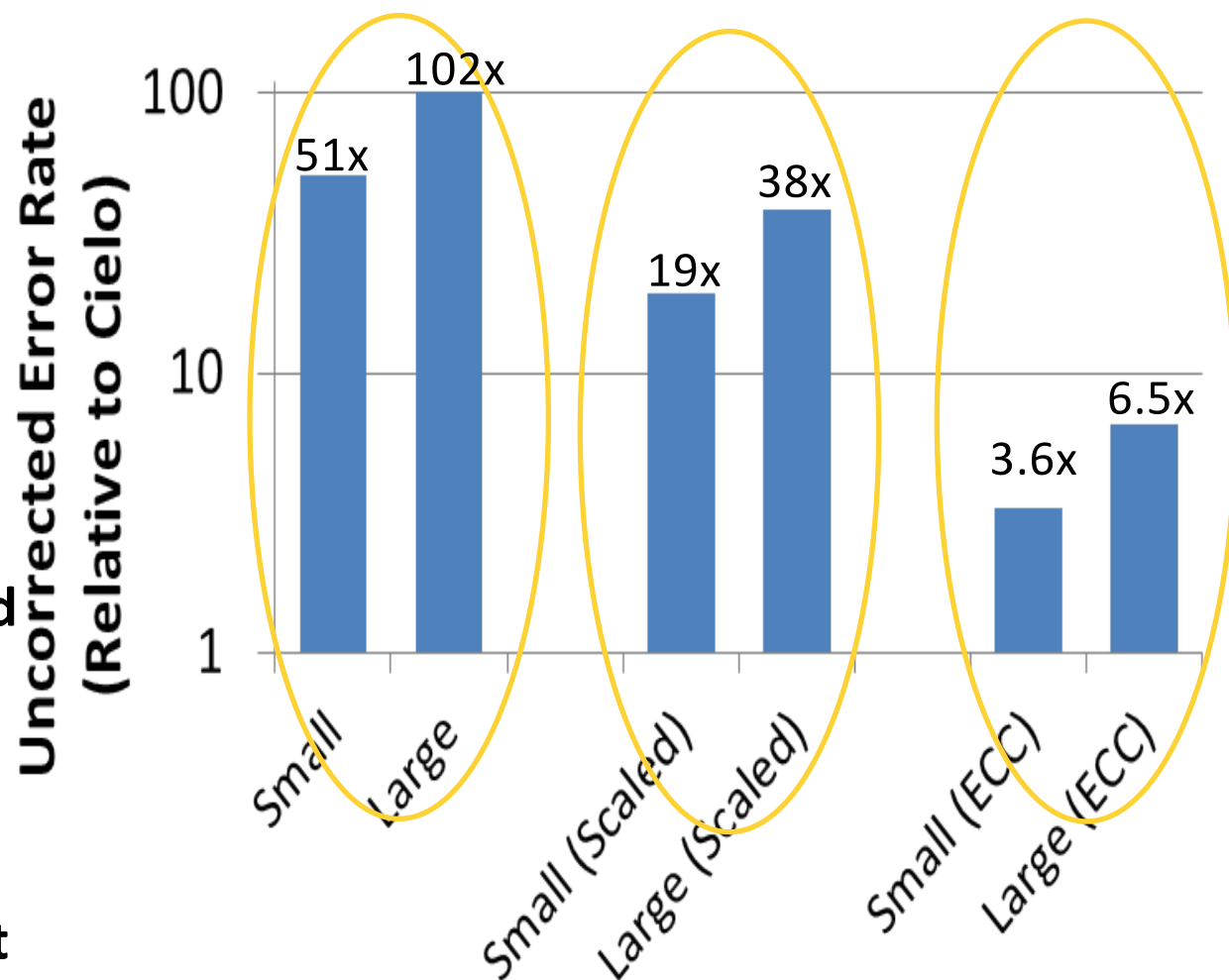
- Sky is falling

▲ Scale faults per current trend

- Sky falls more slowly
- Switch to FinFETs may make this even better

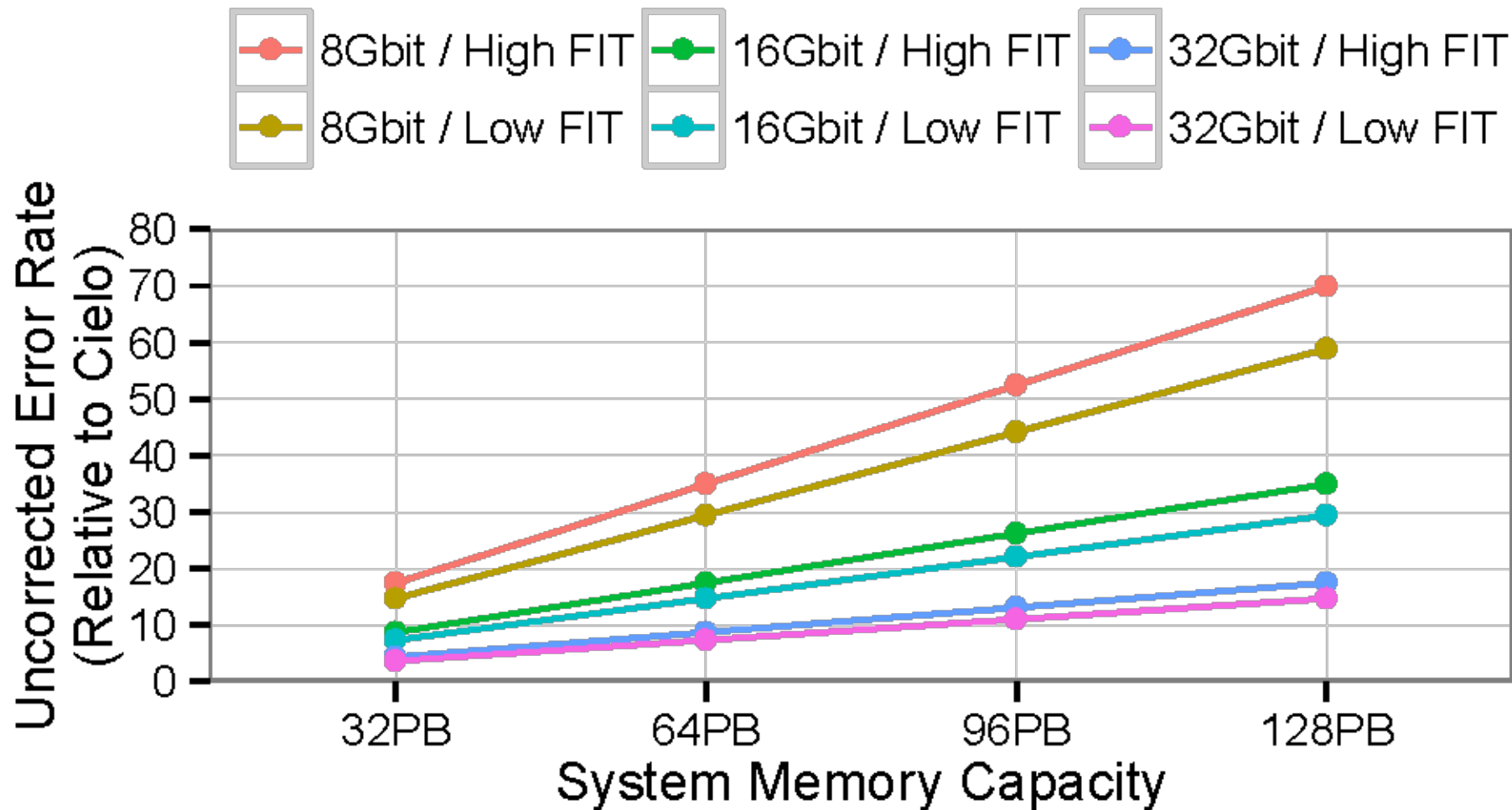
▲ Add some engineering effort

- Sky stops falling



SRAM faults are unlikely to be a significantly larger problem than today

