SAND2009-3182C

Combining System Characterization and Novel Execution Models to Achieve Scalable Robust Computing

H. Adalsteinsson, J. Brandt, B. Debusschere, A. Gentile, J. Mayo, P. Pebay, D. Thompson, M. Wong

Sandia National Laboratories

http://ovis.ca.sandia.gov

Sandia
National
Laboratories

Motivation

- Platform growth in size and complexity
  - ~ 100 cores per CPU
  - Heterogeneous CPUs
  - Multiple CPUs per node
    - Hierarchical access to memory, communications, etc.
  - Thousands of nodes
  - Millions of cores
- Expected system wide decrease in mean time to component failure (~constant per component)
- Many current MPI applications are not robust to component failure and don’t take platform non-uniformity into account
  - Hangs if one process fails to reach a barrier (can happen if underlying component failure occurs)
  - Decompositions don’t take resources/hierarchies into account → imbalanc

Big Goal

- Infinitely scalable and 100% efficient applications and platforms

Moderate but attainable Goals

- Understand current and future architectures
  - Failure modes and root causes
  - Associated failure and pre-failure symptoms
    - Detect
    - Predict
  - Monitor and communicate system state and state change (Granularity?)
    - System/Resource manager
    - Application
  - Implications of hierarchical and heterogeneous resources on application performance
    - Propagation of faults

Moderate but attainable Goals Cont.

- Monitoring and analysis infrastructure that can work hand in hand with the system and application to facilitate high performance and resilience to faults

- Novel execution models that are able to take advantage of new and future platform architectures and leverage advanced monitoring and analysis capabilities

Current interrelated Projects aligned with goals

- OVIS: A tool for scalable real time monitoring and analysis
- Quantification of ability to predict failures
- Novel execution models to enable scalable, reliable, and self-balancing applications

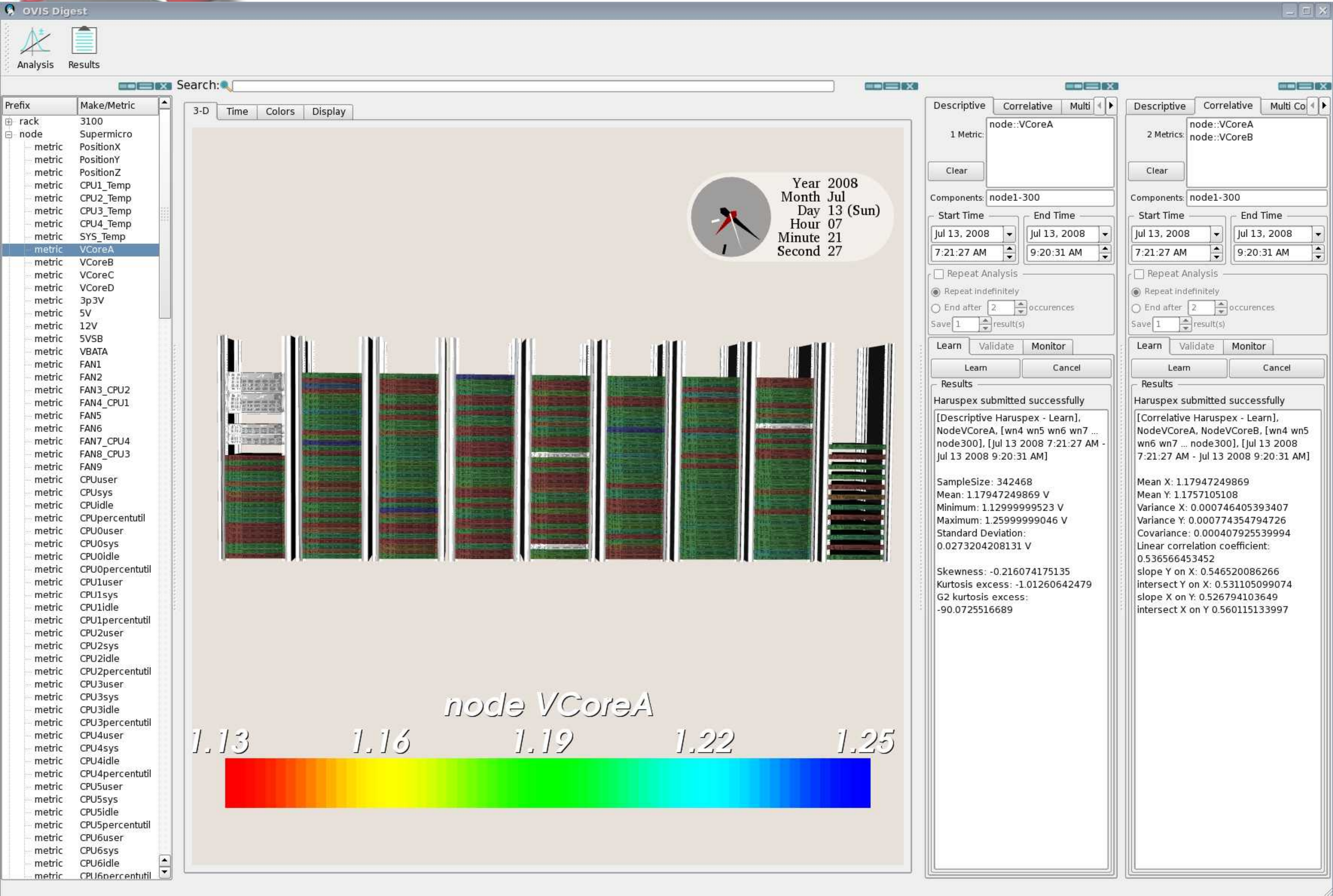
OVIS Goals

- Scalable exploration of measurable attributes via both visual and quantitative analysis
- Run time anomaly detection
  - Hypothesize that statistical outliers can be an indication of impending failure
- Component health evaluation
  - Relative
  - Degradation
- Inference based root cause analysis
- Probability based failure prediction
- Interaction with
  - Resource management
  - Applications
  - System administrators

