SAND2008-2914P

Supercomputer Resilience Research at Sandia

Invited Talk at Oak Ridge National Laboratories
May 8, 2008

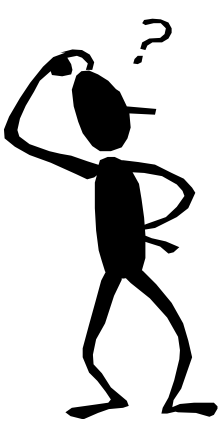
Jon Stearley
jrstear@sandia.gov
Scalable Systems Architecture (1422)

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company,
for the United States Department of Energy's National Nuclear Security Administration
under contract DE-AC04-94AL85000.

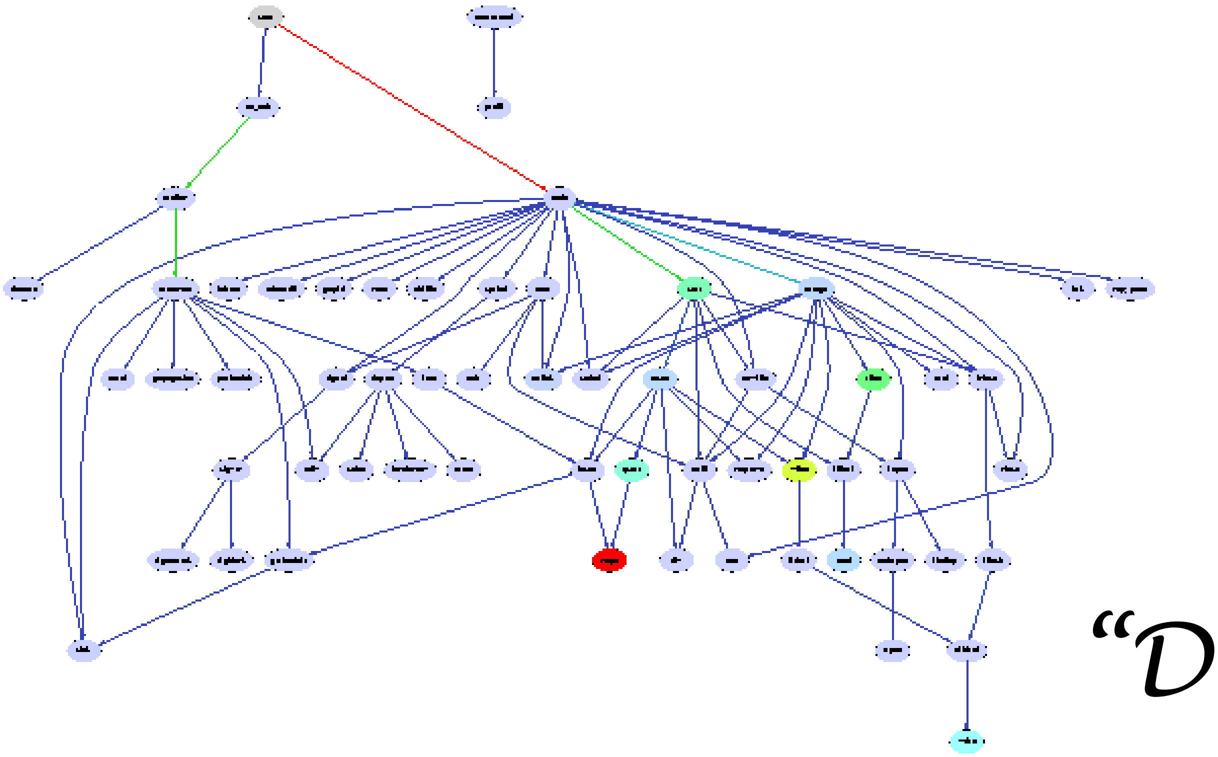
RAS Metrics: Status Quo

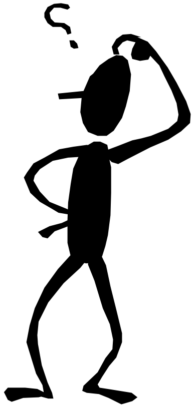
"A computer is in one of two situations. It is either known to be bad or it is in an unknown state."

Mike Levine (PSC)



"Up!"





"Down!"

RAS Metrics: Need and Challenges

Everyone uses the same terms (eg MTBF)
but different definitions and measurements.