DRAM: Projecting to Exascale



DRAM: Projecting to Exascale

Uncorrected error rate

- 10-70x error rate of current systems
- Is the sky falling?

This is not just a problem for exascale

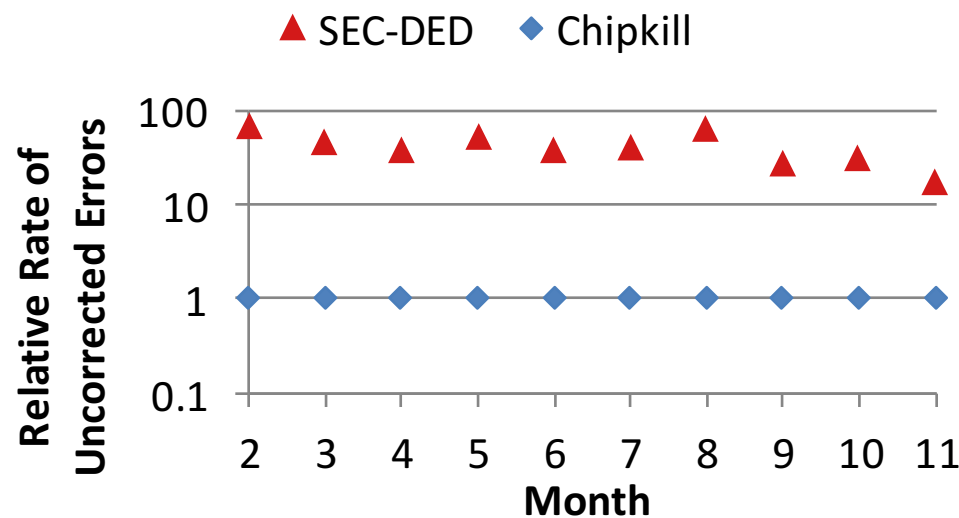
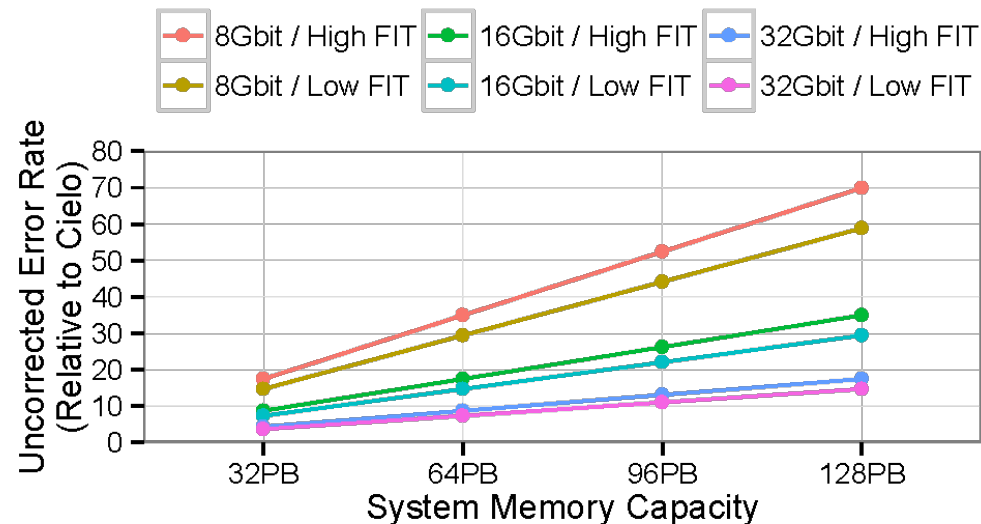
- Cost problem for data centers / cloud
- Reliability problem in client (smart cars)?

