Do you know what your automated repair service is doing?

Moises Goldszmidt

Microsoft Research

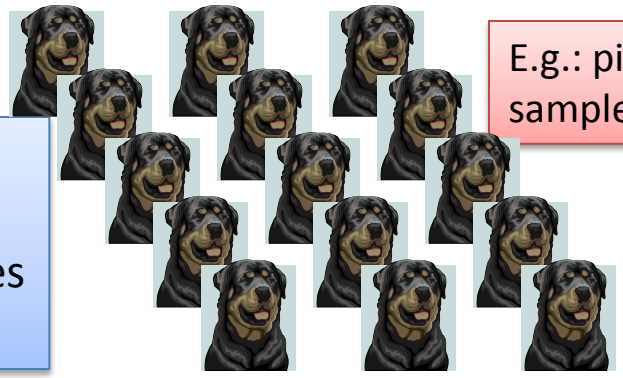
**Joint work with Mihai Budiu and Yue Zhang

The birth of automated repair services

- Need to increase computers/operator ratio
  - Automation gets an economic push
- High availability (QoS) requirements
  - Need to embed tactical/reactive actions in the loop
- “Computers will fail; be prepared” [ROC]
  - Gets designers really thinking about tactical reactive actions (e.g. micro-reboots, check-points)

Autopilot-like repair service

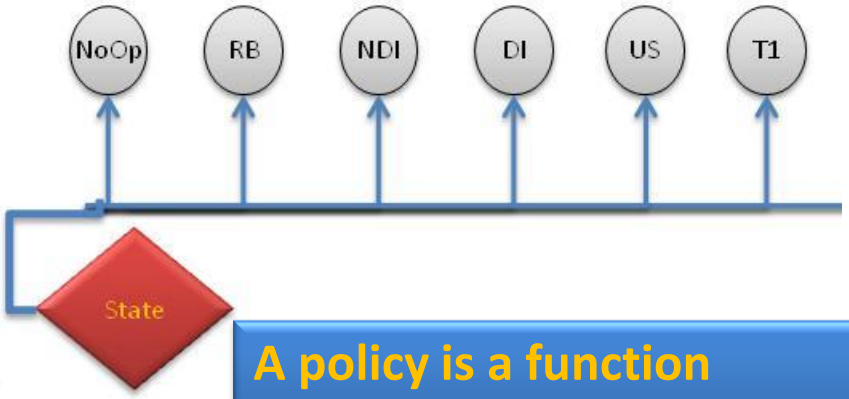
Watchdogs: Asynchronously monitoring machines and sending signals



E.g.: ping, execute transaction, sample cpu, etc.

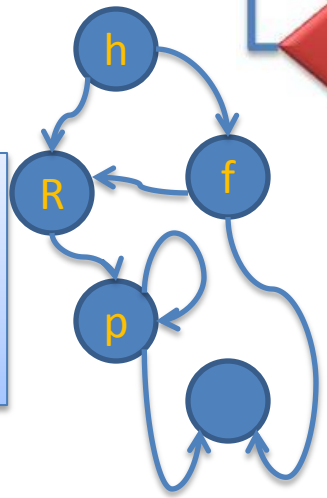
Each machine has a state associated with it

E.g.: healthy, probation, faulty, rebooted_once, etc.



A policy is a function from State to Repair Action

State transitions are regulated by an automaton. A signal or a repair action will cause a state transition



E.g.:
If probation do_nothing.
If rebooted_once reboot.
If dead call tier_1 operator