Features of current release (2.0)

- Visual analysis
  - Realistic rendering of physical system
  - Components colored by relative magnitude of measurements
  - Play live feeds or historic data to see temporal variation
- Analysis engines
  - Descriptive statistics
  - Multi-variate correlative statistics
  - Bayesian inference
- Learn and Monitor modes of operation

OVIS (TLCC)



Prefix	Make/Metric
rack	3100
node	Supermicro
metric	PositionX
metric	PositionY
metric	PositionZ
metric	CPU1_Temp
metric	CPU2_Temp
metric	CPU3_Temp
metric	CPU4_Temp
metric	SYS_Temp
metric	VCoreA
metric	VCoreB
metric	VCoreC
metric	VCoreD
metric	3p3V
metric	5V
metric	12V
metric	5VSB
metric	VBATA
metric	FAN1
metric	FAN2
metric	FAN3_CPU2
metric	FAN4_CPU1
metric	FAN5
metric	FAN6
metric	FAN7_CPU4
metric	FAN8_CPU3
metric	FAN9
metric	CPUuser
metric	CPUsys
metric	CPUidle
metric	CPUpercentutil
metric	CPU0user
metric	CPU0sys
metric	CPU0idle
metric	CPU0percentutil
metric	CPU1user
metric	CPU1sys
metric	CPU1idle
metric	CPU1percentutil
metric	CPU2user
metric	CPU2sys
metric	CPU2idle
metric	CPU2percentutil
metric	CPU3user
metric	CPU3sys
metric	CPU3idle
metric	CPU3percentutil
metric	CPU4user
metric	CPU4sys
metric	CPU4idle
metric	CPU4percentutil
metric	CPU5user
metric	CPU5sys
metric	CPU5idle
metric	CPU5percentutil
metric	CPU6user
metric	CPU6sys
metric	CPU6idle
metric	CPU6percentutil

1 Metric: node::VCoreA

Components: node1-300

Start Time: Jul 13, 2008 7:21:27 AM

End Time: Jul 13, 2008 9:20:31 AM

☐ Repeat Analysis

● Repeat indefinitely

○ End after 2 occurences

Save 1 result(s)

Results

Haruspex submitted successfully

```
[Descriptive Haruspex - Learn],
NodeVCoreA, [wn4 wn5 wn6 wn7 ...
node300], [Jul 13 2008 7:21:27 AM -
Jul 13 2008 9:20:31 AM]

SampleSize: 342468
Mean: 1.17947249869 V
Minimum: 1.12999999523 V
Maximum: 1.25999999046 V
Standard Deviation:
0.0273204208131 V

Skewness: -0.216074175135
Kurtosis excess: -1.01260642479
G2 kurtosis excess:
-90.0725516689
```

2 Metrics: node::VCoreA node::VCoreB

Components: node1-300

Start Time: Jul 13, 2008 7:21:27 AM

End Time: Jul 13, 2008 9:20:31 AM

☐ Repeat Analysis

● Repeat indefinitely

○ End after 2 occurences

Save 1 result(s)