Solutions are out there

- Including for die-stacked DRAM?
- Lots of people working on this...

Historical example

- Chipkill vs. SEC-DED



DRAM subsystems need higher reliability than today, but will likely get it

Conclusions



- **Large systems require reliable design and reliability modeling**
- **Field data analysis is necessary to correlate reliability models and guide DOE investments**
 - Must measure the underlying fault rate to correctly evaluate the model
 - Must track component supplier to make proper conclusions
- **Collaboration between DOE researchers, vendors, and integrators, and facilities is critical to achieving this**

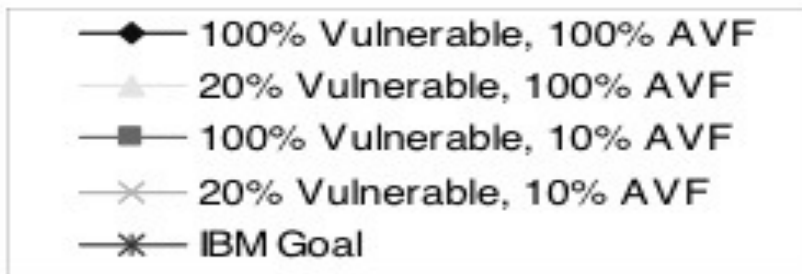
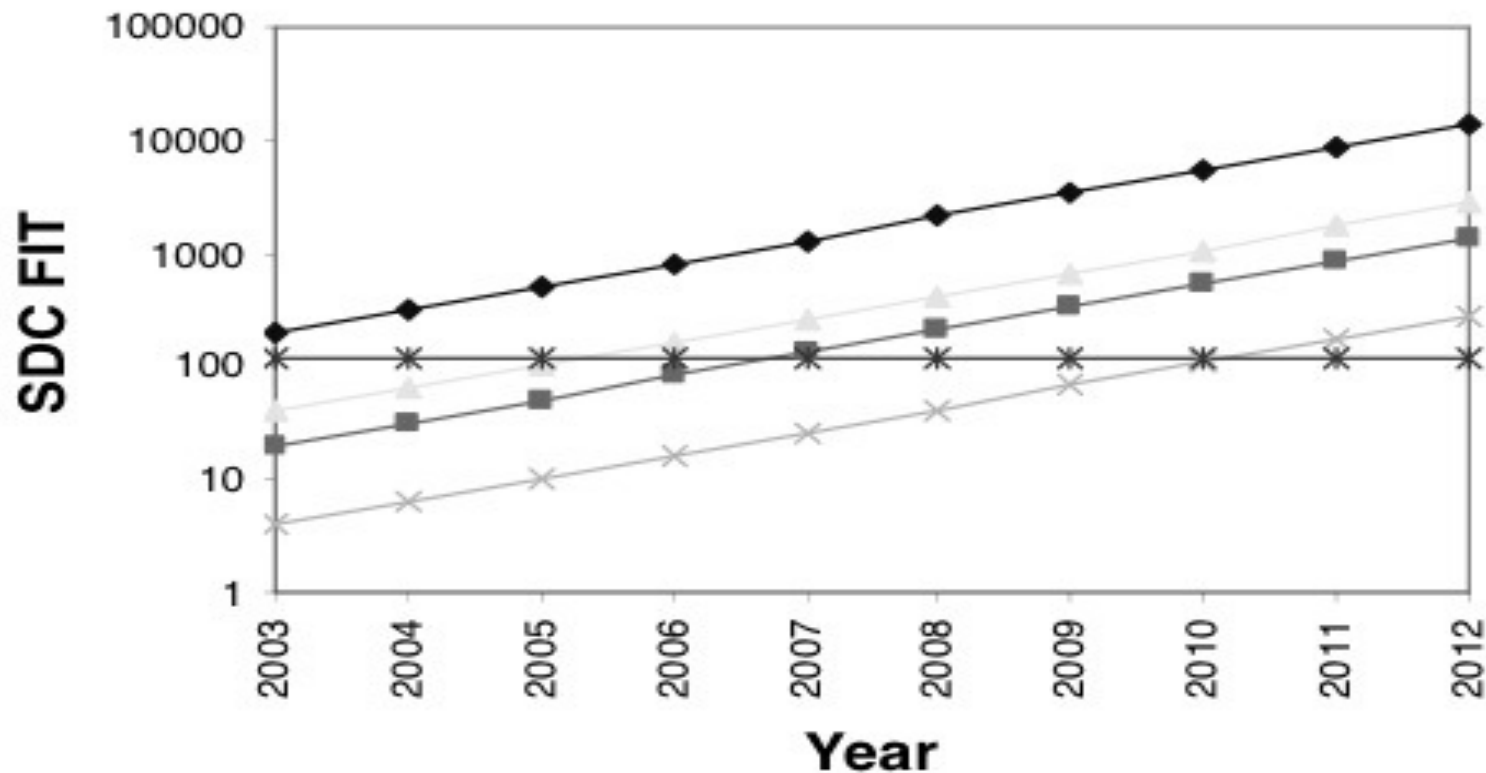
Risk Factors for the Future