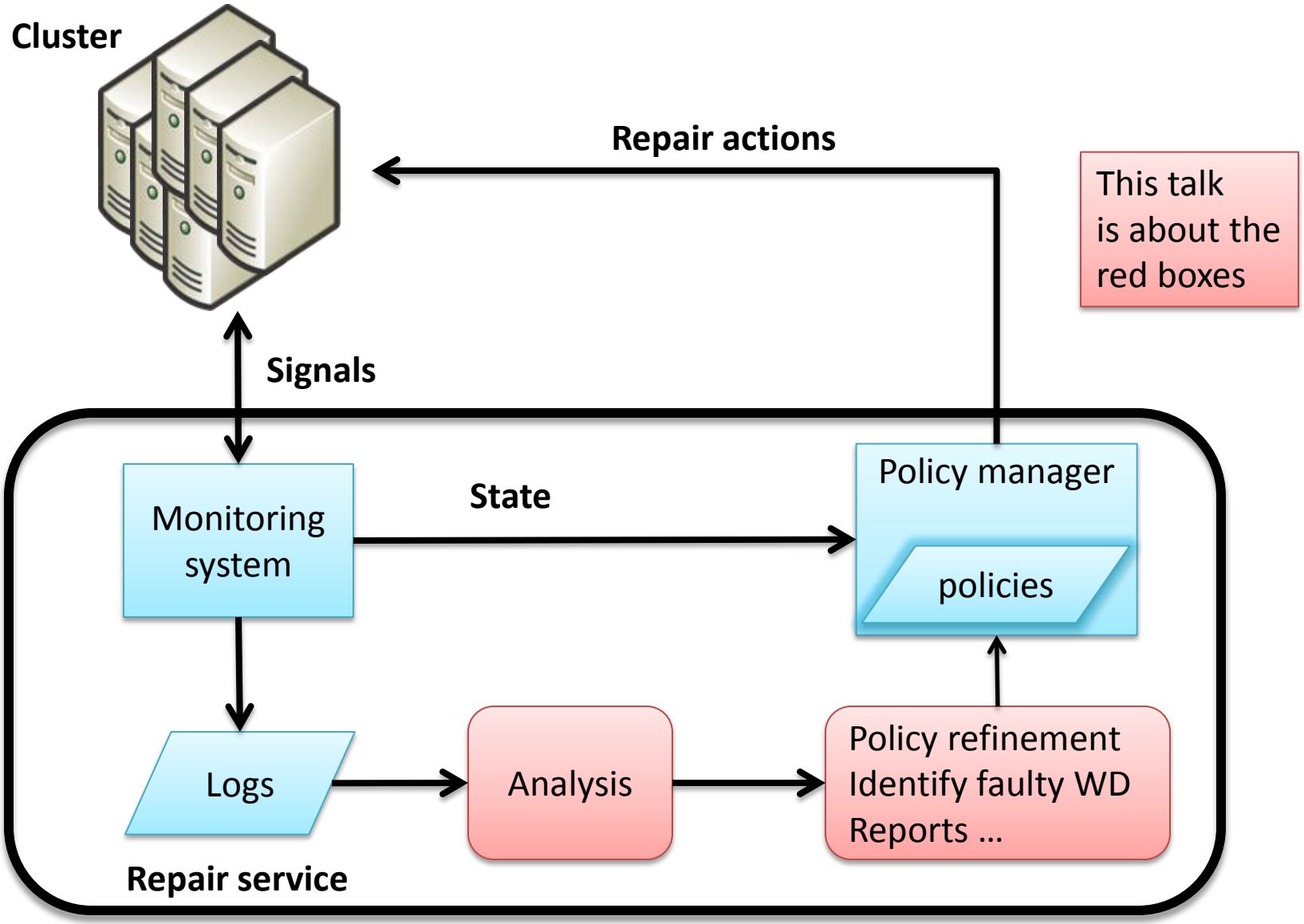
Some remarks…

- Automated diagnosis? Who needs it?
  - Relegated to the “human action”
- Intelligence is in the sensor
  - Watchdogs are designed by the stake holders
- Very rich sources of information in the logs
  - Challenge is how to extract it and put it to use

IT WORKS GREAT! -- for search ☺

- Final criteria → Qos (availability) and $$ numbers
- What about other "properties"? What if the

1) How long does a machine stay alive after a reboot?
2) How much time does a machine spend on the failure state
3) Which watchdogs are reliable?
4) Which are predictive of a failure in a machine being repaired with a reimage?
5) ……

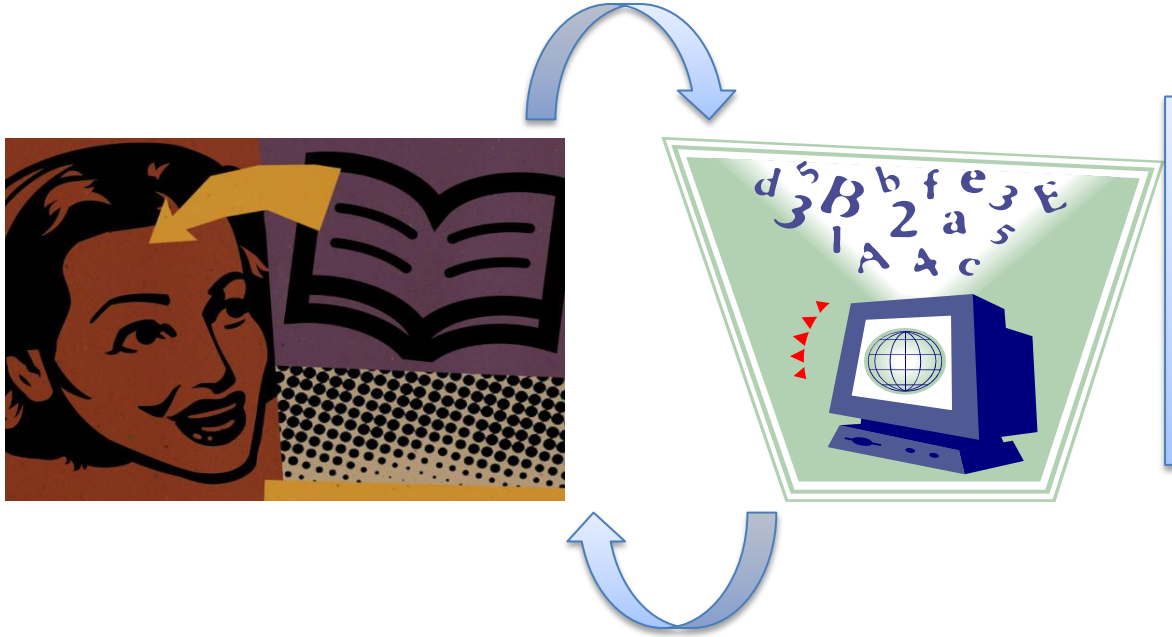

Cluster
Repair actions
This talk
is about the
red boxes
Signals
Monitoring
system
State
Policy manager
policies
Logs
Analysis
Policy refinement
Identify faulty WD
Reports …
Repair service

