



---

# Defining and Measuring Supercomputer Reliability, Availability, and Serviceability (RAS)

Jon Stearley  
Sandia National Laboratories

April 29, 2005  
Linux Clusters Institute 2005 Conference

Updated paper at <http://www.cs.sandia.gov/~jrstear/ras/defs.pdf>



---

## Outline

**Problem:**

Can't agree on terms!?!?

**Proposal:**

State model

Definitions

Measurements

**A primary goal of this work is foster discussion:**

**ASK QUESTIONS at any time!!!**

Updated paper at <http://www.cs.sandia.gov/~jrstear/ras/defs.pdf>



# Expensive Supercomputers (but poor RAS?)

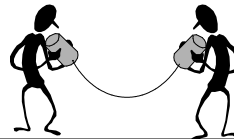


The lack of standardized RAS definitions and measurements:

- Obscures meaningful discussion of the real issues
- Delays real RAS improvements
- Increases costs  
(in all phases: procurement, operation, and end-of-life determination)



My supercomputer is SO RELIABLE!!!



that depends on what you mean by "fault"...



## e.g. Procurement



“Failure of single component will not cause the full system to become unavailable...”

(Red Storm, Purple, Thunder, Q)

“MTBI for full system shall be greater than 50 hours... for a single application”

“MTBI for full system (reboot) shall be greater than 100 hours...”


(over how many samples?)

(Red Storm)


“100 hour capability jobs (90% of system) will successfully complete 95% of the time...” (= 79 days of failure-free computing?)

“Over any 4 week period, the system will have an effectiveness level of at least 95%...”

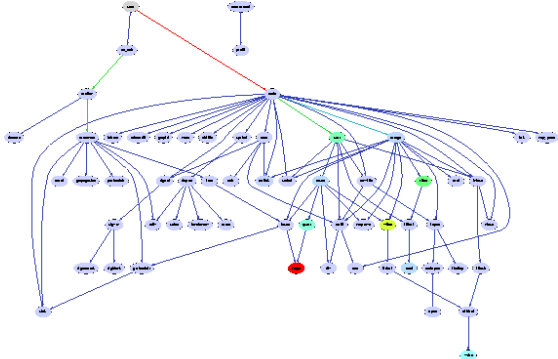
(Purple)




# System State




**“A computer is in one of two situations. It is either known to be bad or it is in an unknown state.”**  
**- Mike Levine (PSC)**






Is the “system” “up” (yet)?

Updated paper at <http://www.cs.sandia.gov/~jrstea/ras/defs.pdf>



# e.g. Operation



Excerpt from <http://www.nersc.gov/users/status/AvailStats/>:

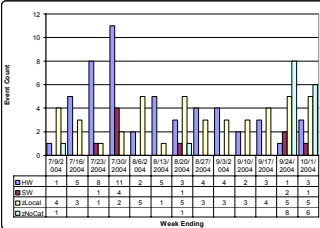
System Availability Details

FY05 - FEBRUARY 2005										
System	Scheduled		Un-Scheduled			Overall Avail %	Sched Avail %	MTBI (Hours)	MTTR (Hours)	MTBF (Day:Hour:Min)
	HW	SW	HW	SW	Other					
Parallel	99.61%	99.22%	99.87%	99.21%	100.00%	97.92%	99.07%	226	4.8	9 05:46
Storage	99.26%	99.39%	99.67%	99.91%	100.00%	98.23%	99.58%	207	3.4	8 13:00
File Servers	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%			
Math/Vis Servers	100.00%	99.82%	100.00%	99.85%	100.00%	99.66%	99.84%	1208	4.00	25 11:48

\*Mean Time Between Interruptions = Total wall clock hours/total number of downtime periods  
 \*\*Mean Time To Restoral = Total downtime hours/total number of downtime periods  
 \*\*\*Mean Time Between Failures = Total wall clock hours - Total downtime hours/Total downtime hours - 1

