Faults, Errors and Failures in Extreme-Scale Supercomputers

Christian Engelmann, Ph.D.

Senior Scientist & Group Leader
Intelligent Systems and Facilities Group
Advanced Computing Systems Research Section
Computer Science and Mathematics Division
Oak Ridge National Laboratory

ORNL is managed by UT-Battelle, LLC for the US Department of Energy

Motivation

- Resilience in extreme-scale supercomputers is an optimization problem between the key design and deployment cost factors:
  - Performance, resilience, and power consumption
- The challenge is to build a reliable system within a given cost budget that achieves the expected performance.
- This requires fully understanding the resilience problem and offering efficient resilience mitigation technologies.
  - What is the fault model of such systems?
  - What is the impact of faults on applications?
  - How can mitigation in hard-/software help and at what cost?

Characterizing Supercomputer Faults, Errors and Failures

Novel Ideas:

- Applies a unified taxonomy for supercomputer faults, errors and failures
- Understanding resilience is a data analytics problem, requiring fusion and analysis of different logs and system health data

Impact:

- Develops an understanding of observed and inferred supercomputer reliability conditions
- Extrapolates this knowledge to future systems
- Enables the systematic improvement of resilience in extreme-scale systems
- Keeps applications running to a correct solution in a timely and efficient manner in spite of frequent faults, errors, and failures

Accomplishments:

- Analyzed 1.2 billion node hours of logs from the Jaguar, Titan, and Eos systems at OLCF
- Developed tools for analyzing logs and creating a fault, error and failure catalog
- Created novel modeling techniques to characterize temporal and spatial failure behavior

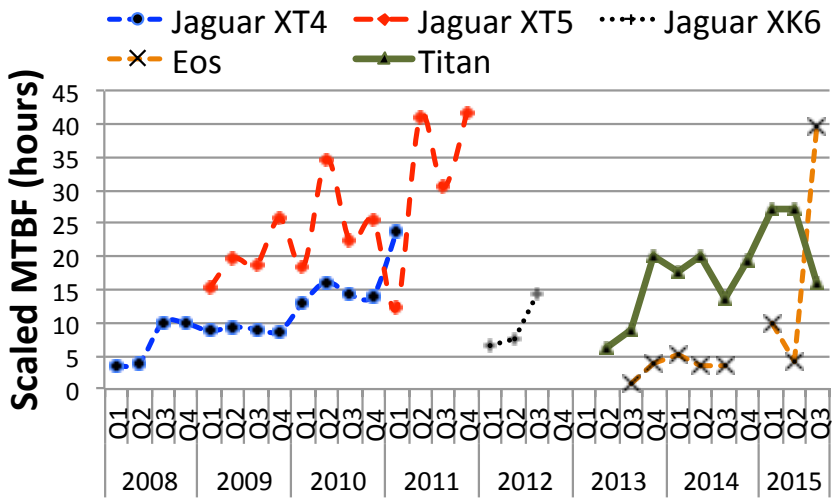
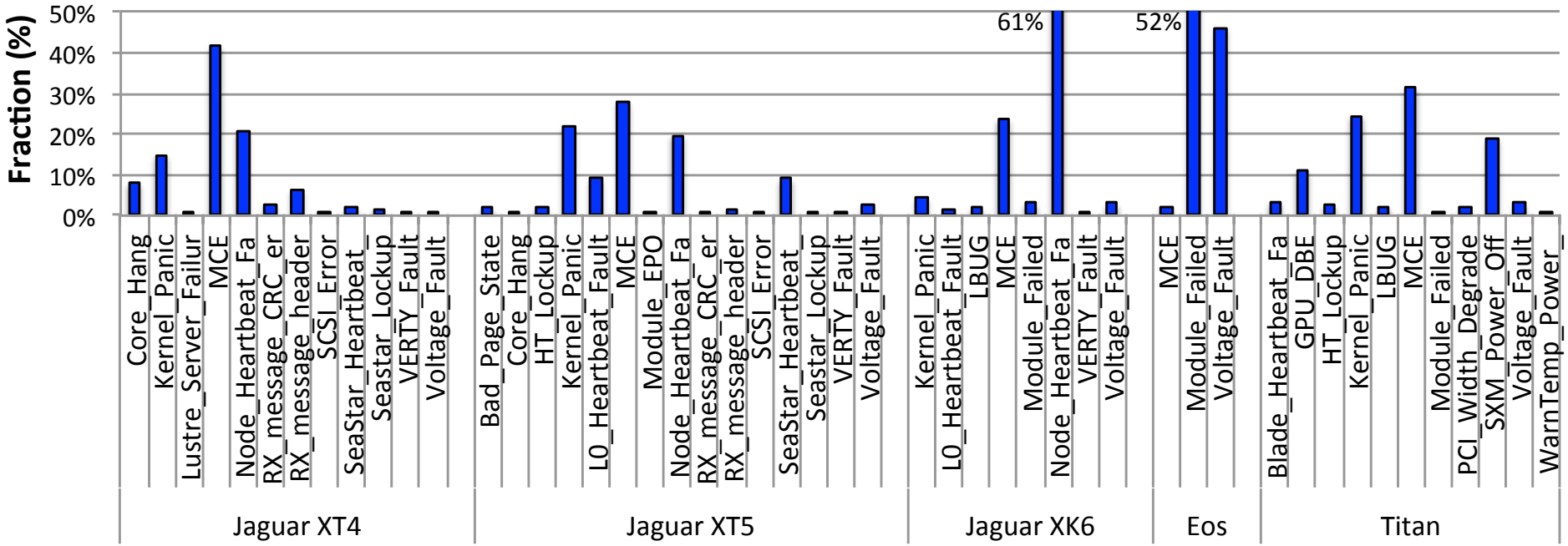


Figure: Each system goes through phases of high and low stability due to continuous efforts of system administrators to improve overall system reliability

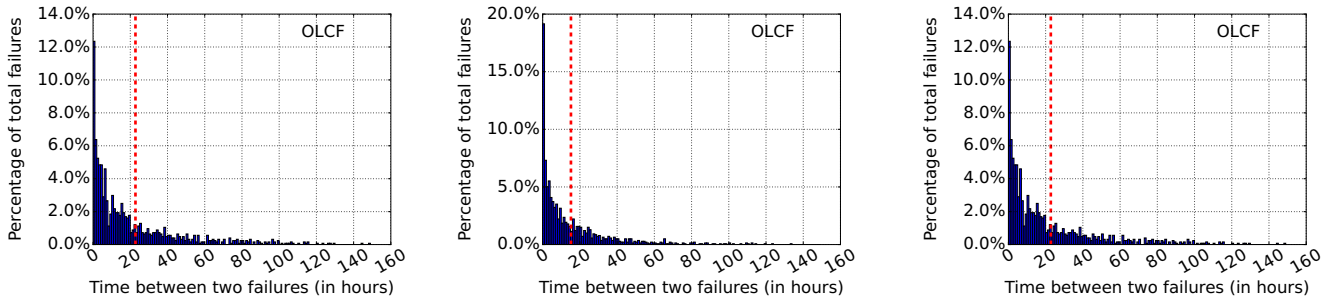
Saurabh Gupta, Devesh Tiwari, Tirthak Patel, and Christian Engelmann. **Reliability of HPC systems: Large-term Measurement, Analysis, and Implications.** SC'17. DOI 10.1145/3126908.3126937.

Characterizing Supercomputer Faults, Errors and Failures



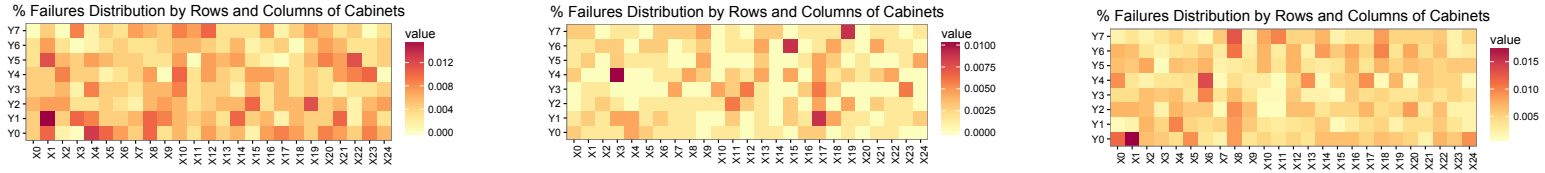
Fraction of each failure type on the studied systems

Characterizing Supercomputer Faults, Errors and Failures



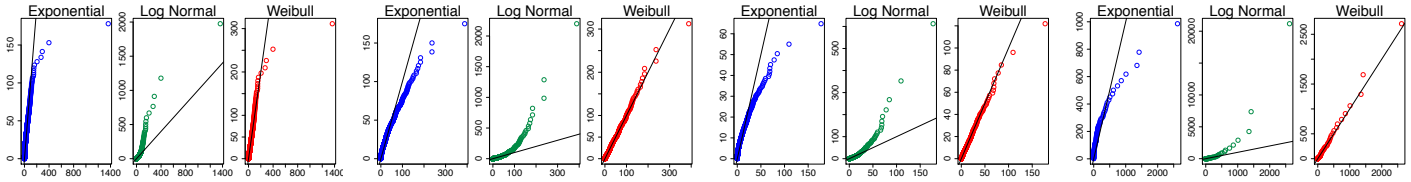
(a) Jaguar XT4 (b) Titan (c) Eos

Failure inter-arrival time for 3 studied systems (MTBF as red vertical line)