- BAD PRACTICE!!! (eg procurements and operations)
- BAD SCIENCE!!! (eg quantify algorithm performance)

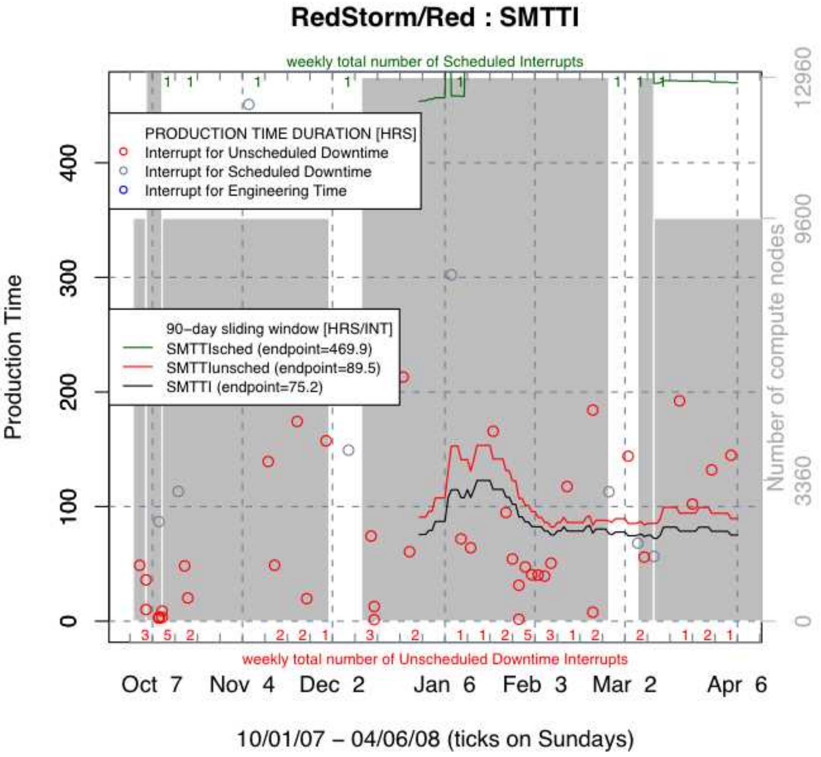
Challenges:

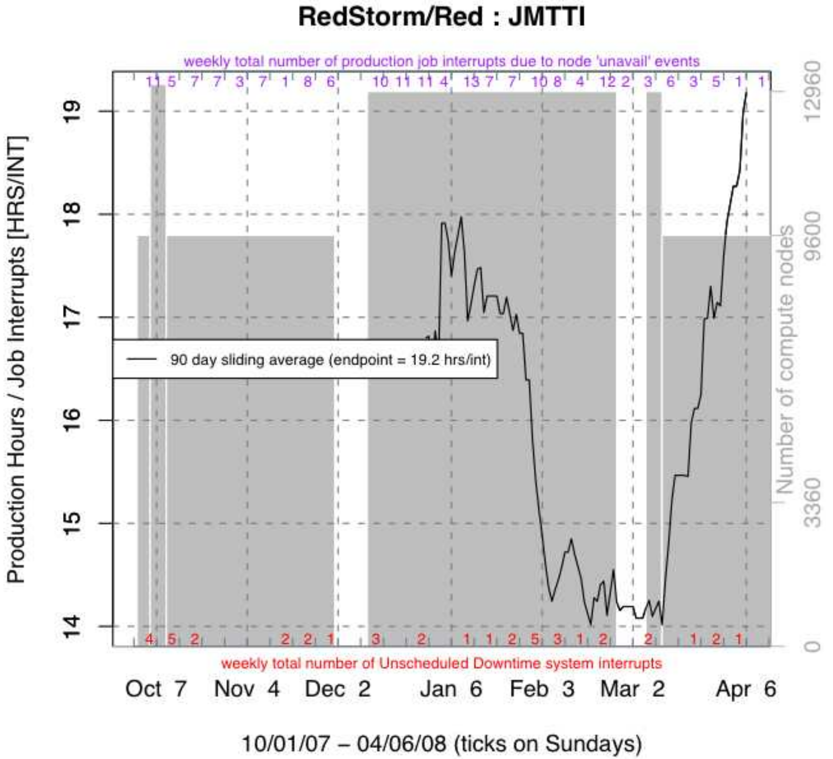
1. Agree on definitions and measurements
   eg: from sysadmin, user, or manager perspective?
2. Alter our spoken and written language.
3. Alter the necessary operational processes and procedures.