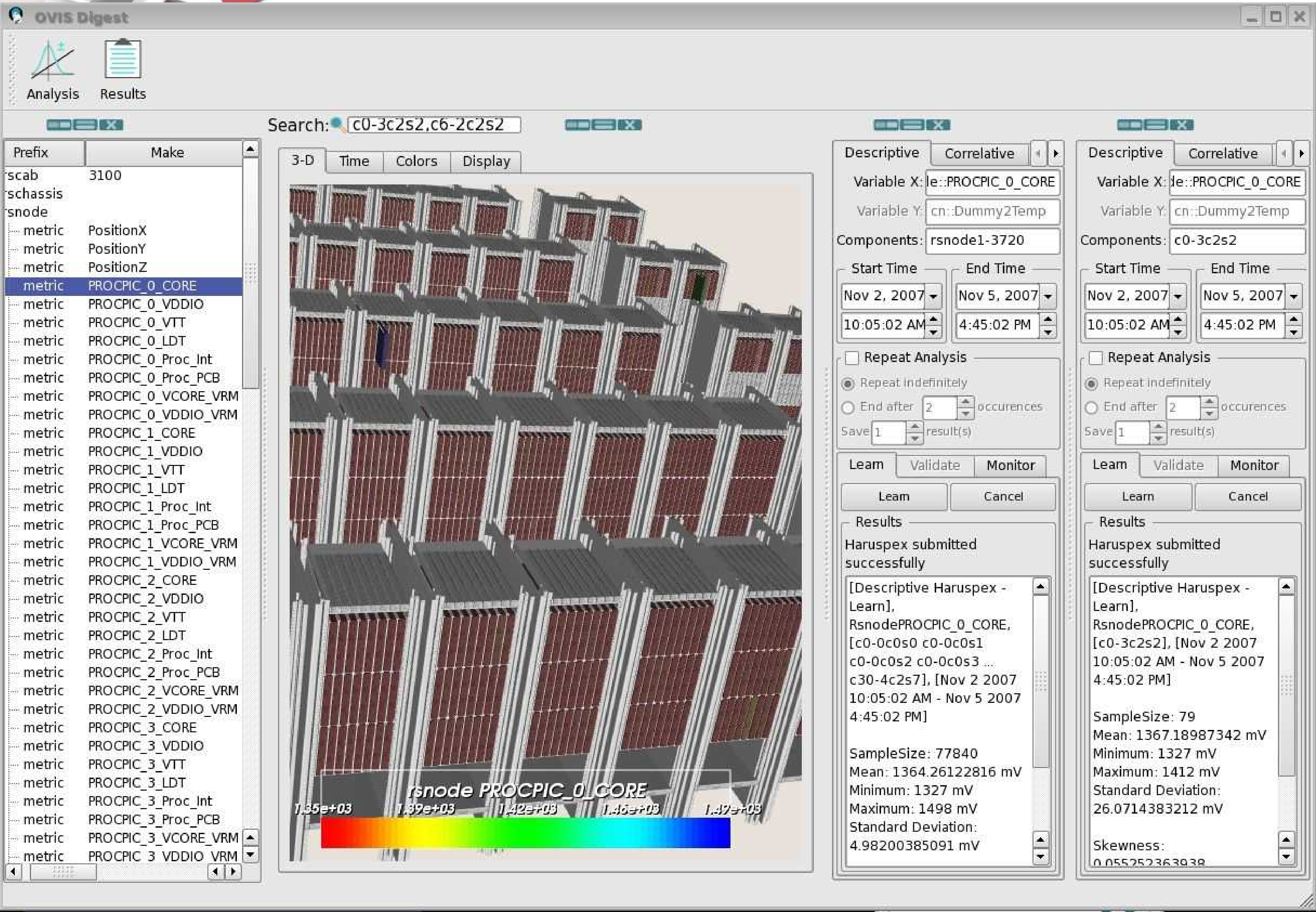
Results

Haruspex submitted successfully

```
[Correlative Haruspex - Learn],
NodeVCoreA, NodeVCoreB, [wn4 wn5
wn6 wn7 ... node300], [Jul 13 2008
7:21:27 AM - Jul 13 2008 9:20:31 AM]

Mean X: 1.17947249869
Mean Y: 1.1757105108
Variance X: 0.000746405393407
Variance Y: 0.000774354794726
Covariance: 0.000407925539994
Linear correlation coefficient:
0.536566453452
slope Y on X: 0.546520086266
intersect Y on X: 0.531105099074
slope X on Y: 0.526794103649
intersect X on Y 0.560115133997
```

OVIS (Red Storm)



Search: c0-3c2s2,c6-2c2s2

Prefix	Make
rscab	3100
rschassis	
rsnode	
metric	PositionX
metric	PositionY
metric	PositionZ
metric	PROCPIC_0_CORE
metric	PROCPIC_0_VDDIO
metric	PROCPIC_0_VTT
metric	PROCPIC_0_LDT
metric	PROCPIC_0_Proc_Int
metric	PROCPIC_0_Proc_PCB
metric	PROCPIC_0_VCORE_VRM
metric	PROCPIC_0_VDDIO_VRM
metric	PROCPIC_1_CORE
metric	PROCPIC_1_VDDIO
metric	PROCPIC_1_VTT
metric	PROCPIC_1_LDT
metric	PROCPIC_1_Proc_Int
metric	PROCPIC_1_Proc_PCB
metric	PROCPIC_1_VCORE_VRM
metric	PROCPIC_1_VDDIO_VRM
metric	PROCPIC_2_CORE
metric	PROCPIC_2_VDDIO
metric	PROCPIC_2_VTT
metric	PROCPIC_2_LDT
metric	PROCPIC_2_Proc_Int
metric	PROCPIC_2_Proc_PCB
metric	PROCPIC_2_VCORE_VRM
metric	PROCPIC_2_VDDIO_VRM
metric	PROCPIC_3_CORE
metric	PROCPIC_3_VDDIO
metric	PROCPIC_3_VTT
metric	PROCPIC_3_LDT
metric	PROCPIC_3_Proc_Int
metric	PROCPIC_3_Proc_PCB
metric	PROCPIC_3_VCORE_VRM
metric	PROCPIC_3_VDDIO_VRM

Variable X: le::PROCPIC_0_CORE
Variable Y: cn::Dummy2Temp
Components: rsnode1-3720
Start Time: Nov 2, 2007 10:05:02 AM
End Time: Nov 5, 2007 4:45:02 PM
□ Repeat Analysis
● Repeat indefinitely
○ End after 2 occurences
Save 1 result(s)

Results

Haruspex submitted successfully

[Descriptive Haruspex - Learn], RsnodePROCPIC_0_CORE, [c0-0c0s0 c0-0c0s1 c0-0c0s2 c0-0c0s3 ... c30-4c2s7], [Nov 2 2007 10:05:02 AM - Nov 5 2007 4:45:02 PM]

SampleSize: 77840
Mean: 1364.26122816 mV
Minimum: 1327 mV
Maximum: 1498 mV
Standard Deviation: 4.98200385091 mV

Variable X: le::PROCPIC_0_CORE
Variable Y: cn::Dummy2Temp
Components: c0-3c2s2
Start Time: Nov 2, 2007 10:05:02 AM
End Time: Nov 5, 2007 4:45:02 PM
□ Repeat Analysis
● Repeat indefinitely
○ End after 2 occurences
Save 1 result(s)