(a) Jaguar XT5 (b) Jaguar XK6 (c) Titan

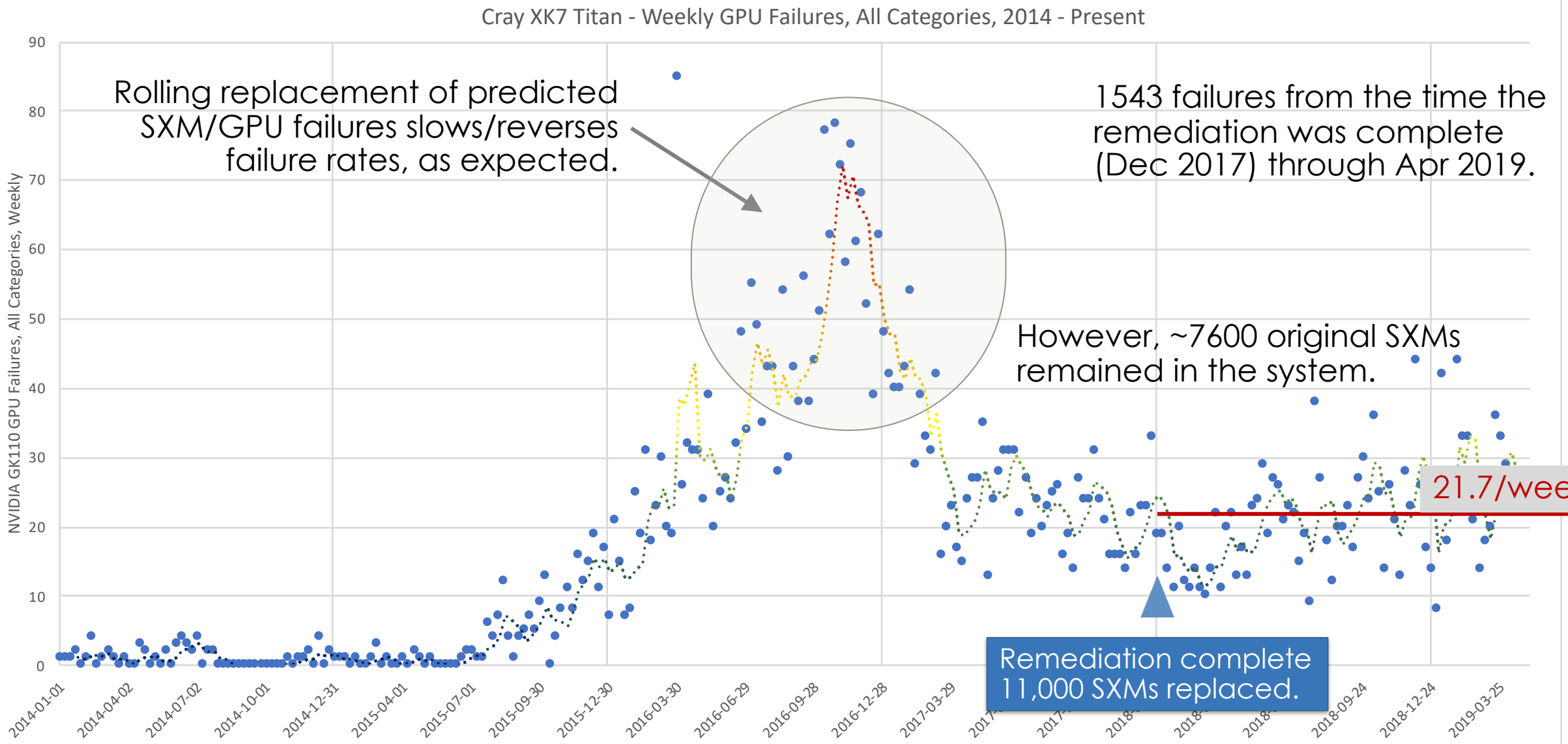
Spatial distribution of failures among cabinets for 3 studied systems



(a) Jaguar XT4 (b) Jaguar XT5 (c) Jaguar XK6 (d) Eos

QQ-plots showing goodness of fit for the failure inter-arrival times for 4 studied systems with different failure probability density functions (Weibull fits best)

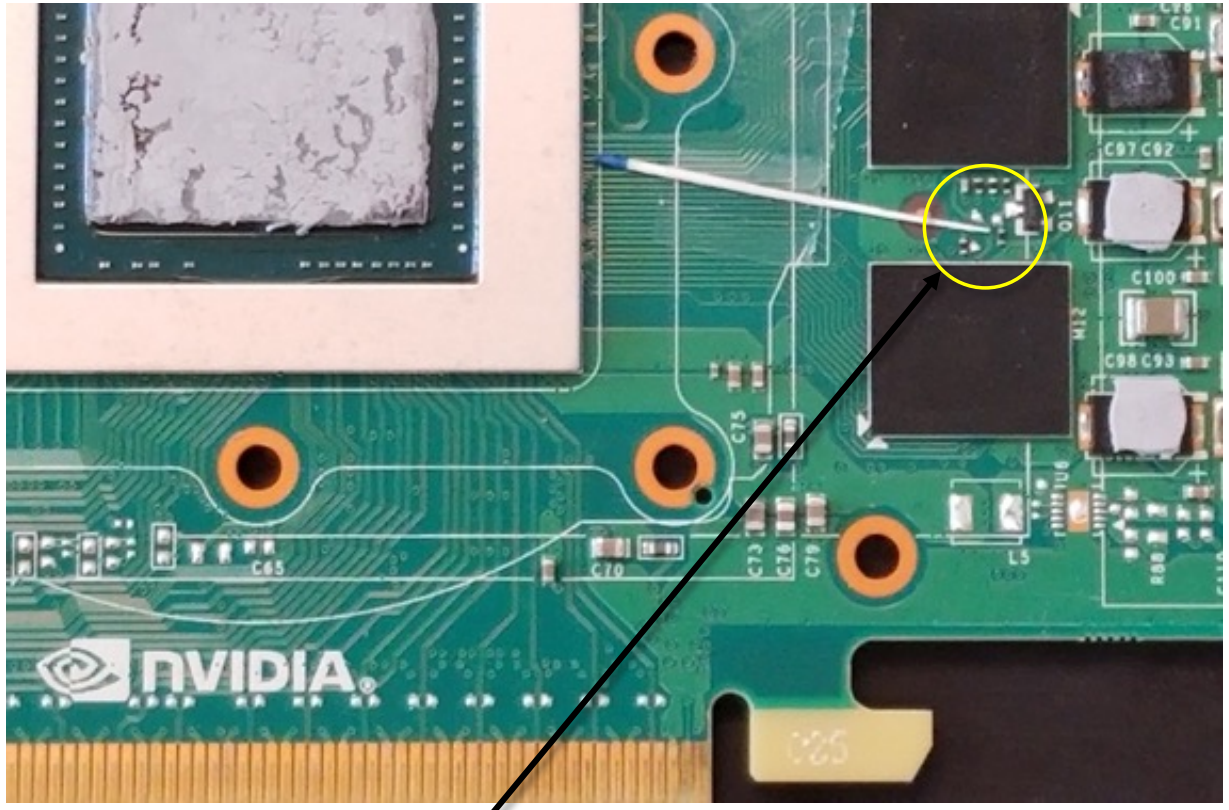
Cray XK7 Titan – Weekly GPU Failures



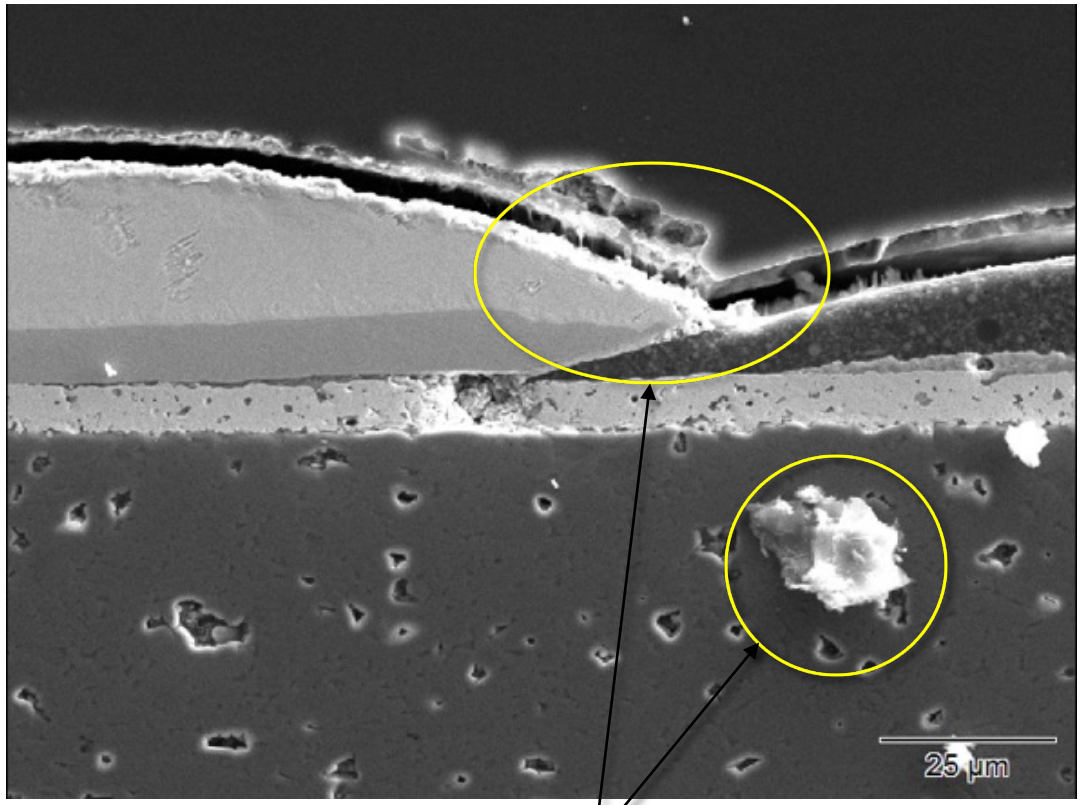
OAK RIDGE National Laboratory

G. Ostrouchov, D. Maxwell, R. Ashraf, C. Engelmann, Mallikarjun Shankar, and James Rogers. **GPU Lifetimes on Titan Supercomputer: Survival Analysis and Reliability.** SC'20. DOI 10.1109/SC41405.2020.00045.