- **SRAM Structures**
 - **Likelihood of faults:** very high
 - **Risk:** low
 - **Why?:** Model for particle strikes on CMOS SRAM remains solid (it's a matter of engineering and cost... no magic required)
- **JEDEC DIMM Structures**
 - **Likelihood of faults:** medium
 - **Risk:** medium (lower if move to chip-kill)
 - **Why?:** Component supplier has more pronounced effect than environmental factors. SEC-DED and DRAM is clearly insufficient
- **Stacked DRAM (HBM, etc...)**
 - **Likelihood of faults:** medium
 - **Risk:** high (*will be lower after field data collected from first sys.*)
 - **Why?:** No field test data to confirm very well thought-out models (might be no issue, but always risk for unverified model)

Reliability of Components Set by Market

(SER rates rising, but...)



Reliability of Components Set by Market

(set point for hardware resilience set by market)

(be slightly more reliable than the OS)

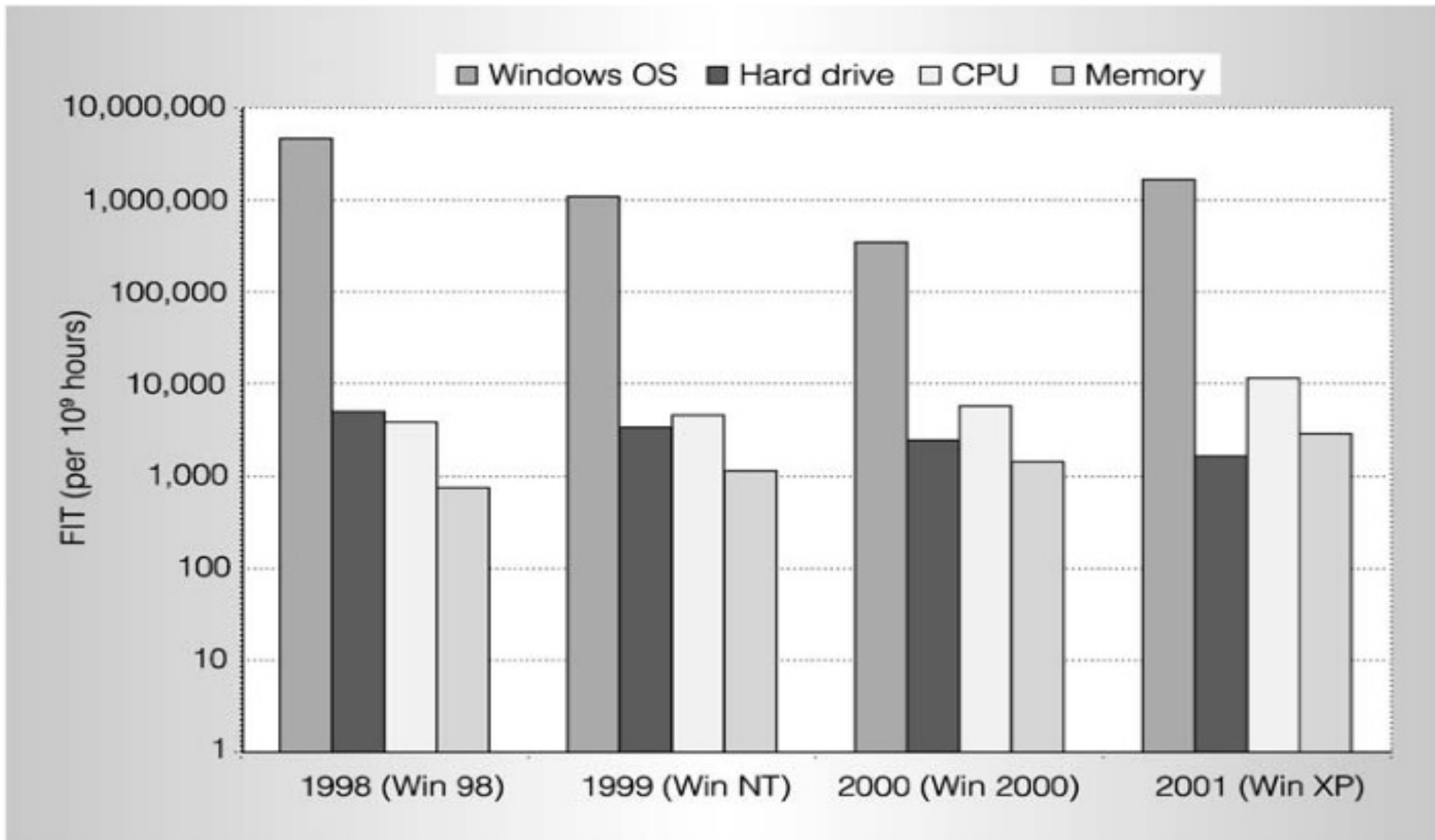