Results

Haruspex submitted successfully

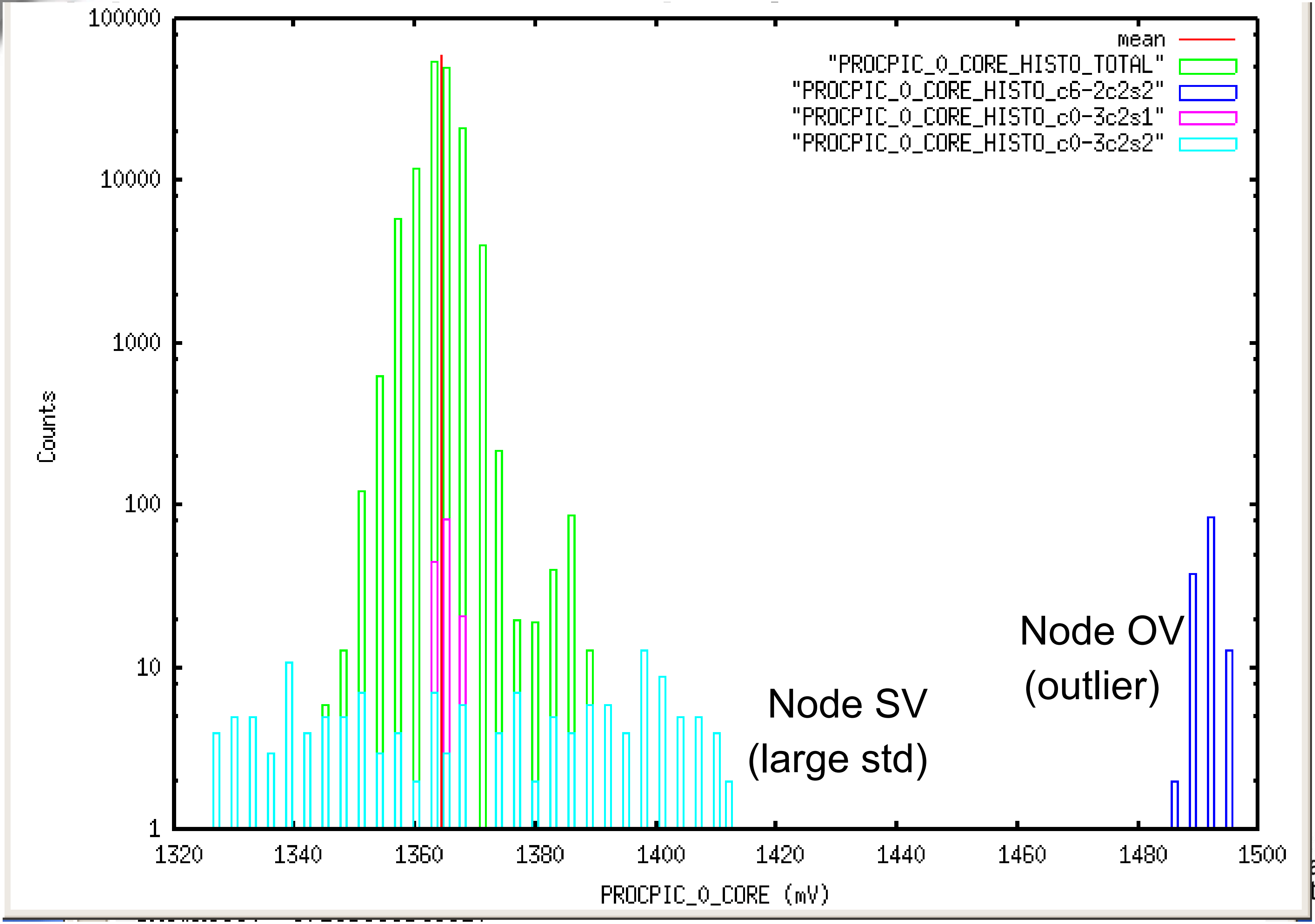
[Descriptive Haruspex - Learn], RsnodePROCPIC_0_CORE, [c0-3c2s2], [Nov 2 2007 10:05:02 AM - Nov 5 2007 4:45:02 PM]

SampleSize: 79
Mean: 1367.18987342 mV
Minimum: 1327 mV
Maximum: 1412 mV
Standard Deviation: 26.0714383212 mV