Root Cause: Non-ASR Components on SXM GPU



NVIDIA SXM – Location of a non-ASR

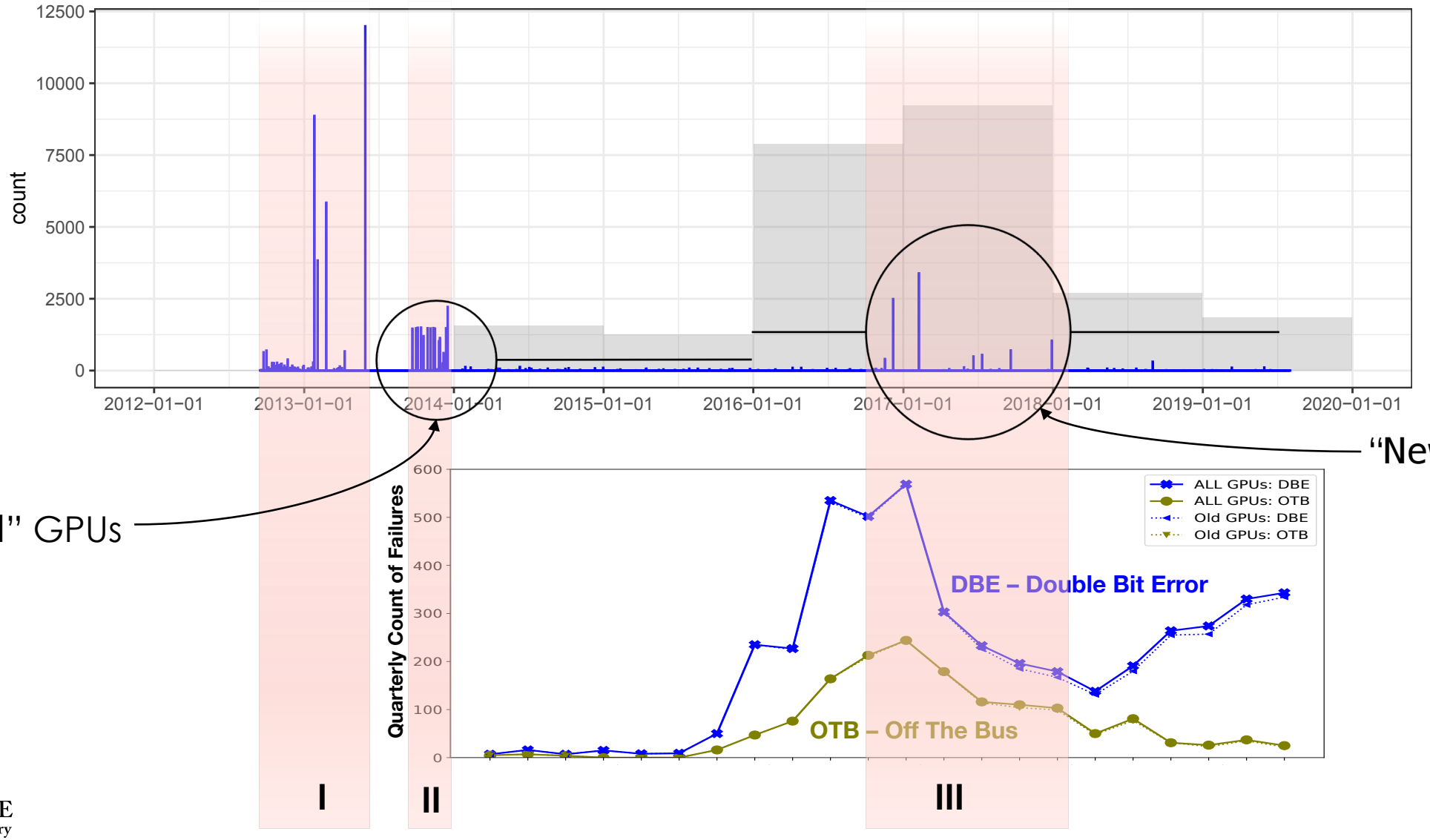


Silver-sulfide corrosion
"Flowers-of-Sulfur"

ASR = Anti-Sulfur Resistor

GPU Failures and Replacements in ORNL's Titan

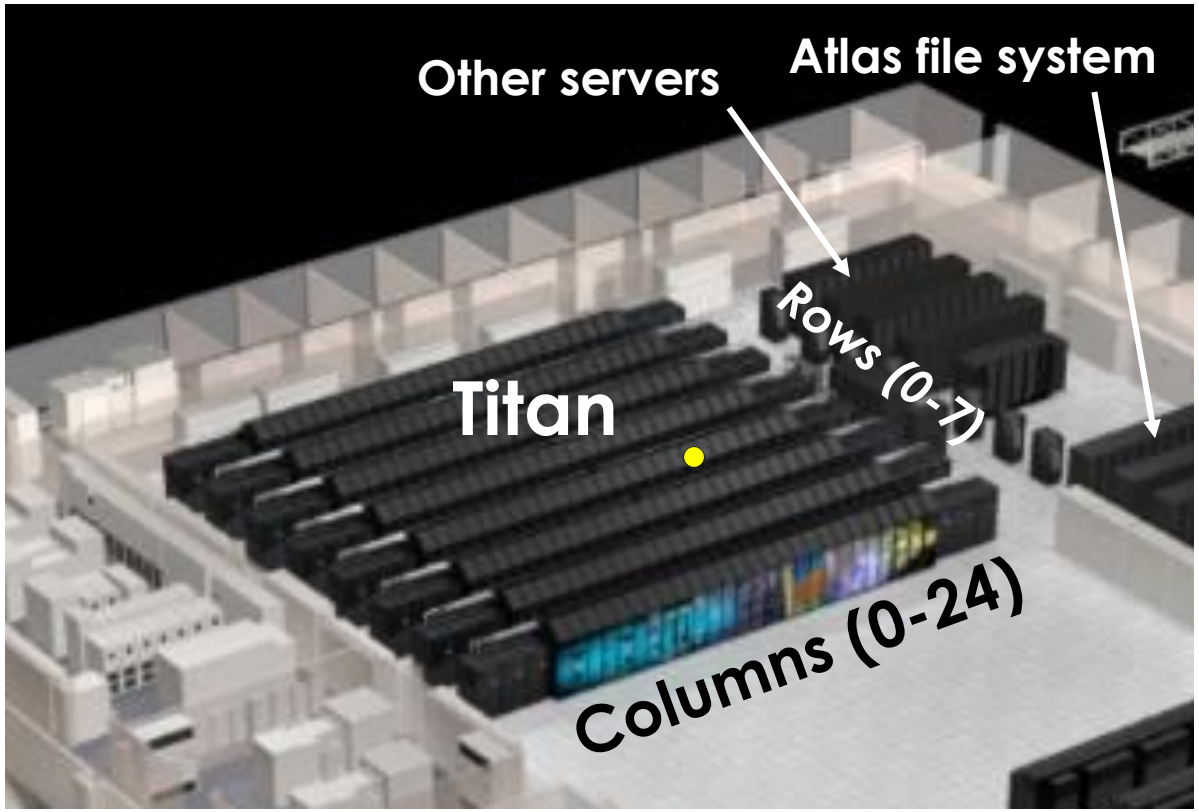
GPU swaps detected at inventories (narrow blue) and yearly sum totals for 2014 and later (wide gray)