### Excerpts from LLNL ASC White “Six Sigma Report”

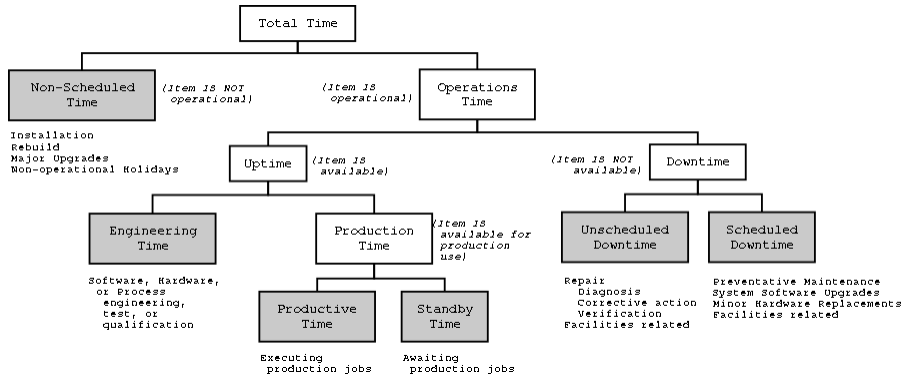
Estimated MTBI					
sector	week ending item	8/27/04		9/3/04	
snow**white*frst+ice					
	# failures TOTAL	7		7	
	# failures HW	4	57%	4	57%
	# failures SW	0	0%	0	0%
	# failures LOCAL	3	43%	3	43%
	# nodes	624		624	
	# hours	168		168	
	# node-hours	104832		104832	
	MTBF (hr)	24		24	
	MTBF (hr/node)	14976		14976	





# State Model (adapted from SEMI-E10)

- Items are **always** in one of the **six** basic states (grey boxes).
- Time is hierarchically categorized (white boxes).



Updated paper at <http://www.cs.sandia.gov/~jrstea/ras/defs.pdf>

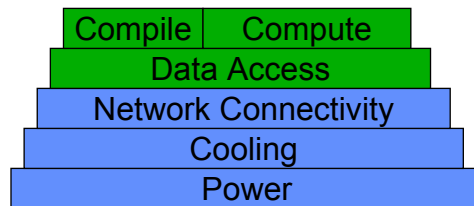


## Definitions: Reliability

### Reliability (IEEE):

The probability that an *item* will:

- **function** without failure
- **under stated conditions**
- **for a specified amount of time.**



$$R(t) = e^{-\lambda t}, \lambda = 1/MTBF?$$



*item* – an all-inclusive term to denote any level of unit (e.g. component, system, etc)



# Cause vs Effect: Failure vs Interrupt

**Failure** – the termination of the ability of an item to perform a required function.  
External corrective action is required into order to restore this ability (e.g. manual reboot, repair, replacement).

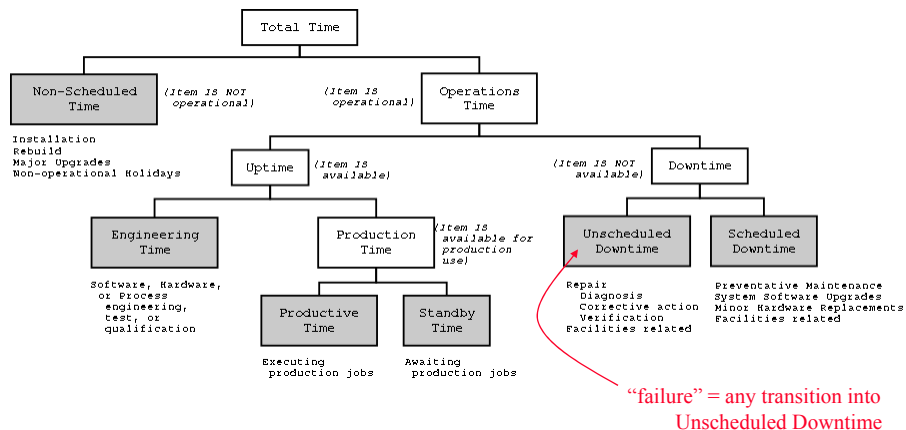
**Interrupt** – the suspension of a process to handle an event external to the process.  
(yes, this ISO9000 definition is insufficient...)