Skewness: 0.055252363938

Examples of Interesting Component Related Features

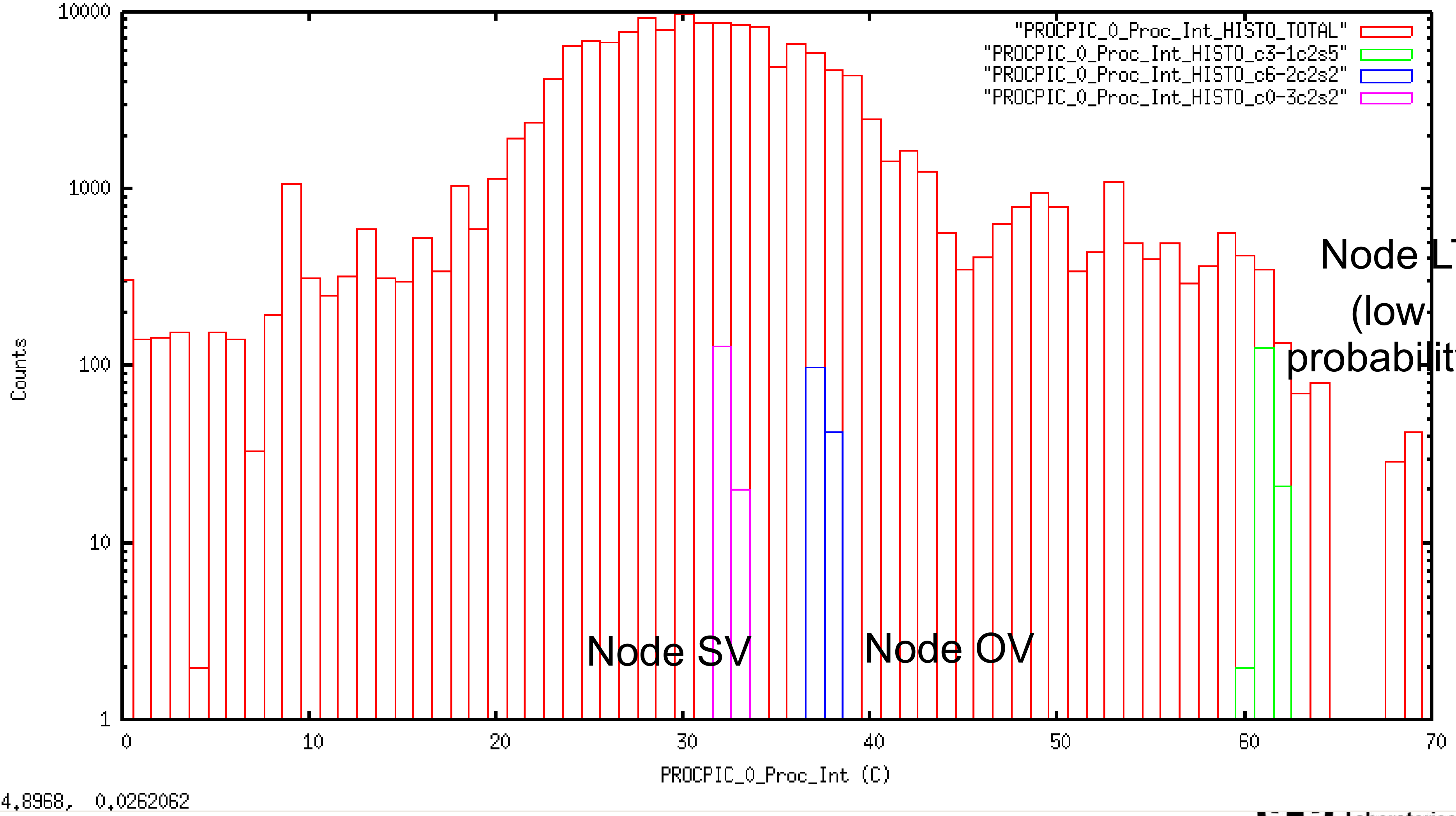
Power supply voltage



Node SV
(large std)

Node OV
(outlier)

Temperature Histogram (log scale)

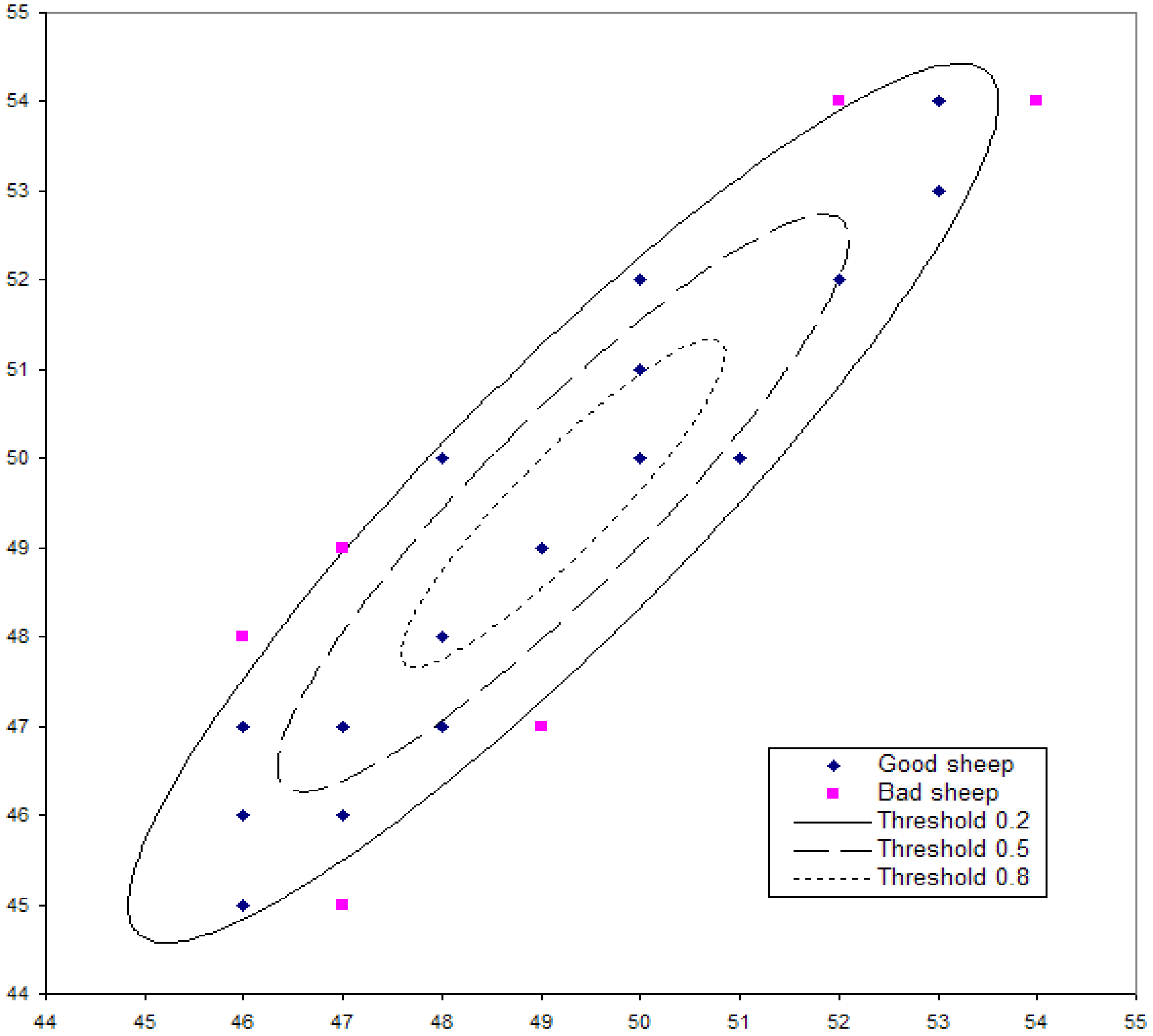


Node LT (low probability)