OAK RIDGE National Laboratory

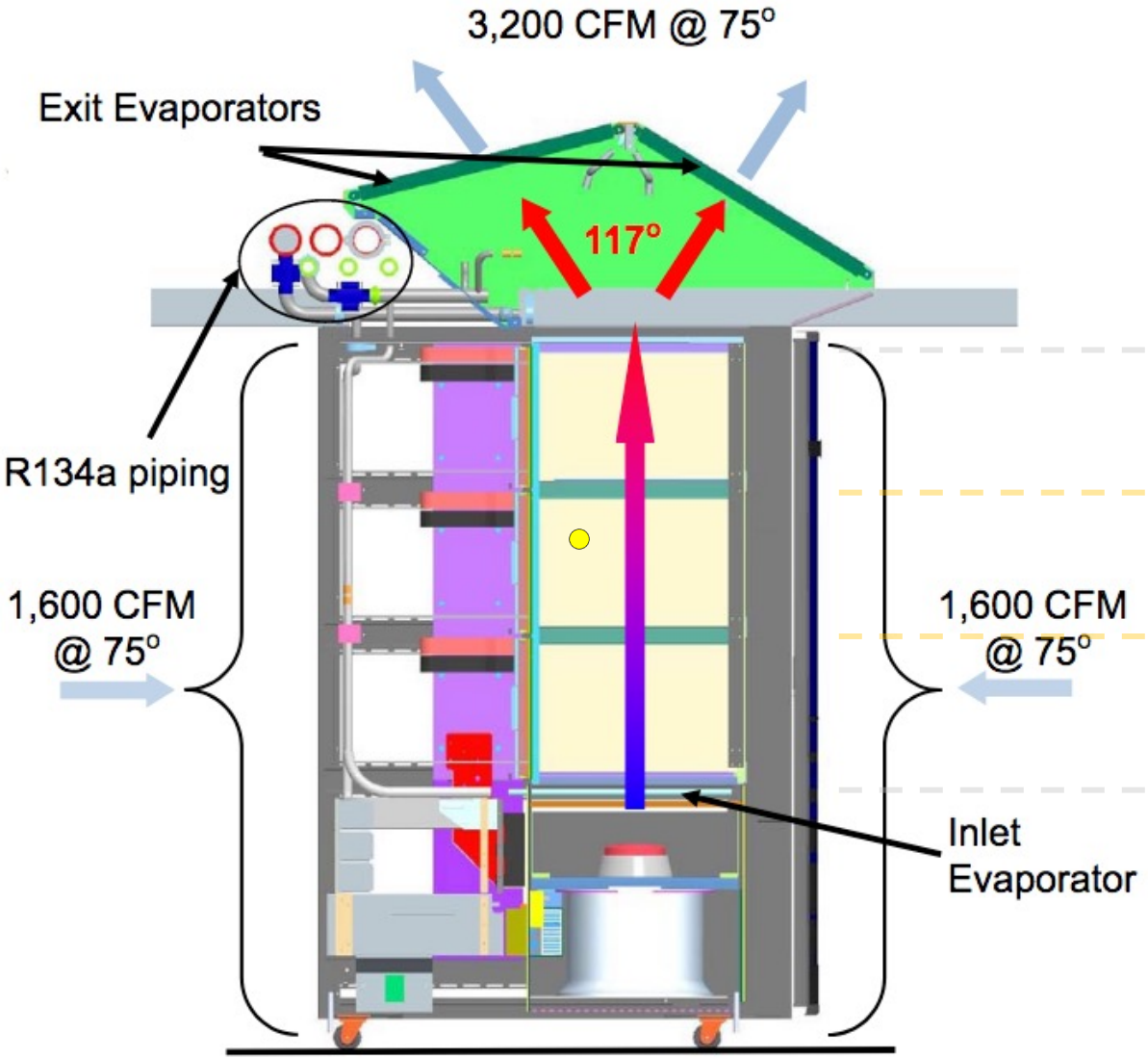
Machine Room Layout: GPU Locators

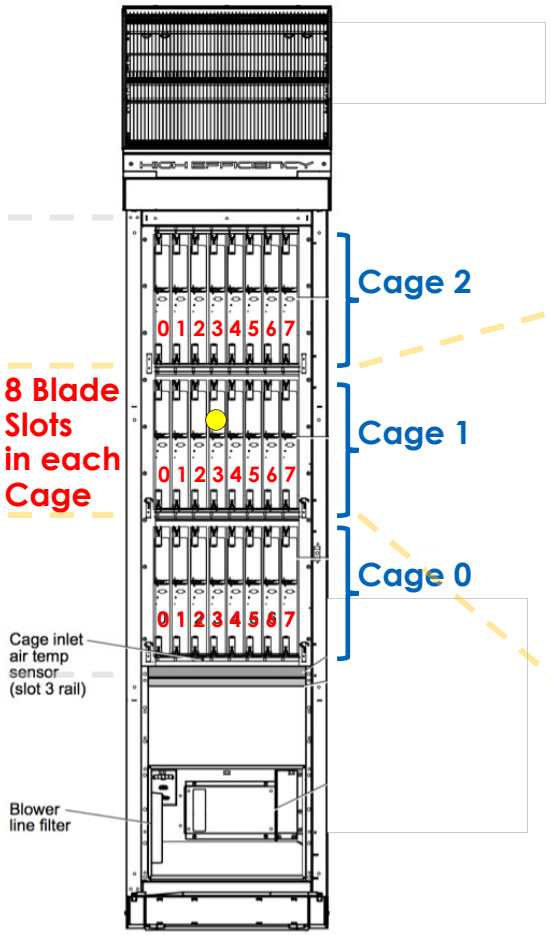
```
c##-#c#s#n#
a     a l o
b     g o d
i     e t e
n
e
t
```

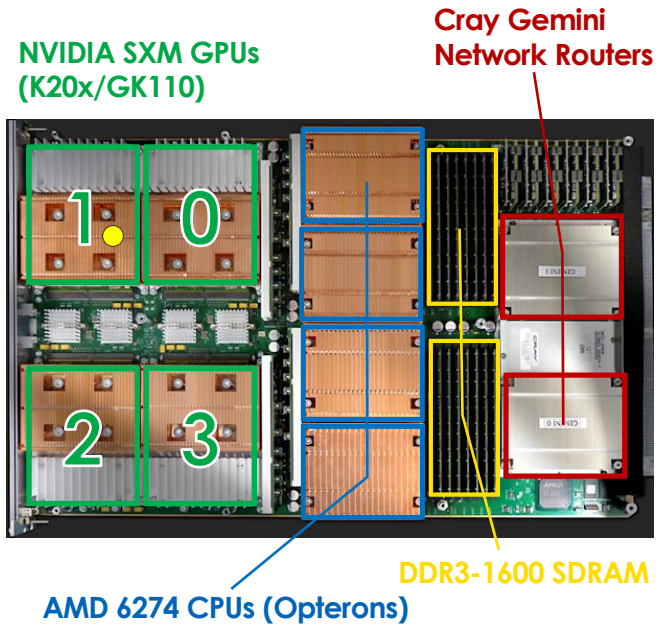


- **c17-4c1s3n1** = cabinet in column **17** – row **4**

Cabinet Mechanical Packaging: Locating a GPU

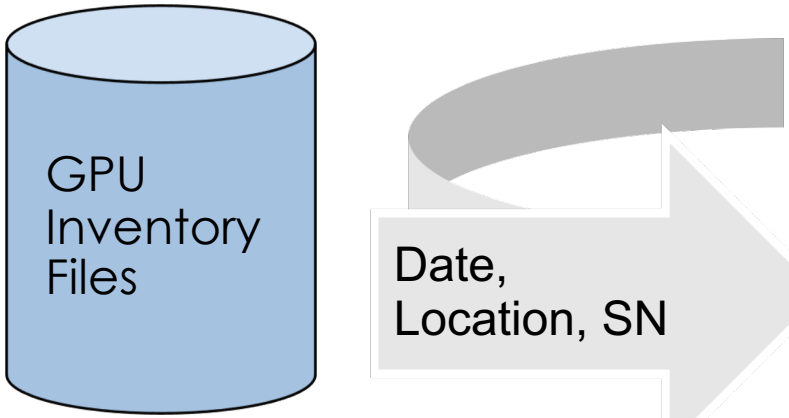


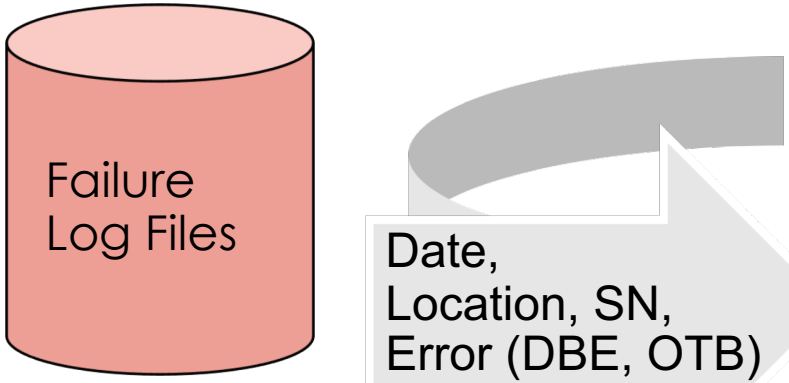


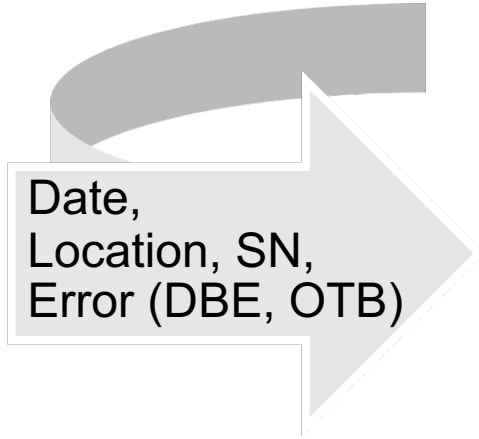


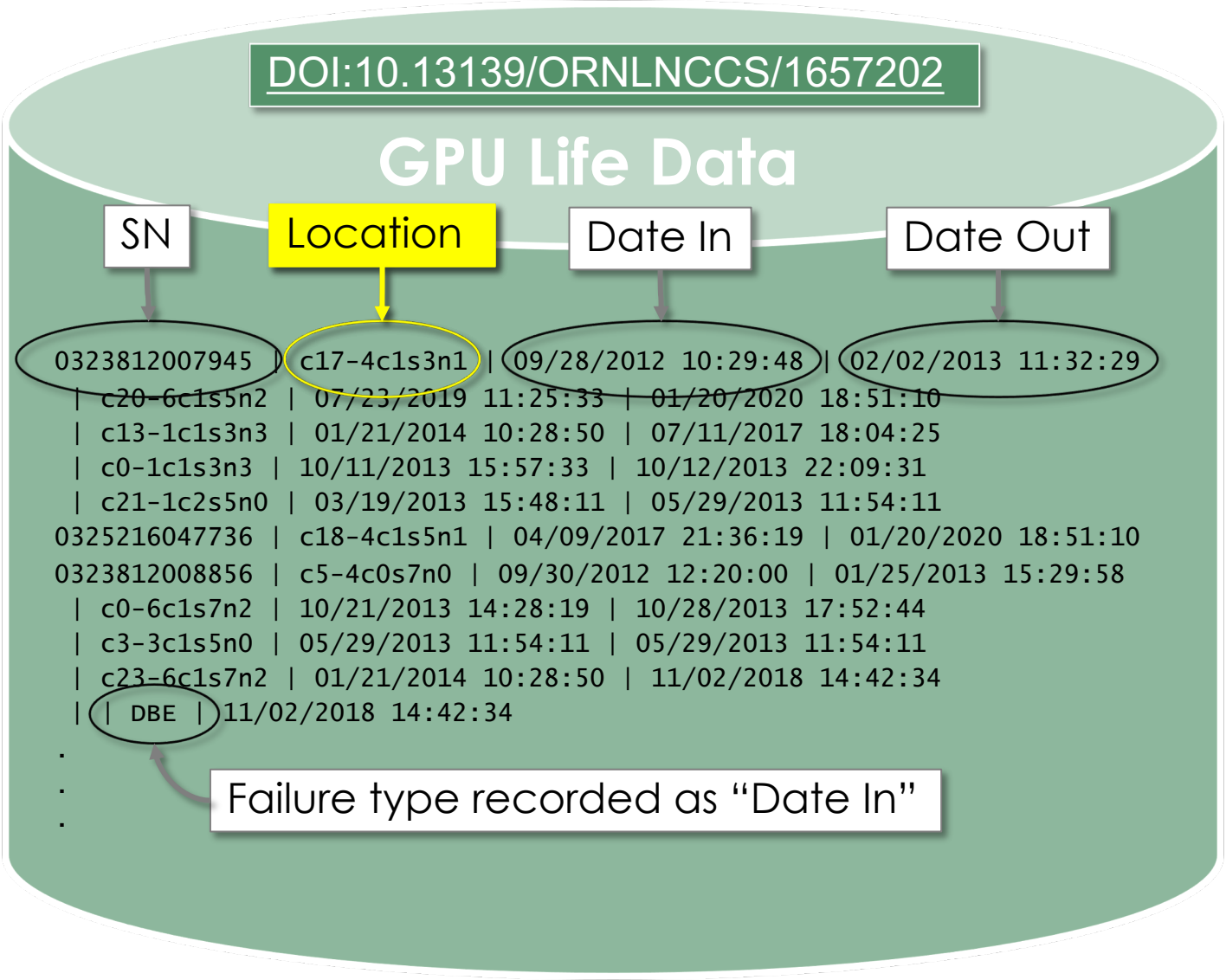
- **c17-4c1s3n1** = cage **1**, slot **3**, node **1**