RAS Metrics: Idea

HPC Resilience Consortium. **Definition and Measurement of High Performance Computing Reliability, Availability, and Serviceability (RAS).** *Supercomputing 2009.*

And a reference implementation, eg:





OPERATIONS STATUS:

QuickTime™ and a
TIFF (LZW) decompressor
are needed to see this picture.

Contacts: Stearley (SNL), Daly (LANL), Hamilton/Cupps (LLNL)

Logs

Goal:

Given system logs, automatically detect faults.

Approach:

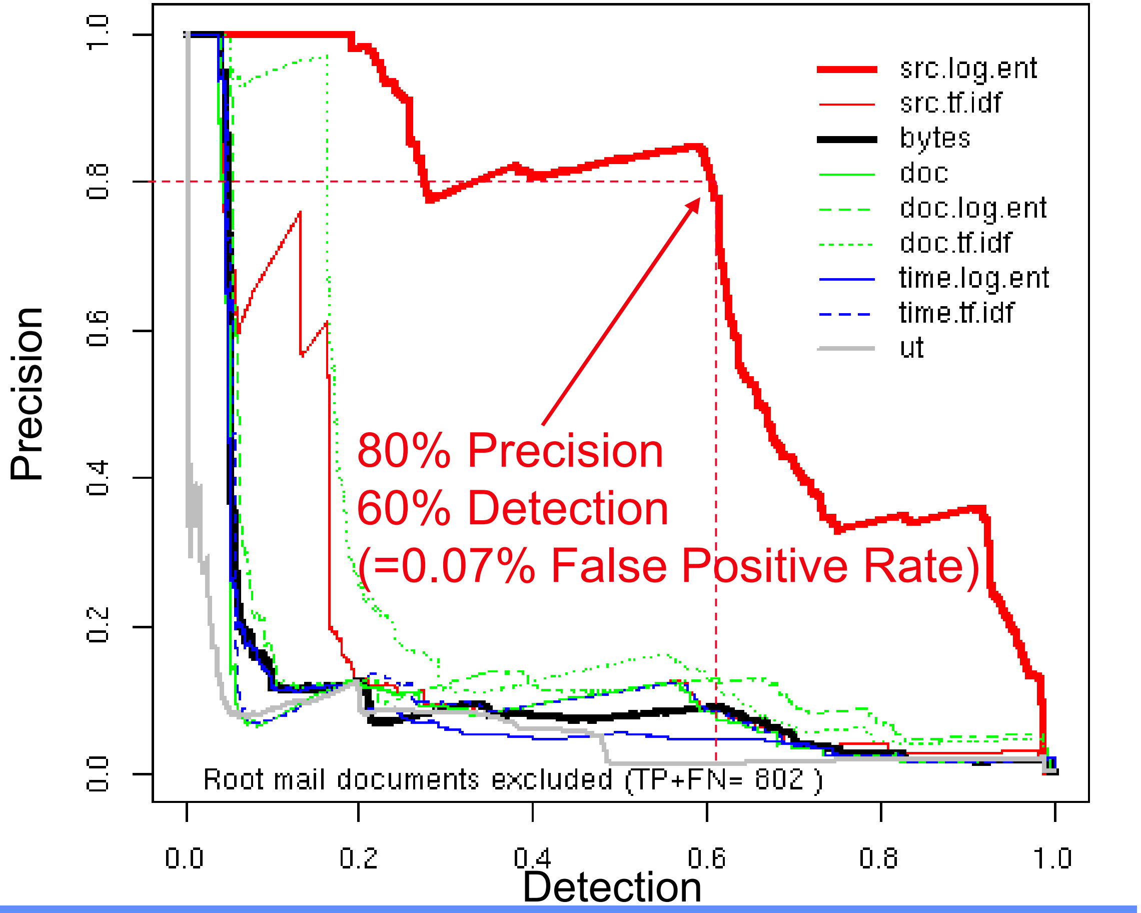
Similar computers correctly executing similar work should produce similar logs
(anomalies often indicate faults).

Quantify:

Measure detection effectiveness based on known fault records.