Node SV

Node OV

Two Variable Temperature Plot with Contours of Relative Probability



Sandia National Laboratories

Challenges

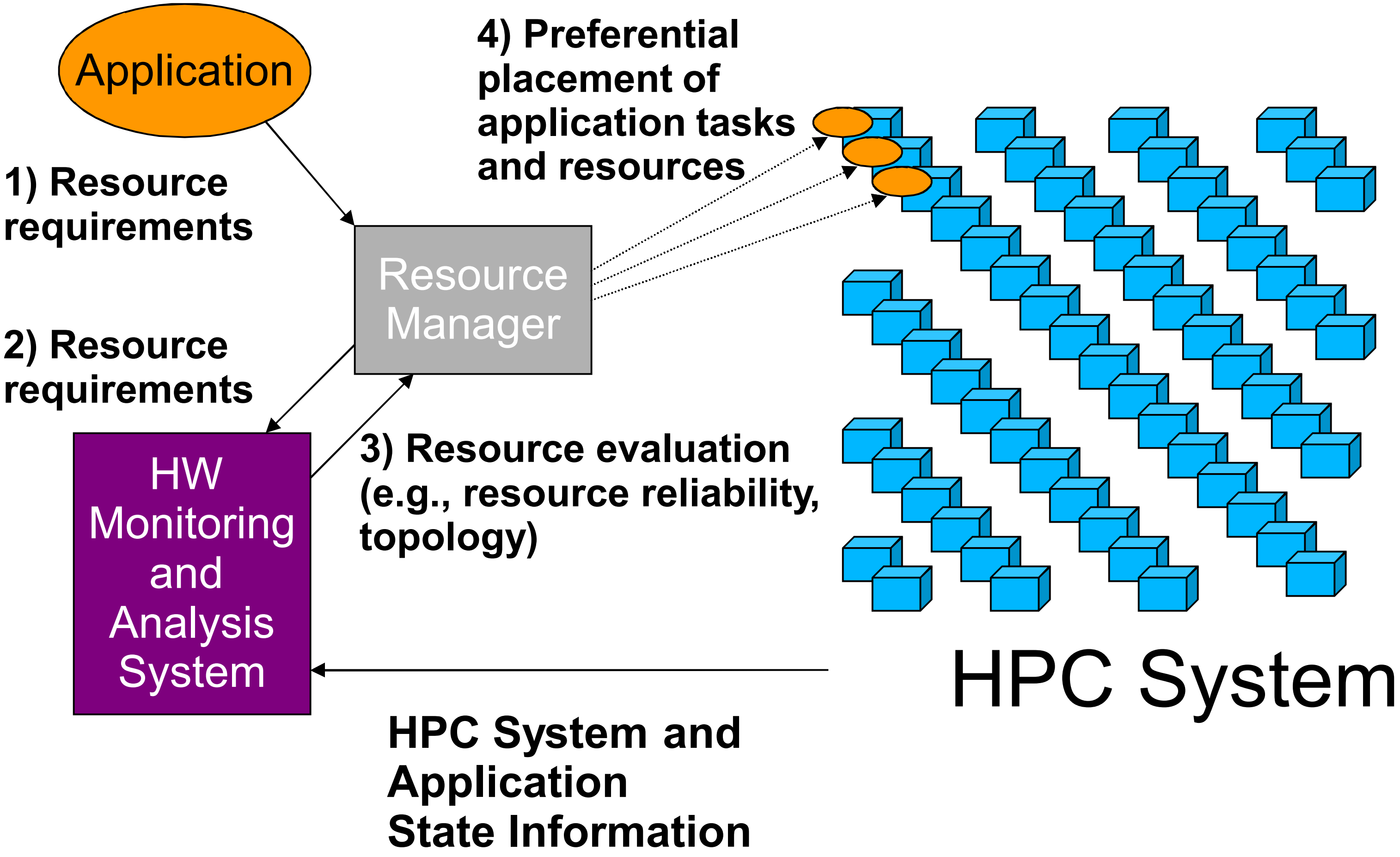
- Long term high frequency (sample every ~seconds) collection of component related data
  - On Whitney ~one million data points per minute with a 5 second sampling interval → ~10GB/day
- Collection of detailed failure data
- Establishing causal relationships between failures and perhaps inter-related behaviors of 10s of variables
- Defining attributes not currently measured that if attainable would significantly contribute to failure prediction
- Quantifying ability to use component characteristic behaviors to predict failures
- Timely feedback to system and application

Quantify Ability to Predict Failures

- Characterization of multivariate distributions
- Time series analysis
- Classifier for identifying events of interest
- Attribute selection
- Root cause analysis via Inference techniques
- Effectiveness scoring algorithms and metrics

Novel execution models to enable scalable, reliable, and self-balancing applications

Preferential Resource Allocation



Application

1) Resource requirements

Resource Manager

2) Resource requirements

HW Monitoring and Analysis System

3) Resource evaluation (e.g., resource reliability, topology)