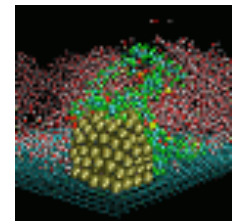
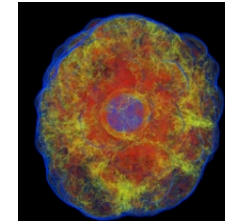
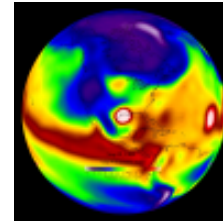
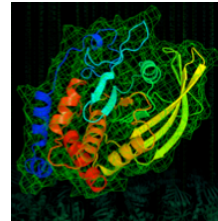
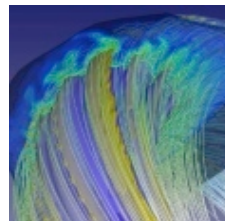
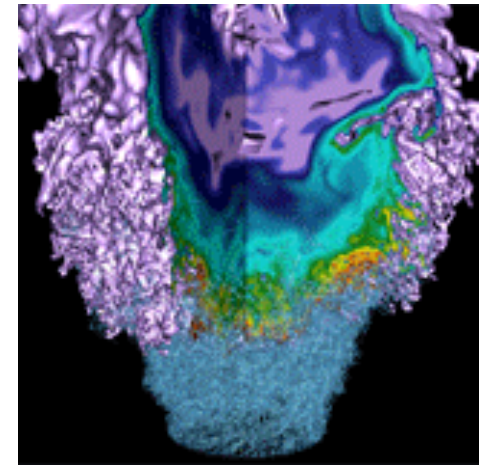
Figure 2. Failures in billions of hours of operation.²⁻⁵

Synergy with Embedded Industry



- The error tolerance requirements for self-driving vehicles are approaching that required
- The same microarchitecture error-tolerance techniques will be employed in both places (*more leverage for HPC resilience*)
- ... I just wrote this a few seconds ago in response to Martin Berzins' question during the break... (so lets just talk about this)

The End

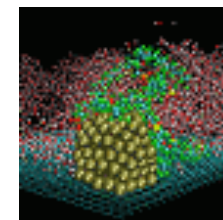
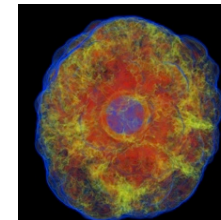
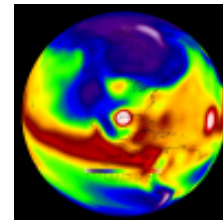
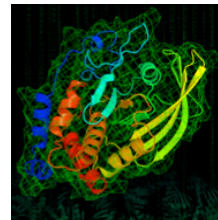
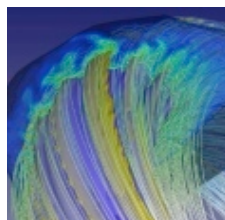
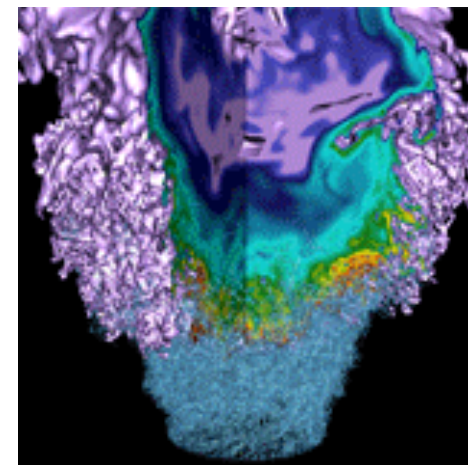