GPU Life Data Built Incrementally from Two Sources





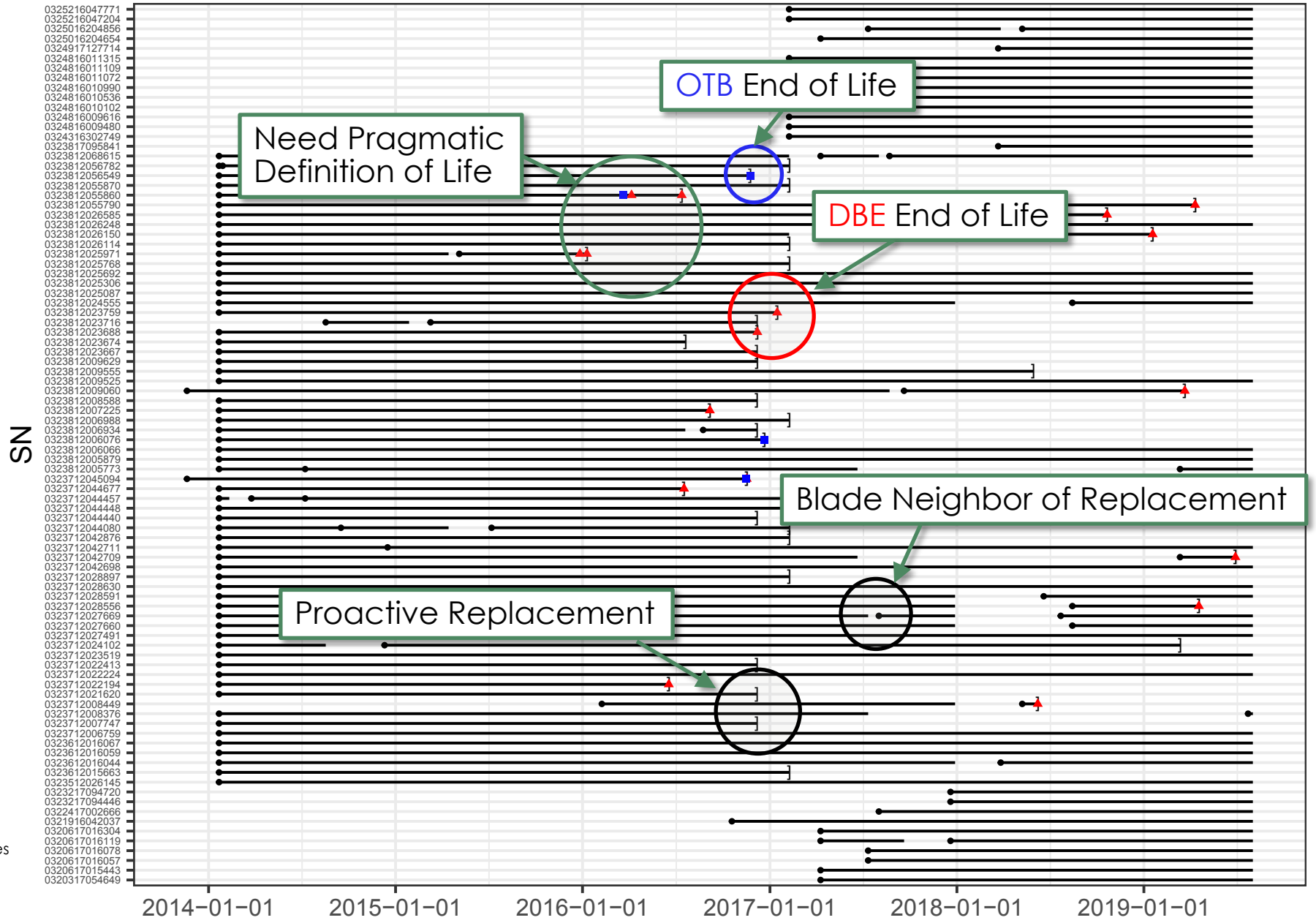




GPU Life Visualization: Serial Number View

Critical for:

- Understanding data
- Defining GPU Life
- Data processing verification

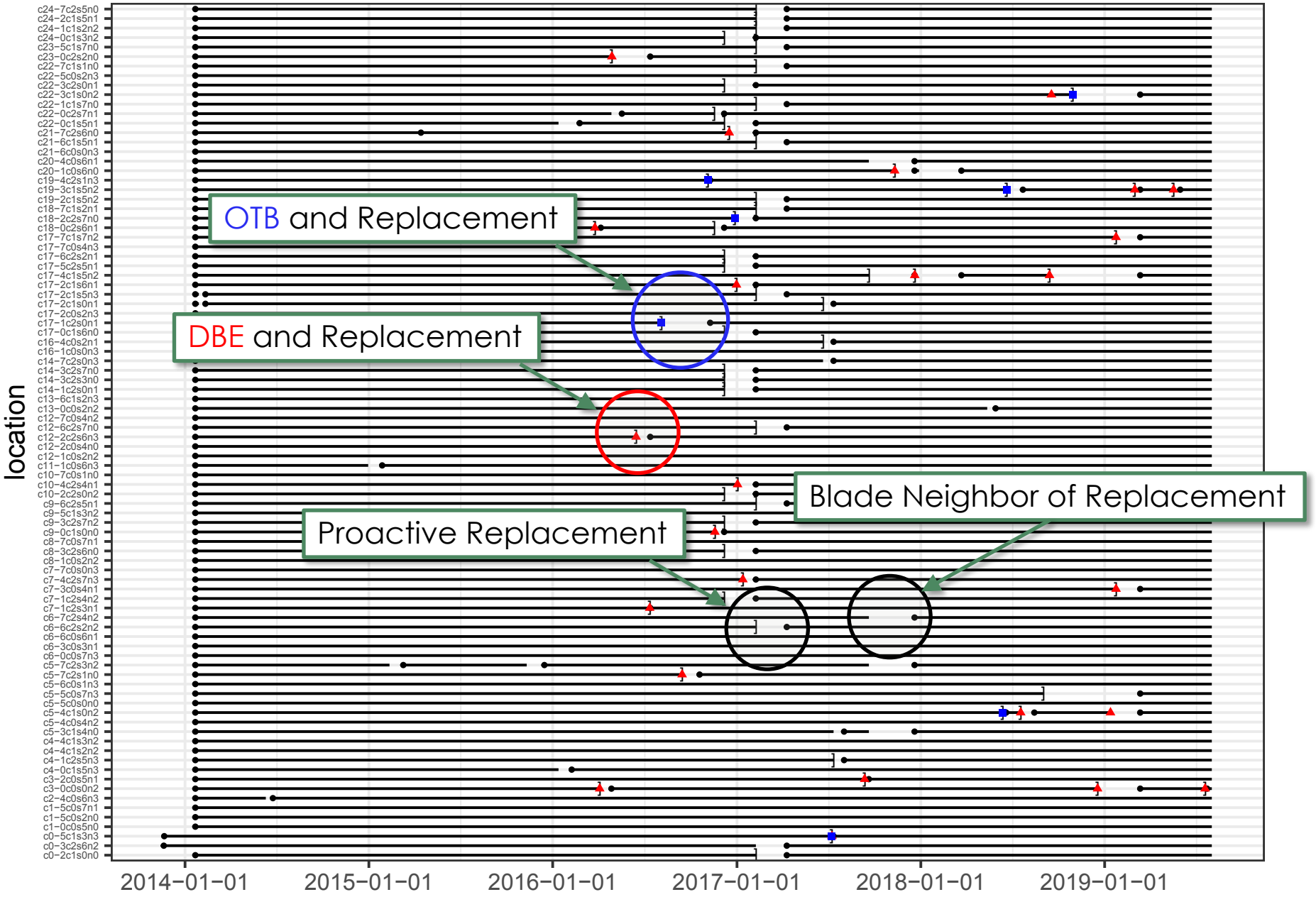


Produced in R via `ggplot2` and `lubridate` packages

GPU Life Visualization: Location View

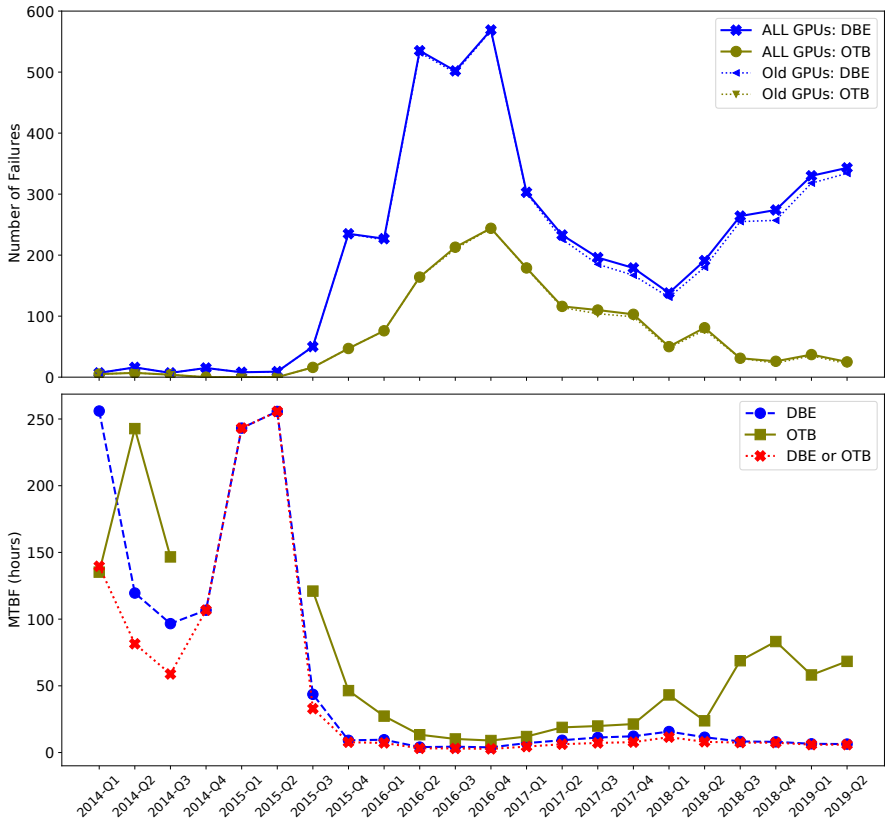
Critical for:

- Understanding data
- Defining GPU Life
- Data processing verification

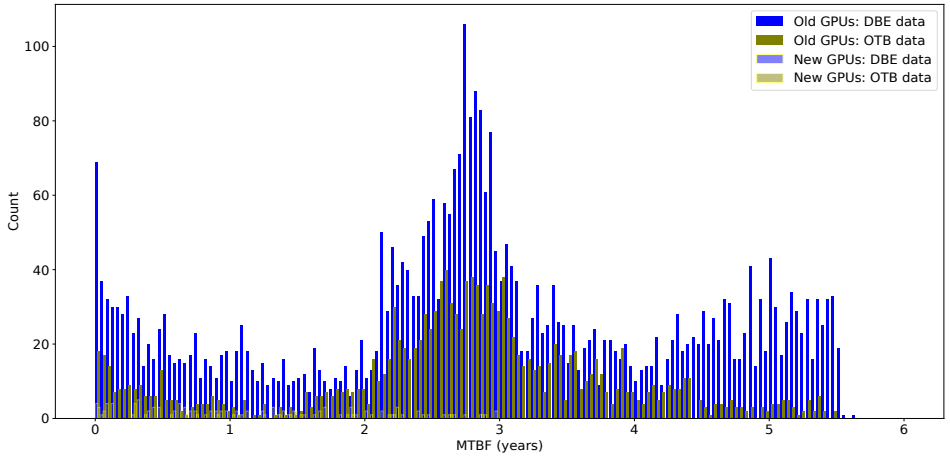


Produced in R via `ggplot2` and `lubridate` packages

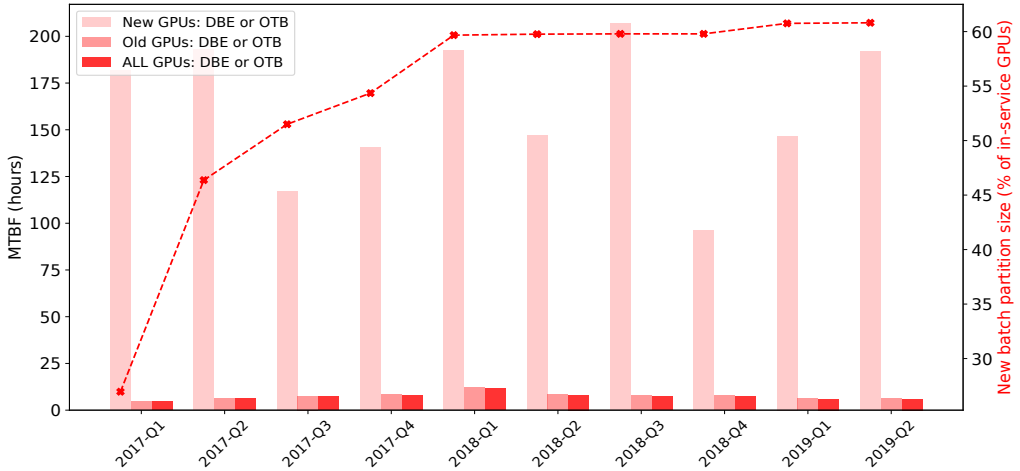
Traditional Reliability in HPC is Focused on MTBF



System-wide Reliability: Quarterly number of failures (top) and MTBF (bottom).



Individual GPU Reliability: MTBF histogram for units that had at least one failure. Interpret carefully: lacks information from units with no failures!



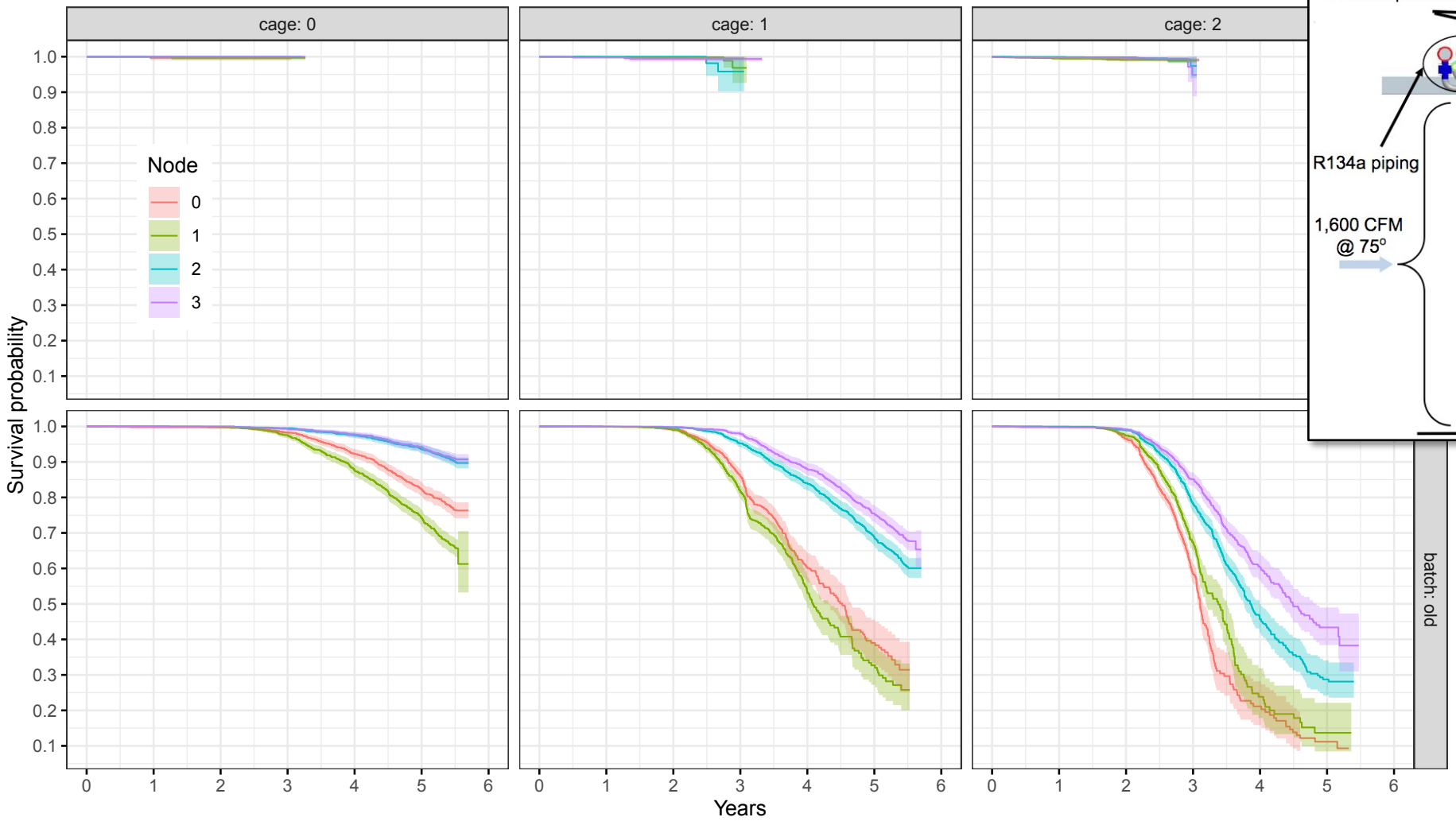
Old–New as Two Partitions: MTBF differs by 12x factor!

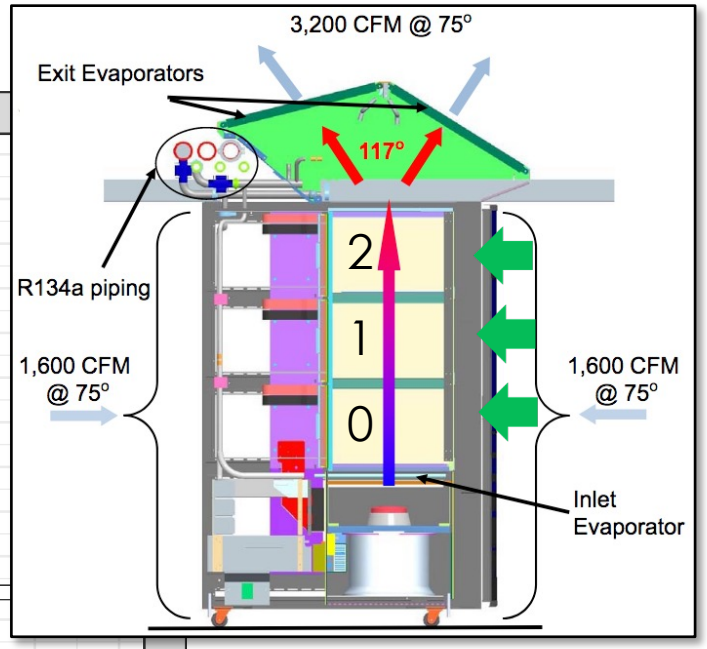
Kaplan-Meyer Survival Analysis

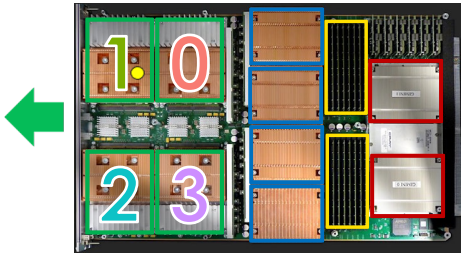
- Commonly used in Biostatistics and Biomedical research*
- Nonparametric
  - If T is failure time and $F(t) = \Pr\{T < t\}$ is the cumulative failure distribution function
  - Then the survival probability, $S(t) = \Pr\{T \geq t\} = 1 - F(t)$, is its complement
  - Recursive computation $S(t_2) = \Pr\{\text{survive from } t_1 \text{to } t_2\}\, S(t_1)$
- Able to incorporate censoring
- Split population into groups
- Available uncertainty estimate

*E. L. Kaplan and P. Meier, "Nonparametric estimation from incomplete observations," *Journal of the American Statistical Association*, vol. 53, no. 282, pp. 457–481, 1958.