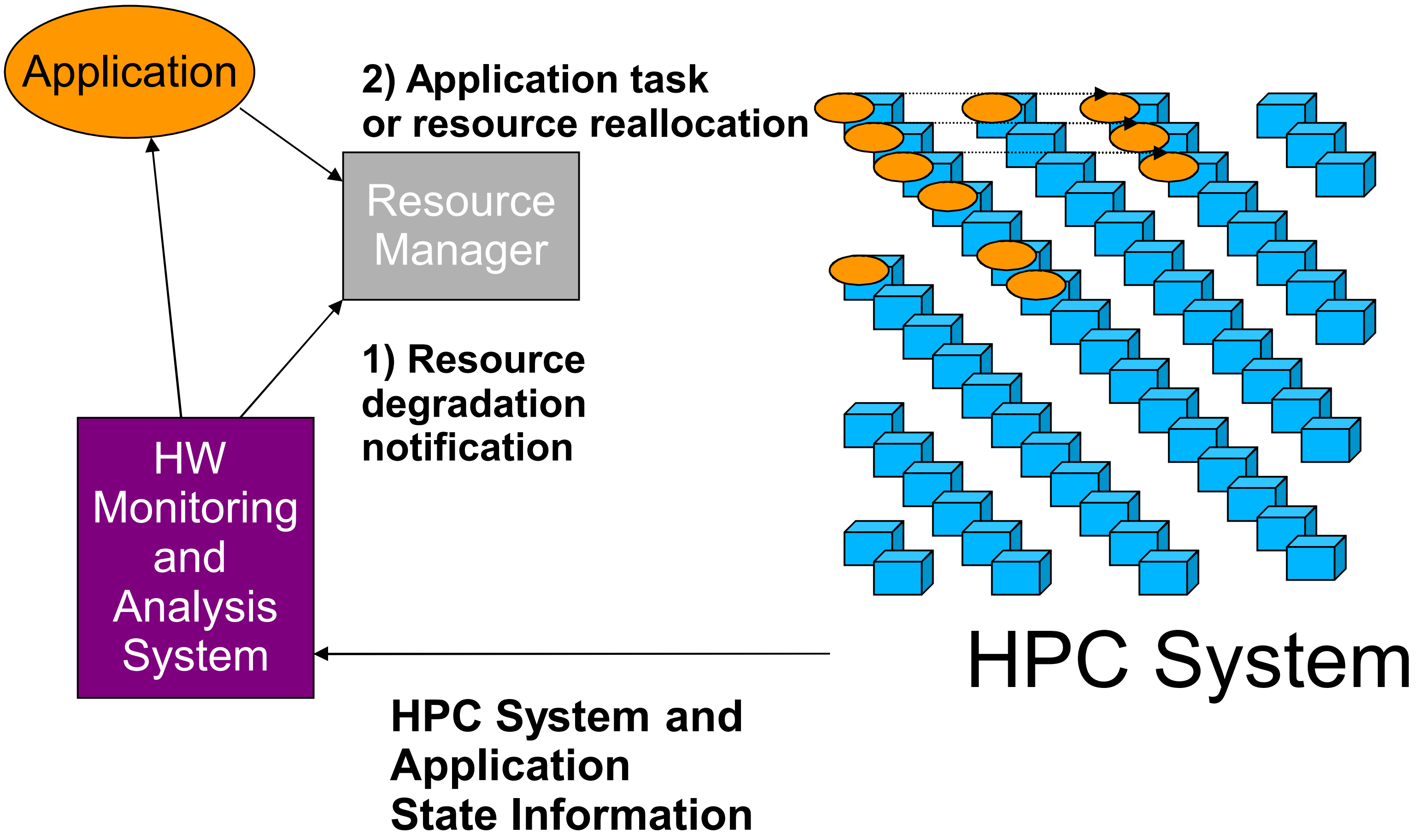
4) Preferential placement of application tasks and resources

HPC System

HPC System and Application State Information

Sandia National Laboratories

Resource Reallocation Upon Failure Prediction



Application

2) Application task or resource reallocation

Resource Manager

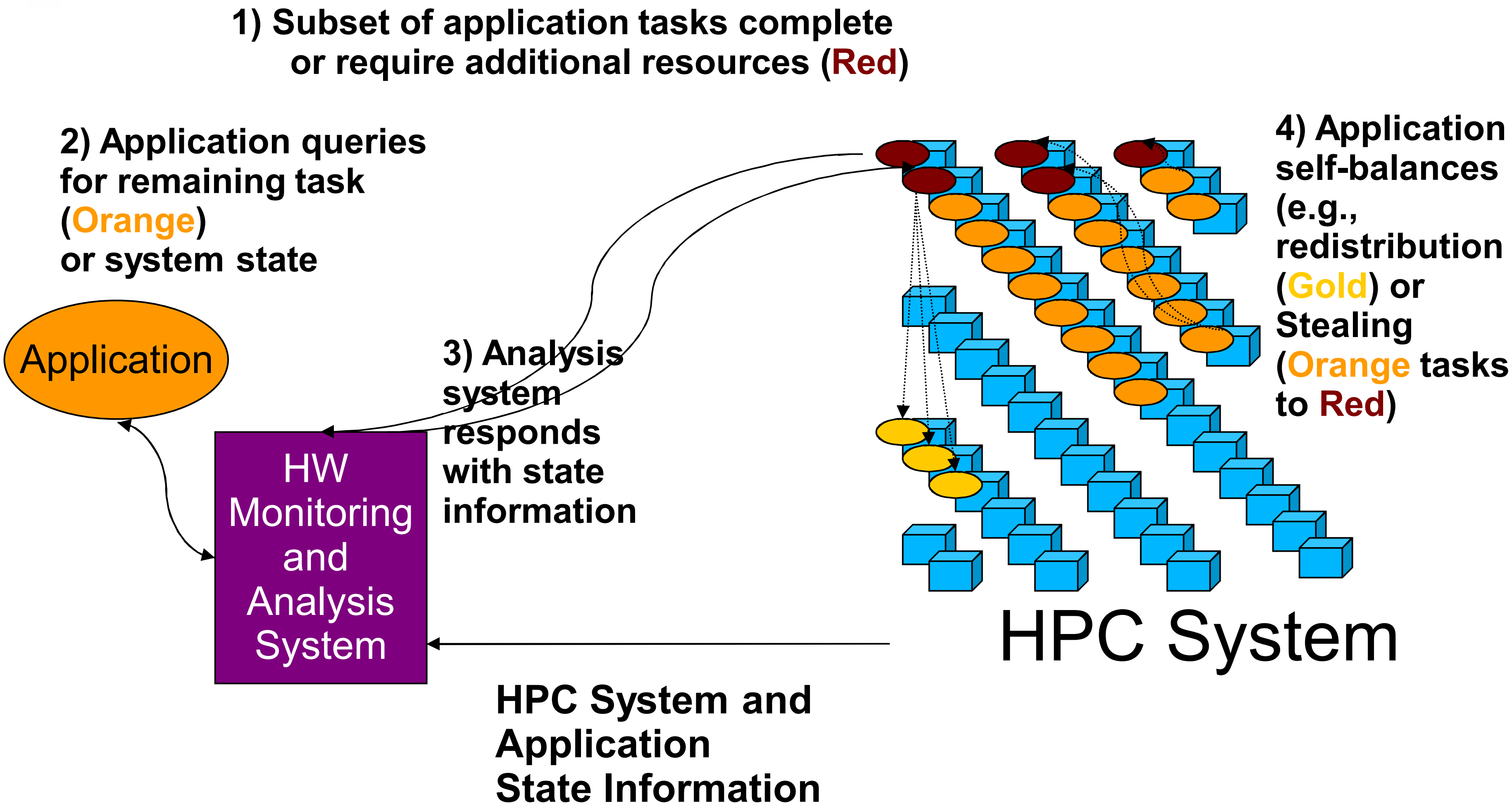
1) Resource degradation notification

HW Monitoring and Analysis System

HPC System

HPC System and Application State Information

Novel Execution Models



1) Subset of application tasks complete or require additional resources (Red)