References

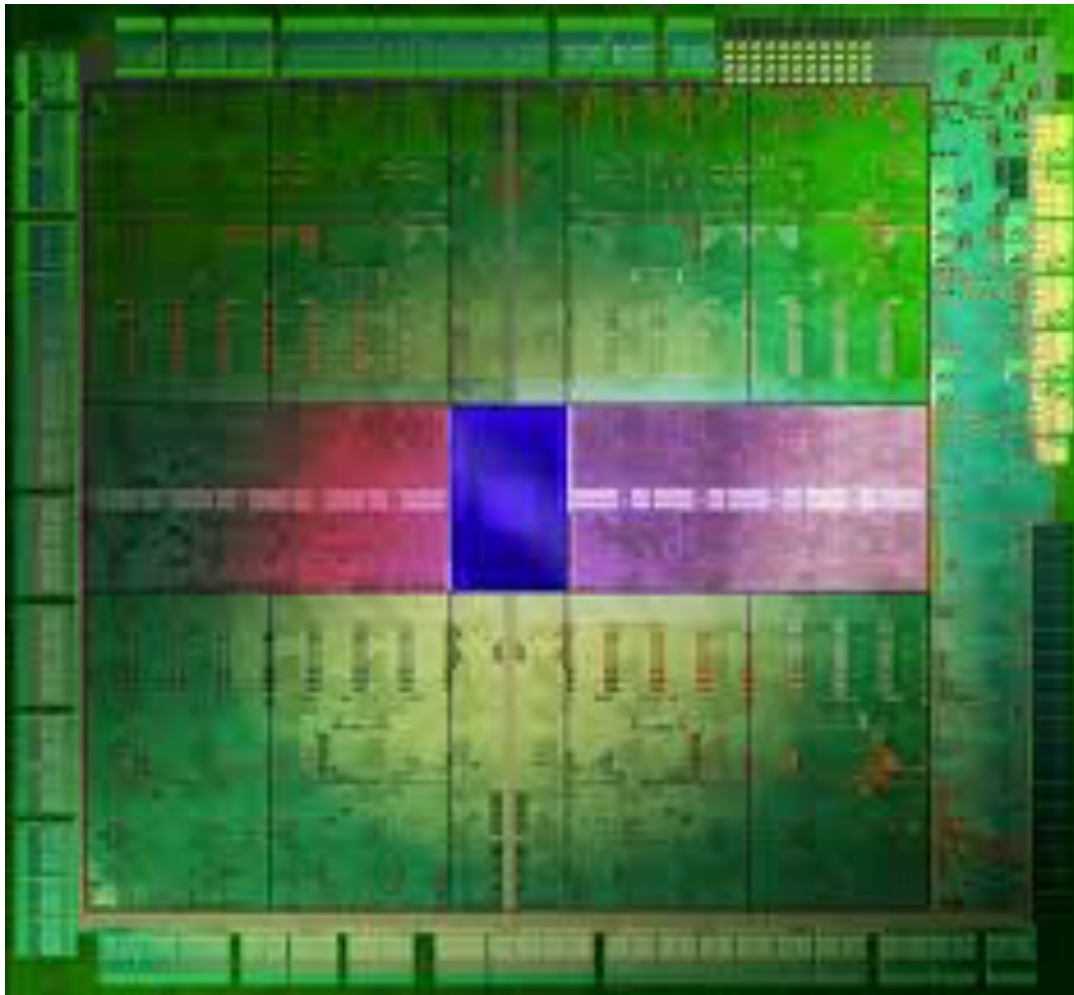


- V. Sridharan, N. DeBardleben, S. Blanchard, K. Ferreira, J. Stearley, and S. Gurumurthi. Memory errors in modern systems: The good, the bad, and the ugly. *International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, 2015.
- V. Sridharan, J. Stearley, N. DeBardleben, S. Blanchard, and S. Gurumurthi. Feng shui of supercomputer memory: Positional effects in DRAM and SRAM faults. *International Conference on High Performance Computing, Networking, Storage and Analysis (SC13)*, 2013.
- V. Sridharan and D. Liberty. A study of DRAM failures in the field. *International Conference on High Performance Computing, Networking, Storage and Analysis (SC12)*, 2012
- A. Avizienis, J.-C. Laprie, B. Randell, and C. Landwehr. Basic concepts and taxonomy of dependable and secure computing. *IEEE Transactions on Dependable and Secure Computing*, 2004.
- C. Di Martino, Z. Kalbarczyk, R. K. Iyer, F. Baccanico, J. Fullop, and W. Kramer. Lessons learned from the analysis of system failures at petascale: The case of Blue Waters. *International Conference on Dependable Systems and Networks*, 2014.
- B. Schroeder, E. Pinheiro, and W.-D. Weber. DRAM errors in the wild: a large-scale field study. *SIGMETRICS*, 2009.

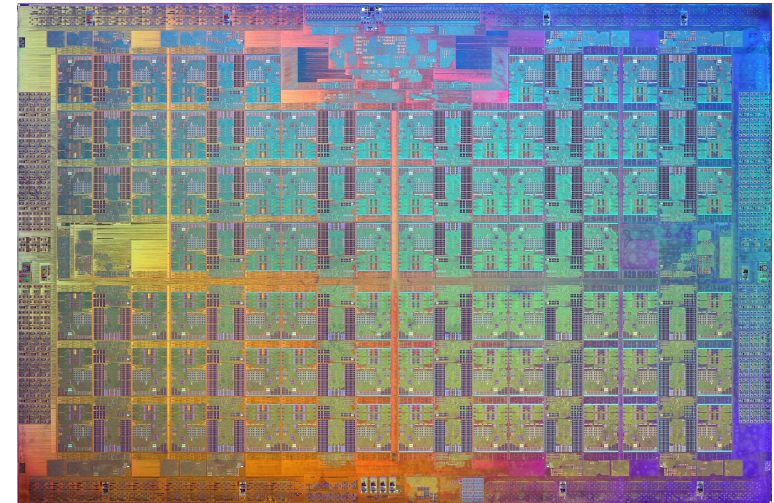
Backup



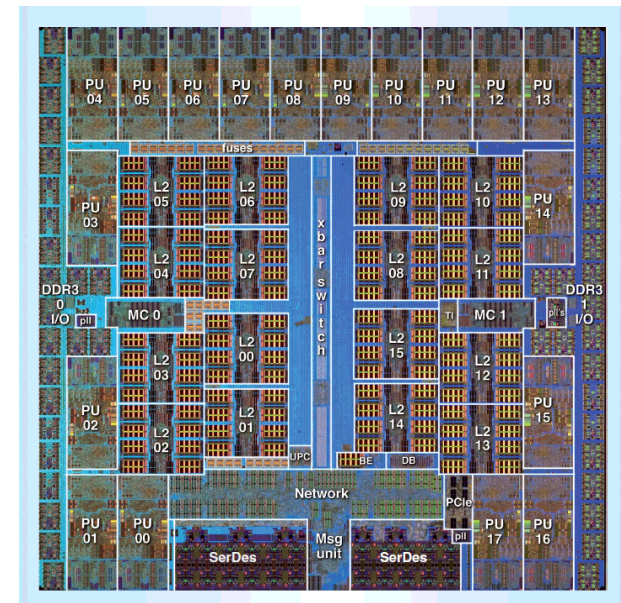
Die Sizes: Area affects probability of fault from energetic particles (e.g. cosmic rays)