Cage and Node Effect Explainable by Airflow in Cabinet







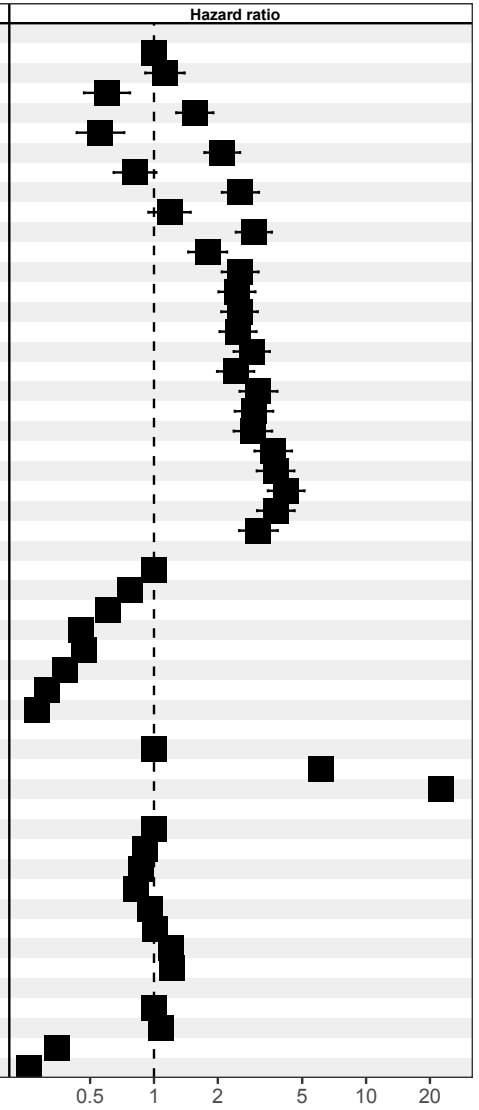
Cox Proportional Hazards Regression Model

- Commonly used in Biostatistics and Biomedical research*
- Able to adjust for covariate effects
- Each GPU is like a patient, affected by its location (treatment)
- The hazard for patient k is $H_k(t) = H_0(t)e^{\sum_1^n \beta_i x_i}$
  - Base hazard rate, $H_0(t)$, multiplied by a function of covariates (hazard coefficient)
- Semiparametric model
  - Baseline hazard is nonparametric (no functional shape assumption)
  - Hazard coefficient is a parametric function of covariates
- Assumes hazards are proportional

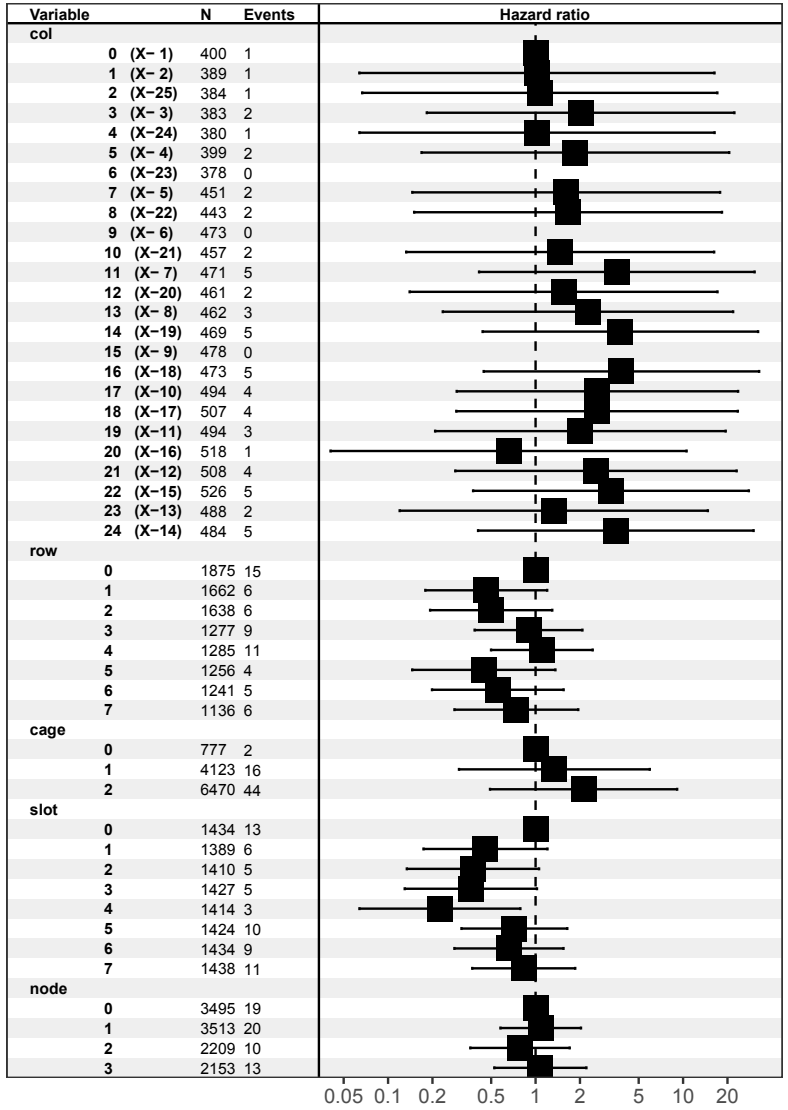
*D. R. Cox, "Regression models and life-tables," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 34, no. 2, pp. 187– 220, 1972.
We use R packages `survival` and `survminer`.

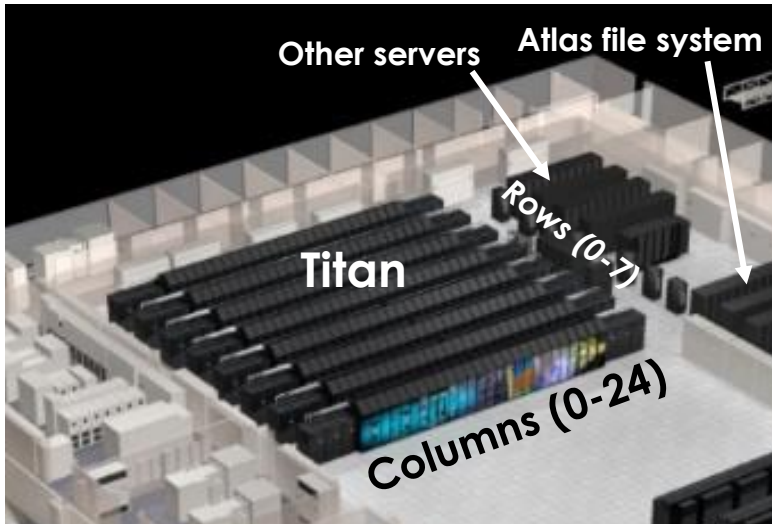
Strong Signal in old Batch, Pattern Similar to K-M Analysis

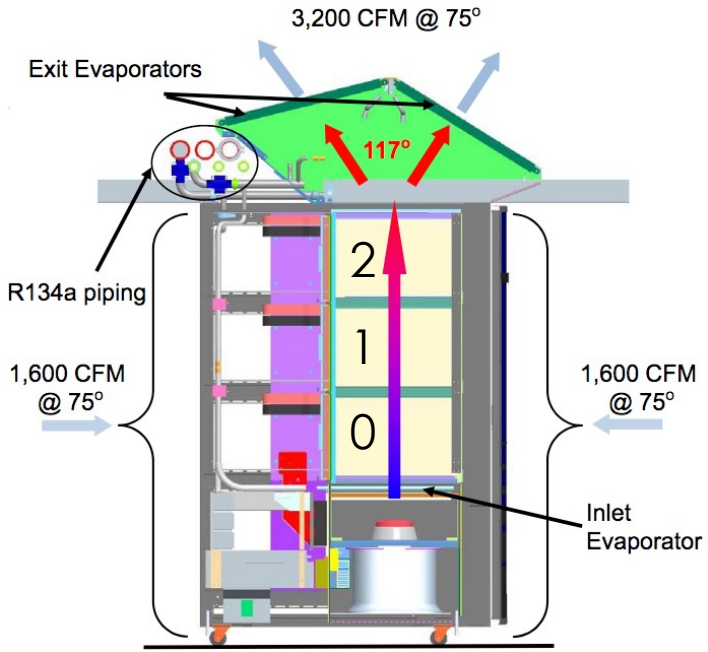
Variable		N	Events
col			
0	(X− 1)	744	173
1	(X− 2)	759	169
2	(X−25)	772	97
3	(X− 3)	739	217
4	(X−24)	755	88
5	(X− 4)	759	266
6	(X−23)	752	125
7	(X− 5)	744	225
8	(X−22)	758	131
9	(X− 6)	758	259
10	(X−21)	739	180
11	(X− 7)	760	233
12	(X−20)	756	228
13	(X− 8)	759	242
14	(X−19)	743	228
15	(X− 9)	753	257
16	(X−18)	754	224
17	(X−10)	758	207
18	(X−17)	758	194
19	(X−11)	751	192
20	(X−16)	766	217
21	(X−12)	759	218
22	(X−15)	755	244
23	(X−13)	743	220
24	(X−14)	743	193
row			
0		2363	795
1		2352	639
2		2351	565
3		2363	596
4		2371	687
5		2334	577
6		2339	472
7		2364	696
cage			
0		6240	1107
1		6275	1746
2		6322	2174
slot			
0		2355	637
1		2353	592
2		2349	575
3		2352	553
4		2364	606
5		2349	633
6		2357	694
7		2358	737
node			
0		4712	1388
1		4722	1494
2		4697	1195
3		4706	950