**Failures regard items; interrupts regard work.**  
**Failures *may* cause interrupts.**

Updated paper at <http://www.cs.sandia.gov/~jrstea/ras/defs.pdf>

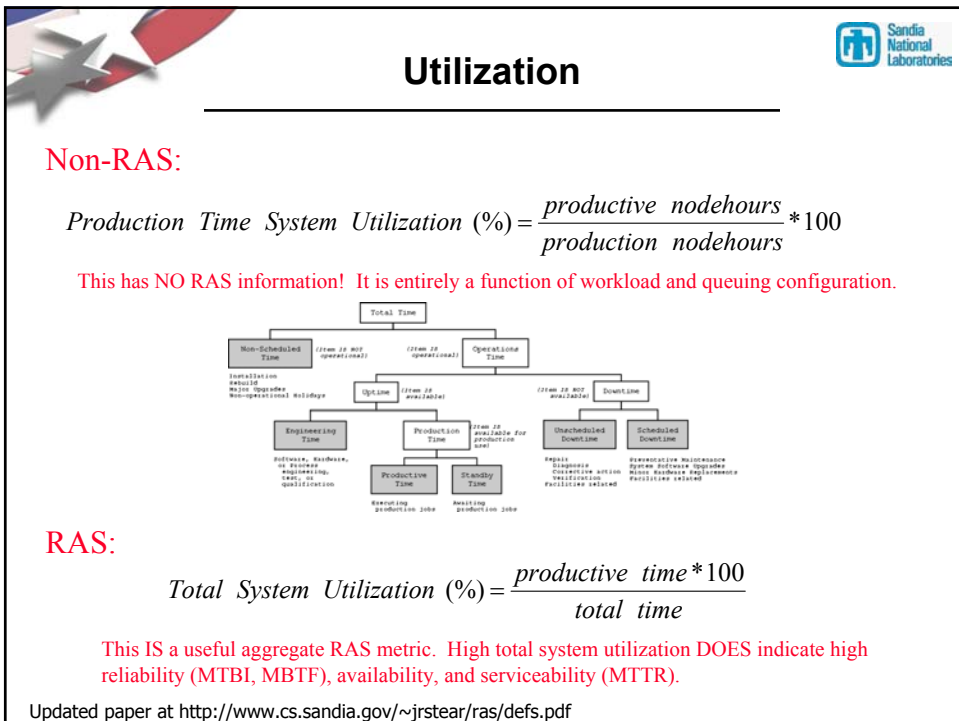
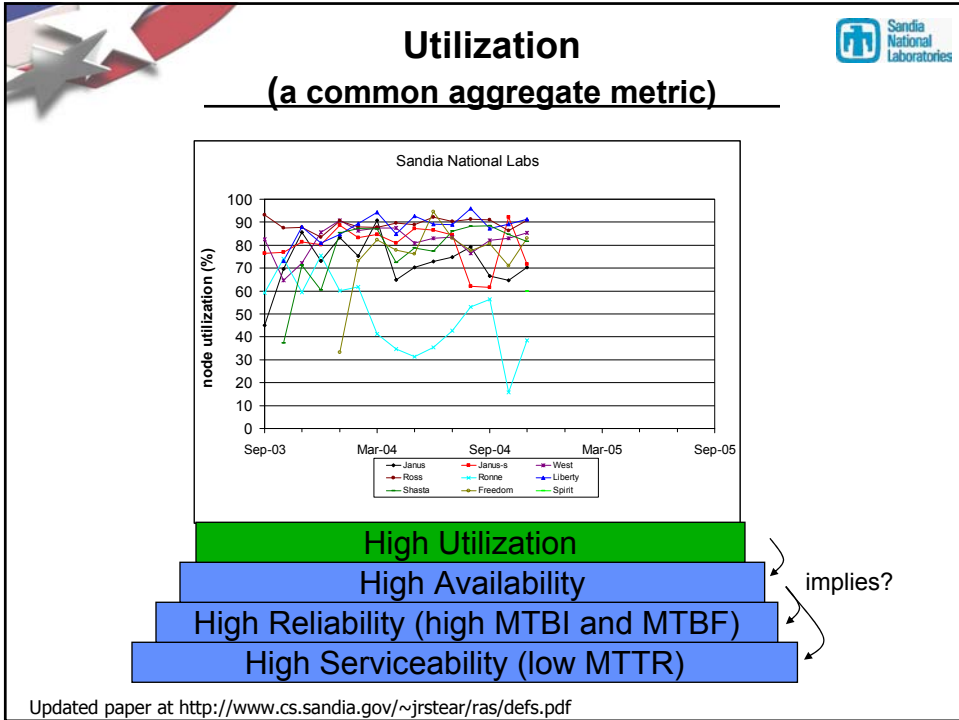


# State Model



A general model, requiring system-specific details for application.