NVIDIA Maxwell: 600mm²



Intel KNL: 360mm²
(same area for *BG/Q* chip)





Observing Ultra-High Energy Cosmic Rays with Smartphones

Daniel Whiteson, Michael Mulhearn, Chase Shimmin, Kyle Cranmer, Kyle Brodie, Dustin Burns

(Submitted on 10 Oct 2014 (v1), last revised 22 Oct 2015 (this version, v2))

We propose a novel approach for observing cosmic rays at ultra-high energy ($> 10^{18}$ eV) by repurposing the existing network of smartphones as a ground detector array. Extensive air showers generated by cosmic rays produce muons and high-energy photons, which can be detected by the CMOS sensors of smartphone cameras. The small size and low efficiency of each sensor is compensated by the large number of active phones. We show that if user adoption targets are met, such a network will have significant observing power at the highest energies.

Comments: version 2

Subjects: **Instrumentation and Methods for Astrophysics (astro-ph.IM)**; High Energy Astrophysical Phenomena (astro-ph.HE); High Energy Physics – Phenomenology (hep-ph); Instrumentation and Detectors (physics.ins-det)

Cite as: **arXiv:1410.2895 [astro-ph.IM]**
(or **arXiv:1410.2895v2 [astro-ph.IM]** for this version)

Submission history

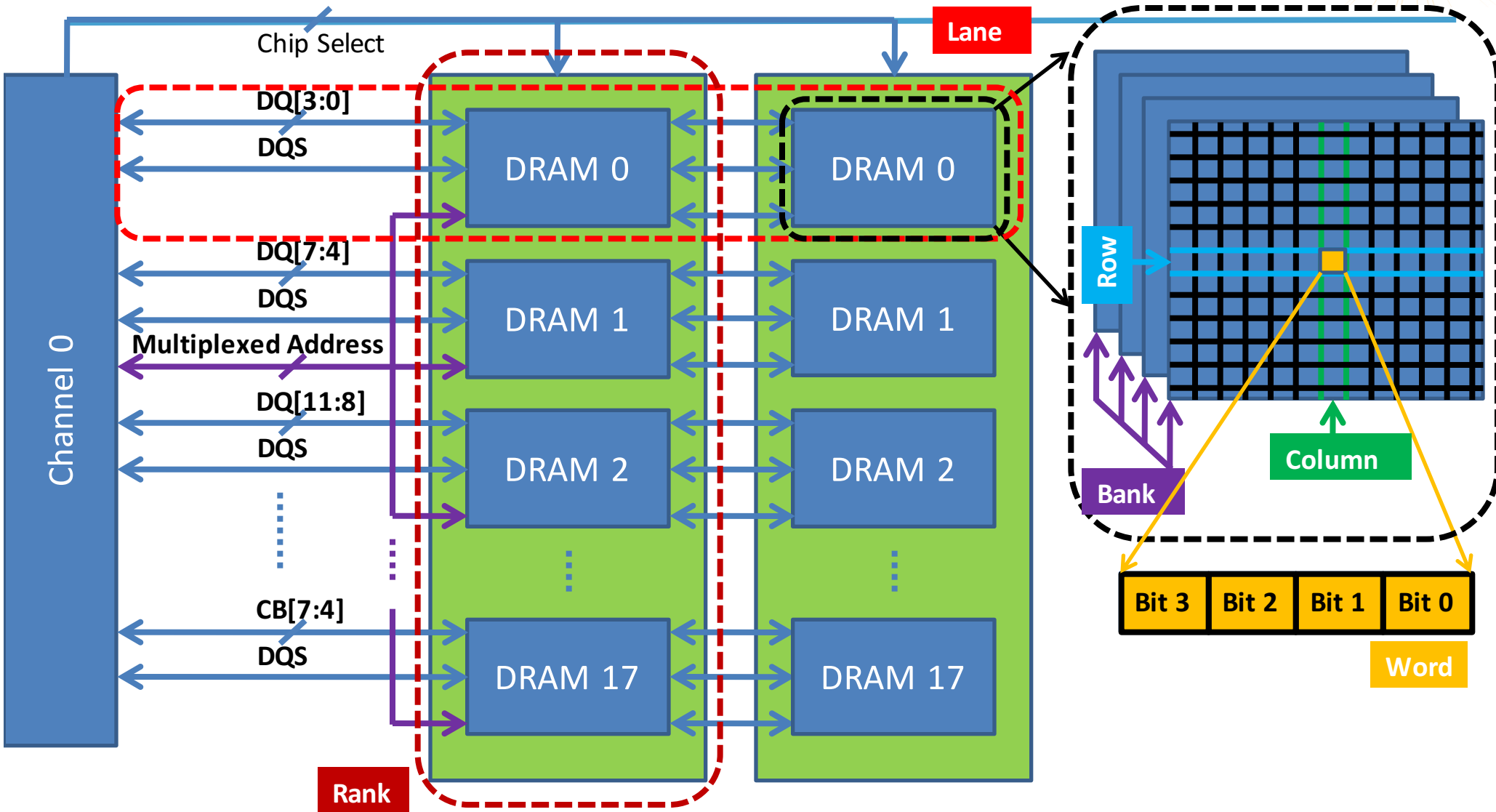
From: Daniel Whiteson [[view email](#)]