2) Application queries for remaining task (Orange) or system state

3) Analysis system responds with state information

4) Application self-balances (e.g., redistribution (Gold) or Stealing (Orange tasks to Red)

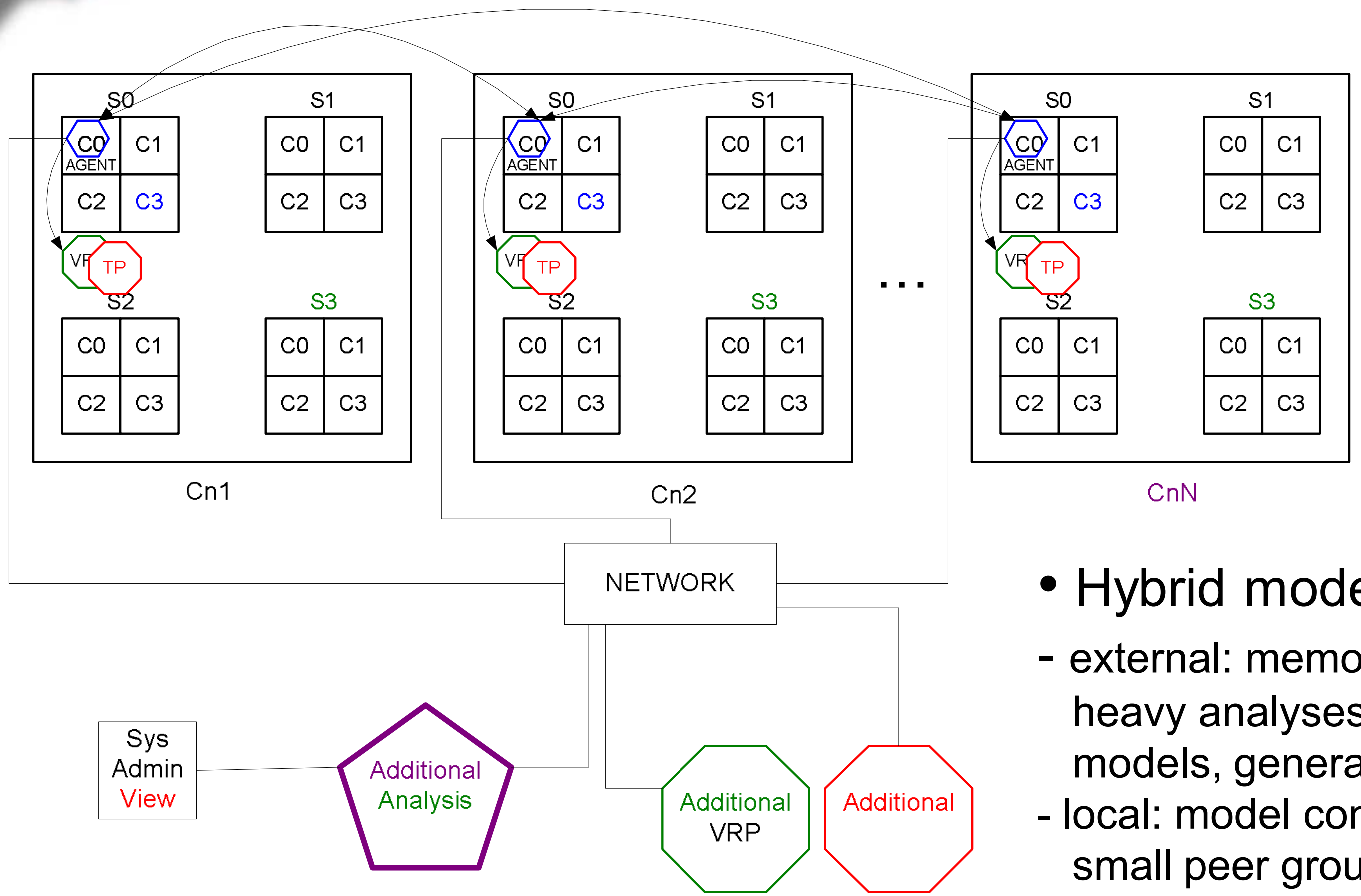
Application

HW Monitoring and Analysis System

HPC System

HPC System and Application State Information

Architecture



- Hybrid model for analysis

- external: memory and state heavy analyses - generate models, generate distributions
- local: model comparisons, small peer group, allow reasonable uncertainty based on limited world view

Challenges

- Determining criteria for task-to-resource mapping
  - Heterogeneous and hierarchical environment
  - Characterization of relevant information
- Availability of resource state information
- Infrastructure for timely analysis + resource manager + application interaction

Future/Ongoing Work

- Quantification of ability to predict failures
  - Hardware numerical data collection
  - Failure modes, symptoms
  - Analysis
  - Missing information identification
- Infrastructure to support up to date resource state knowledge with ability to share that information with both system and application in a useable time frame
  - Small memory footprint agent based
    - Low latency interaction
    - Time vs. global accuracy tradeoff
  - Description of resource state/requirements
    - Total memory, memory bandwidth, compute intensity, etc.
  - Duration of relevancy of historical data

Questions?

http://ovis.ca.sandia.gov

Demo in ASC booth at SC08