Updated paper at <http://www.cs.sandia.gov/~jrstea/ras/defs.pdf>





# Mean Time Between Job Interrupts

Common:

$$MTBI = \frac{\text{total time}}{\text{number of interrupts}}$$

← Easy to calculate (but assumes downtime is negligible)

Proposed:

**Job Interrupt - The unexpected interruption of an active job.**

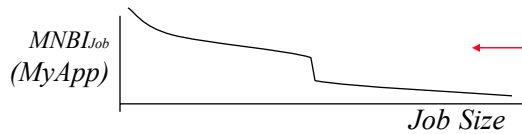
$$MTBI_{Job} = \frac{\text{production time}}{\text{number of job interrupts}}$$

← More precise (but no work info)

← More specific

$$MNBI_{Job} = \frac{\text{productive nodehours}}{\text{number of job interrupts}}$$

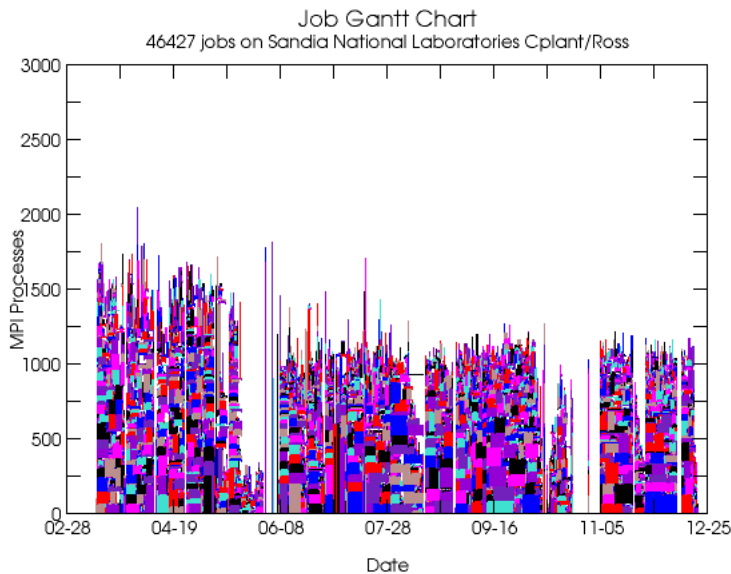
← Includes workload information



← Conceptual plot of how long an application is likely to run before experiencing an interrupt, as a function of job size



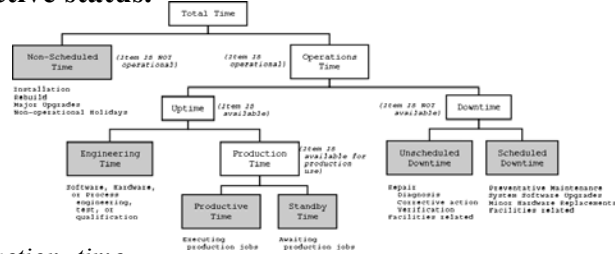
# Workload information is vital!





# Mean Time Between Node Failures

**Node Failure – an event requiring that the node ( a component) enter a downtime status before it can enter a productive status.**



$$MTBF_{Node} = \frac{\text{production time}}{\text{number of node failures}}$$

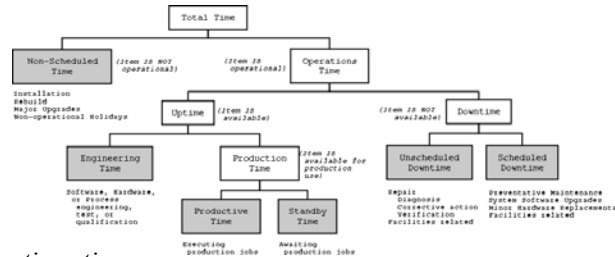
$$MNBF_{Node} = \frac{\text{productive nodehours}}{\text{number of node failures}} \quad \leftarrow \text{Includes workload information}$$

Updated paper at <http://www.cs.sandia.gov/~jrstea/ras/defs.pdf>



# Mean Time Between System Failures

**System Failure – an event requiring that the system (the majority of components) enter a downtime status before any component may enter a productive status.**



$$MTBF_{System} = \frac{\text{production time}}{\text{number of system failures}}$$