[v1] Fri, 10 Oct 2014 20:00:07 GMT (151kb,D)

[v2] Thu, 22 Oct 2015 20:25:31 GMT (619kb,D)

[Which authors of this paper are endorsers?](#) | [Disable MathJax](#) ([What is MathJax?](#))

memory channel Organization

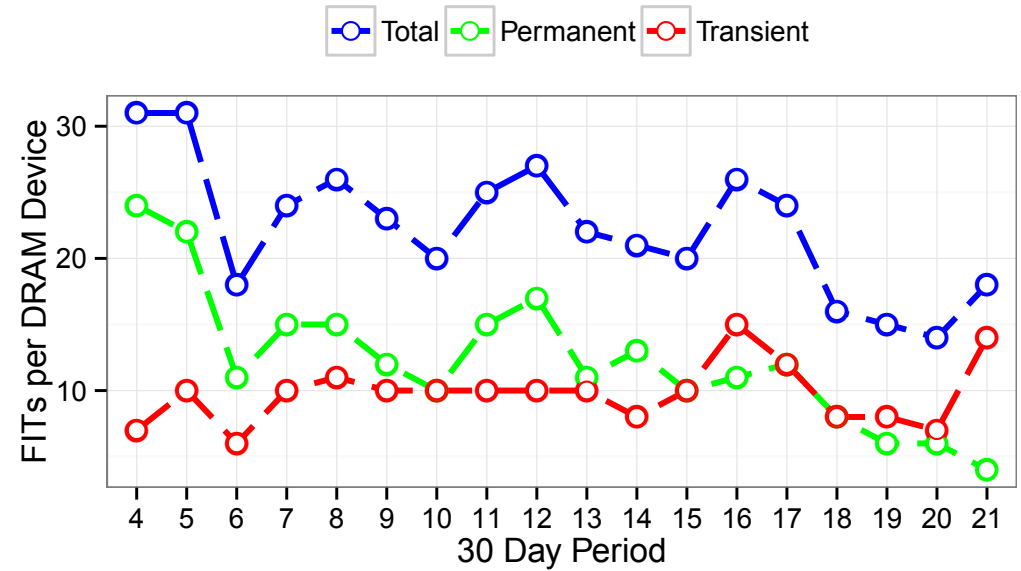


Each logical entity (e.g. row, rank) shares control logic

DRAM FAULT Rates and MODES

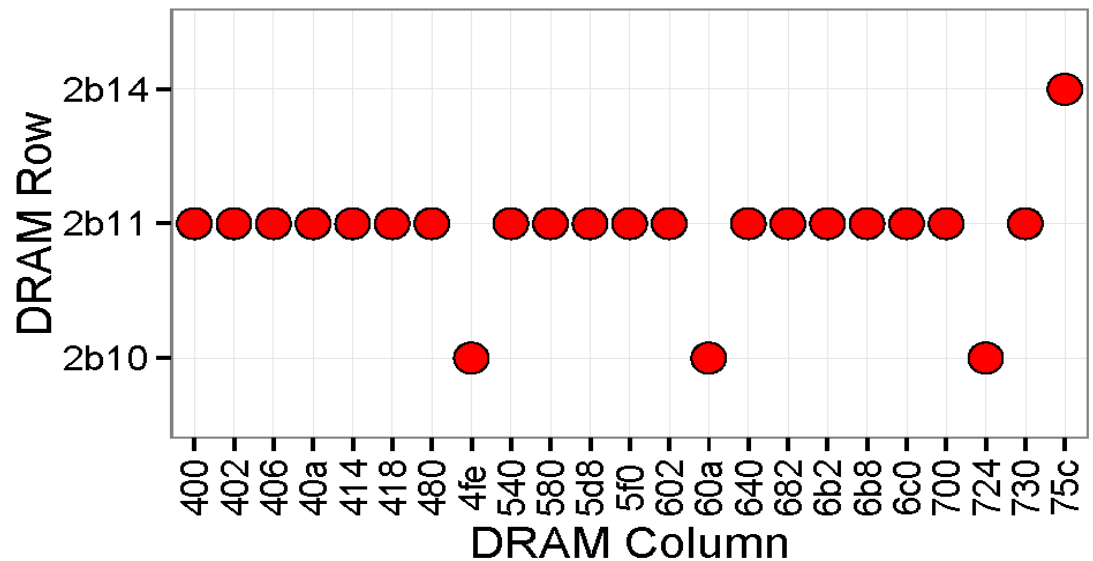
▲ Fault rates

- Constant rate of transient faults
- Declining rate of permanent faults
- >50% permanent faults



▲ Fault modes

- Often affect multiple rows/columns:



The more more general techniques have more overhead (like TMR), but can be used for broad array of code without any understanding of the code. Mike Heroux' s resilient elliptic solver is very specific to the mathematical formulation.