Measure Detection Effectiveness

33 Unsupervised Classifiers Tested!



80% Precision
60% Detection
(=0.07% False Positive Rate)

Root mail documents excluded (TP+FN= 802)

<u>NWCC/Spirit Data</u>
512 Nodes, 23 Days
8.3M log messages
36K terms, 243K docs
3.9K emails!
P=62 ; 802 N=243K

True Class:

Alarm Class:	*P*	*N*
P	TP	FP
N	FN	TN

TP=True Positives
FP=False Positives
FN=False Negatives
TN=True Negatives

Metrics:
Alarm Precision = TP/(TP+FP)
Event Detection = TP/(TP+FN)

System Logs

Are: Ubiquitous! Informational! Vast!

How do you find the few lines of key information among thousands of log files and millions of lines of time-stamped text???

QuickTime™ and a
TIFF (LZW) decompressor
are needed to see this picture.

Finding Needles in a Craystack

1. Which files contain useful information?
2. Which words convey useful information?
3. Any patterns?

To be Useful,

It Must be Understandable

(to the sysadmins)

1. Which files contain useful information?

files

words

$|(GL)_j|$

G

time

L

abnormal

“interestingness”

normal

QuickTime
TIFF (LZW) de
are needed to see

“interestingness”
(aka “information”) is purely
mathematical (=$|(GL)_j|$).

$G_{i,j}=1+H_i$, $L=\log_2(tf_{i,j})$

$H_i=\sum_j p_{ij}\log_2(p_{ij})/\log_2(d)$

where $p_{ij}= tf_{i,j} /\sum_j tf_{i,j}$
and $tf_{i,j}$ is how many times the
i’th word occurs in the
j’th file

How do you find the few lines of key information among thousands of log files and millions of lines of time-stamped text???

QuickTime™ and a
TIFF (LZW) decompressor
are needed to see this picture.

2. Which words convey useful information?

A gold mine!!!

QuickTime™ and a
TIFF (LZW) decompressor
are needed to see this picture.

2. Which words convey useful information?

And this file…

QuickTime™ and a
TIFF (LZW) decompressor
are needed to see this picture.

2. Which words convey useful information?

Has nuggets!

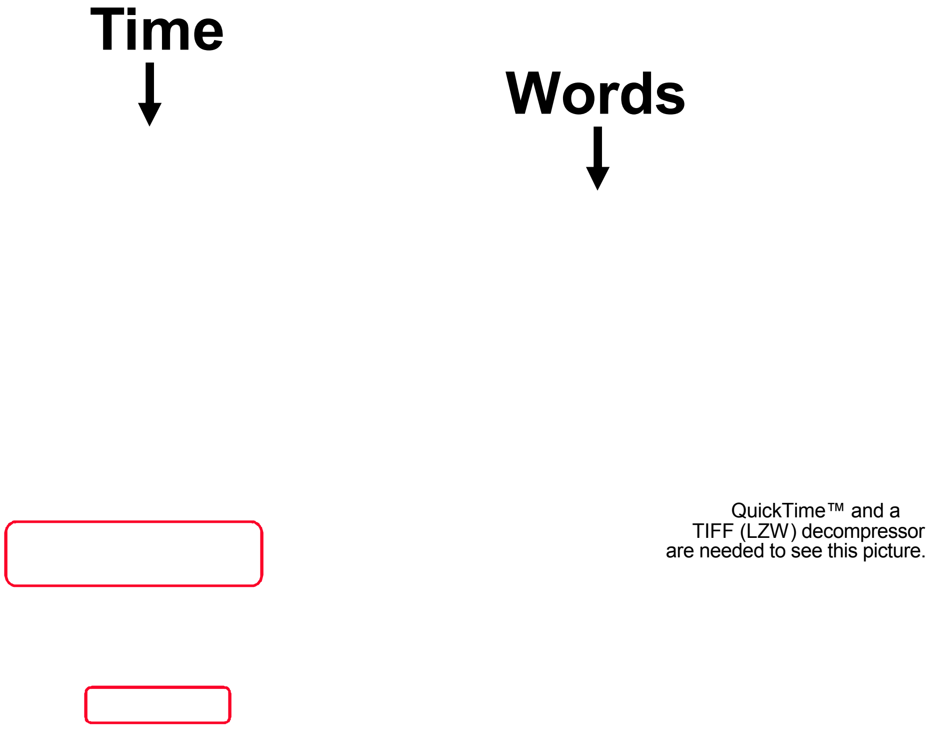
QuickTime™ and a
TIFF (LZW) decompressor
are needed to see this picture.