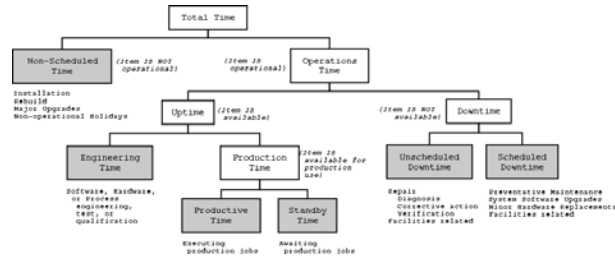
$$MNBF_{System} = \frac{\text{productive nodehours}}{\text{number of system failures}} \quad \leftarrow \text{Includes workload information}$$

Updated paper at <http://www.cs.sandia.gov/~jrstea/ras/defs.pdf>



# Mean Time Between Service Interrupts

**Service Interrupt – any event which disrupts full service to users (for any reason).**



$$MTBIService = \frac{\text{production time}}{\text{number of service interrupts}}$$

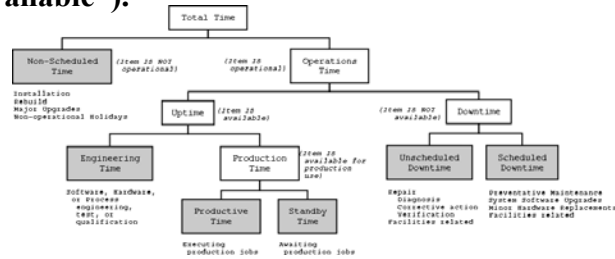
$$MNBI_{Service} = \frac{\text{productive nodehours}}{\text{number of service interrupts}} \quad \leftarrow \text{Includes workload information}$$

Updated paper at <http://www.cs.sandia.gov/~jrstea/ras/defs.pdf>



# Availability

**Availability - the fraction of a time period that an item is in a condition to perform its intended function upon demand (“available”).**



$$\text{Total Availability}_{System} (\%) = \frac{\text{uptime}}{\text{total time}} * 100$$

$$\text{Scheduled Availability}_{System} (\%) = \frac{\text{uptime} - \text{downtime}}{\text{scheduled uptime}} * 100 \quad \leftarrow \text{Quantitative expectations exist}$$



## Next Steps

---

- Ongoing discussion and revision...
- Application to specific platforms (e.g. Red Storm at CUG05, e.g. failure categorization)
- Application to specific applications
- Review failure and interrupt distributions (perhaps towards selecting a model for reliability calculation)

Updated paper at <http://www.cs.sandia.gov/~jrstear/ras/defs.pdf>



## Extra Slides...

---

Updated paper at <http://www.cs.sandia.gov/~jrstear/ras/defs.pdf>



## Serviceability

---

### **Serviceability -**

The probability that an item will be retained in, or restored to, *a condition to perform its intended function* within a specified period of time.

(A.K.A. “maintainability” in other communities)

**Higher serviceability reduces the time spent in repair and maintenance (thus increasing availability and uptime ratio respectively)**

Updated paper at <http://www.cs.sandia.gov/~jrstear/ras/defs.pdf>



## Repair vs Maintain?

---

**Repair** – the act of restoring an item to a condition to perform a required function

**Maintainance** – the act of sustaining an item in or restoring it to a condition to perform a required function, usually during scheduled downtime.

**MTTR** – mean time to repair

**MNTR** - mean nodehours to repair (lost work potential)

**MTTB** - mean time to boot system

**MTTR affects availability**  
(**MTBI** and **MTBF** affect availability and reliability)

Updated paper at <http://www.cs.sandia.gov/~jrstear/ras/defs.pdf>



## Exponential Random Variable

---

Components which exhibit a constant failure rate are appropriately modeled as exponential random variables, which have a time-to-failure pdf of  $f(t)=\lambda e^{-\lambda t}$  (and thus a cdf of  $F(t)=1-e^{-\lambda t}$ ), where  $\lambda$  is the constant "failure rate". Using this model:

$$\text{Reliability} = R(t)=e^{-\lambda t} \quad \text{and} \quad \text{MTBF} = 1/\lambda$$

A system MTBI of 50 hours ( $\lambda=0.02$ ) would correspond to a reliability of 0.368. In other words, there is a 36.8% chance that the system will not experience an interrupt within 50 hours.

If we model Red Storm as a series system of 10,000 nodes which fail independently of each other, and we require a system MTBI of 50 hours, this corresponds to a per-node MTBI of 500,000 hours. If the requirement is reduced to a job running on 40% of the nodes, this corresponds to a node MTBI of 200,000.