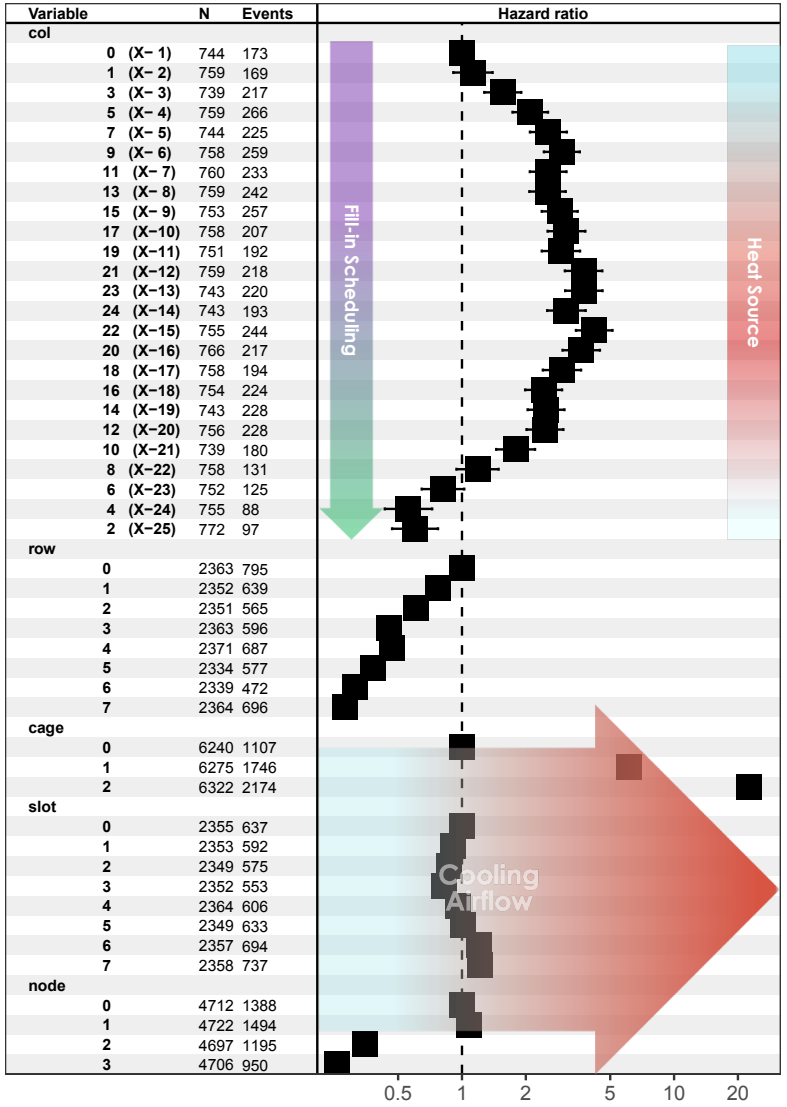
Variable		N	Events
col			
0	(X− 1)	400	1
1	(X− 2)	389	1
2	(X−25)	384	1
3	(X− 3)	383	2
4	(X−24)	380	1
5	(X− 4)	399	2
6	(X−23)	378	0
7	(X− 5)	451	2
8	(X−22)	443	2
9	(X− 6)	473	0
10	(X−21)	457	2
11	(X− 7)	471	5
12	(X−20)	461	2
13	(X− 8)	462	3
14	(X−19)	469	5
15	(X− 9)	478	0
16	(X−18)	473	5
17	(X−10)	494	4
18	(X−17)	507	4
19	(X−11)	494	3
20	(X−16)	518	1
21	(X−12)	508	4
22	(X−15)	526	5
23	(X−13)	488	2
24	(X−14)	484	5
row			
0		1875	15
1		1662	6
2		1638	6
3		1277	9
4		1285	11
5		1256	4
6		1241	5
7		1136	6
cage			
0		777	2
1		4123	16
2		6470	44
slot			
0		1434	13
1		1389	6
2		1410	5
3		1427	5
4		1414	3
5		1424	10
6		1434	9
7		1438	11
node			
0		3495	19
1		3513	20
2		2209	10
3		2153	13





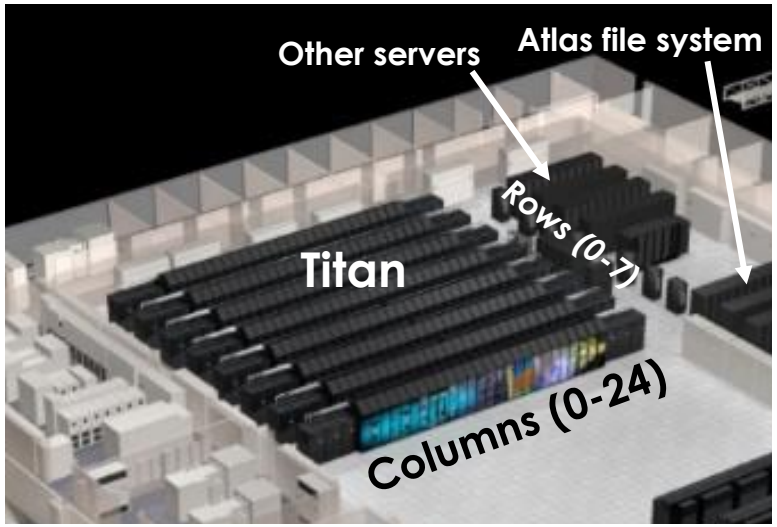


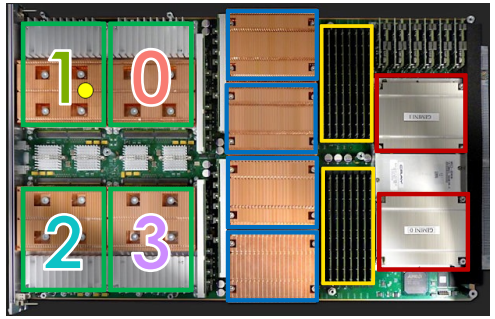
Cooling Architecture and Scheduling Affect Reliability

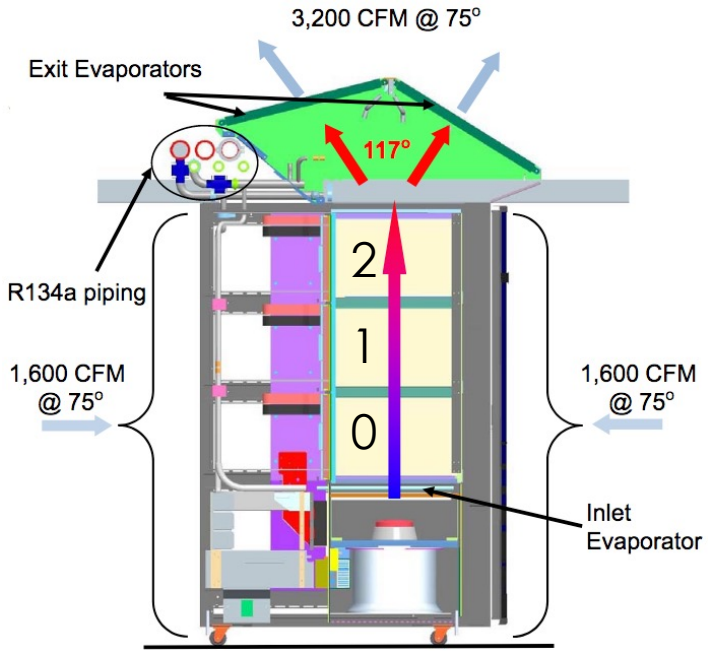


Reordered columns based on Torus network

- Reordered column by fill-in scheduling torus coordinate
- Hot room spot due to other servers and Atlas file system
- Higher cages ran hotter
- Node 0 and 1 ran hotter







Future Research and Development Needs

- We need to design the HPC hardware/software ecosystem to be able to deal with high error and failure rates, expected and unexpected!
  - Resilience research and development is, in part, risk mitigation against the unexpected
  - There is always a cost/benefit trade-off that needs to be considered
  - Resilience mitigation mechanisms should be a toolbox with lots of options
- Resilience should be by design and not as an afterthought
  - Resilience is a crosscutting issue that should be considered everywhere

Short-term Future Research and Development Needs

- Portable system/center monitoring and analysis solutions
  - Collecting the right metrics
  - Proper identification of faults (online)
  - Fast and accurate root cause analysis (online and offline)
  - Using advanced statistical techniques and ML
  - ❖ There is some ongoing work at the facilities, but it is disconnected from recent research
- Low-overhead software mitigation techniques (beyond global checkpoint/restart)
  - OS/R and programming model runtime resilience features
  - Resilience for workflows
  - ❖ There is some ongoing work in fault tolerant programming models, but it is underfunded and community adoption is low

Other Future Research and Development Needs (1/2)

- Smart systems and facilities
  - Autonomous resource management that considers the system/facility state and the involved trade-offs
  - Automatic adaptation of systems and facilities in real-time to emerging reliability issues using AI
  - Machine-in-the-loop operational intelligence (OODA loop to improve productivity and lower costs)
- Resilience in federated/distributed/complex computing environments
  - Instruments/laboratories using edge and center computing for science feedback on experiments
  - Real-time and urgent computing that has specific resilience needs
- Understanding the resilience problem in non von Neumann architectures
  - E.g., neuromorphic computing

Other Future Research and Development Needs (2/2)

- Resilience by design
  - Design space exploration that considers resilience in addition to performance and power/energy
  - Performance/energy/resilience co-design
  - Programming for resilience (higher-level abstractions and programming models)
- Resilient algorithms and probabilistic/approximate computing
  - Algorithm-based fault tolerance
  - Coded computing
  - Naturally resilient algorithms
- End-to-end resilience (integrity of data and computation)
- Intersection between cyber security and resilience

Questions?