on 1 computer
(out of 90)

over 4 hours
(out of 4 months)

2. Which words convey useful information?

Find Patterns



Active collaborations:

- **Correlated anomalies** (time and/or space): Oliner at Stanford
- **Latent Semantic Analysis**: Elliot at LaTech, Dunlavy at SNL
- **Graph Layout** (VxOrd): Martin at SNL
- **Logjamm/Homogenization**: Sery at SNL
- **Word Patterns**: Vaarandi at CCoE (EU)

Impacts

Sisyphus has found:

Malfunctions:

disks, controllers, network interfaces, power supplies, memory

Misuse:

RAID stripe imbalance, inappropriate remote monitoring

Misconfigurations:

BIOS, RAID controller, inconsistent software versions, config typos

Which has enabled focused reactive and proactive responses.

Deployments:

SNL: Red Storm, Thunderbird, Spirit, *TLCC, Corporate IT*

LANL *[monitoring suite]***:** *TLCC, Roadrunner*

Downloads: 450+

See http://www.cs.sandia.gov/sisyphus for more info.

Rich Signatures: Need

We must have fault records in order to measure (and optimize) the performance of fault detection (or prediction) algorithms!

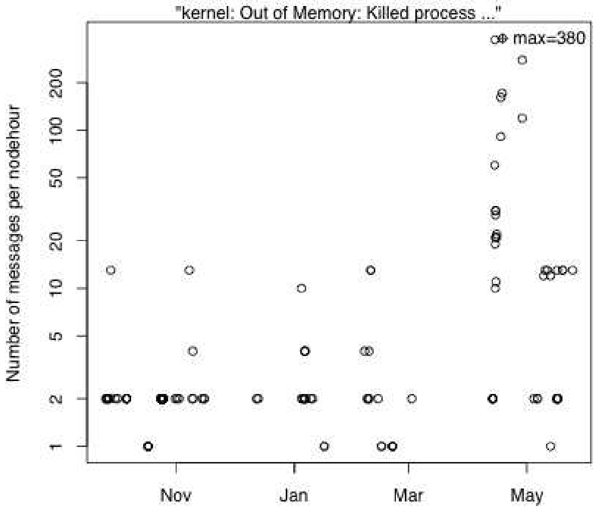
Regular expressions as fault signatures are:

De Facto: the only current way to clearly express fault info

Painful: writing them is error-prone and hard to optimize (specificity, and runtime)

Poor: practically impossible to express context or rate features

We need a painless way to capture fault info from admins and form rich fault signatures!



Rich Signatures: Idea

Latent Semantic Analysis -
mathematically captures content, context, and rate features

Relevance Feedback (RF) - provide a painless way for admins to give:
positive and negative training to researchers (now) and the analysis system (later).

positive: "yes, this log is of interest (eg fault type A)"
negative: "no, this log is not of interest"

Approach:

Compute SVD of term-doc matrix: $\mathbf{X = U\Sigma V^T}$

Rank-reduce to concept-doc matrix: $\mathbf{X_r = U_r\Sigma_r V_r^T}$

Compute doc-doc similarities: $\mathbf{L = X_r^T X_r}$

and concept-concept similarities: $\mathbf{C = X_r X_r^T}$

Use RF, L, and C to detect logs of interest.

SNL LDRD Proposal (‘09-’11)

“System-Directed Resilience for Exascale Platforms”

- **Application Quiescence:** the ability to suspend CPU, network, and storage services used by an individual application without interfering with the progress of other applications.
- **State Management:** the ability to identify, extract, and manage application state in a transparent, efficient, and non-intrusive way.
- **Fault Recovery:** the ability to transparently replace a failed component without restarting the entire application.

PI: Ron Oldfield

Summary

Supercomputer resilience is a rich research area.

SNL momentum and support is increasing
(eg resilience was explicitly prioritized in ‘08 LDRD call).

Standardized definitions and measurements are essential.

Enables good scientific research, engineering, and operation.

Logs are a rich mountain to mine.

They are admins’ primary source of fault info, and
effective automated analyses are greatly needed.