Looking for trouble with Artemis

with
Mihai Budiu

One stop shop for

- **Data collection**
- **Data transformation**
- **Visualization**
- **Statistical analysis, machine learning, and modeling**

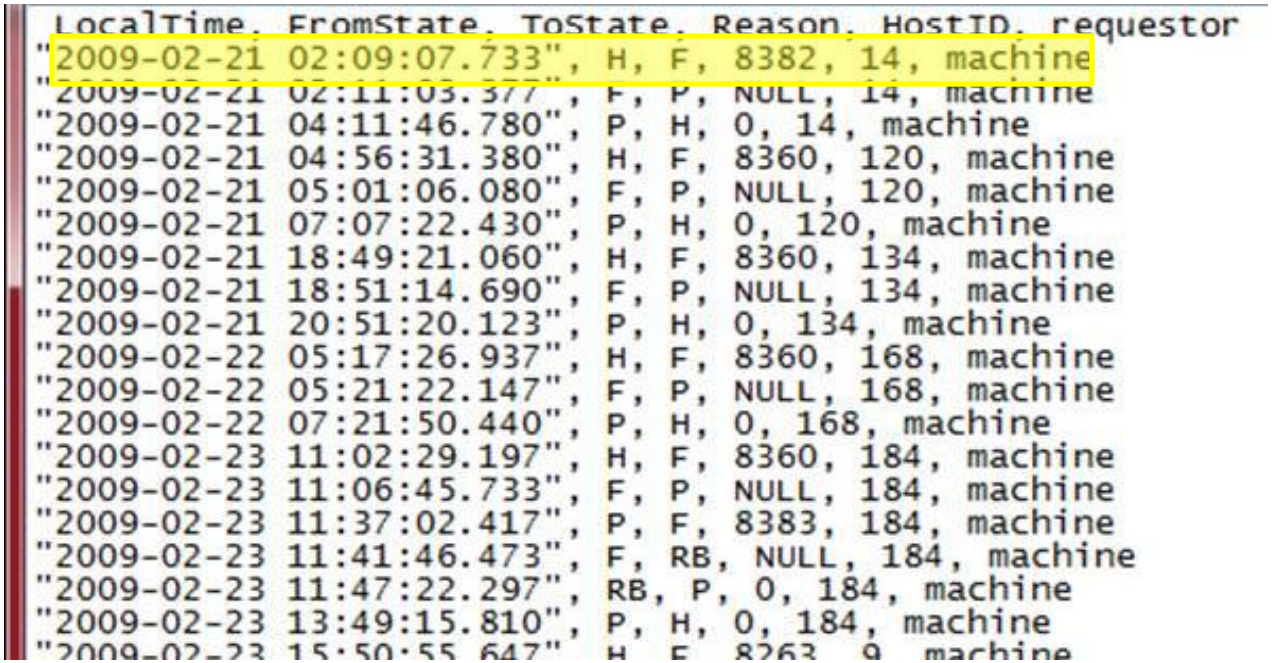


Take advantage of the powerful interaction between the computer's powerful analytics and the human common sense and pattern recognition abilities

Customized Artemis for Windows Live

Logs

Log consisted of
3 months of data
collected from
~ 2k machines



```
LocalTime, FromState, ToState, Reason, HostID, requestor
"2009-02-21 02:09:07.733", H, F, 8382, 14, machine
"2009-02-21 02:11:03.377", F, P, NULL, 14, machine
"2009-02-21 04:11:46.780", P, H, 0, 14, machine
"2009-02-21 04:56:31.380", H, F, 8360, 120, machine
"2009-02-21 05:01:06.080", F, P, NULL, 120, machine
"2009-02-21 07:07:22.430", P, H, 0, 120, machine
"2009-02-21 18:49:21.060", H, F, 8360, 134, machine
"2009-02-21 18:51:14.690", F, P, NULL, 134, machine
"2009-02-21 20:51:20.123", P, H, 0, 134, machine
"2009-02-22 05:17:26.937", H, F, 8360, 168, machine
"2009-02-22 05:21:22.147", F, P, NULL, 168, machine
"2009-02-22 07:21:50.440", P, H, 0, 168, machine
"2009-02-23 11:02:29.197", H, F, 8360, 184, machine
"2009-02-23 11:06:45.733", F, P, NULL, 184, machine
"2009-02-23 11:37:02.417", P, F, 8383, 184, machine
"2009-02-23 11:41:46.473", F, RB, NULL, 184, machine
"2009-02-23 11:47:22.297", RB, P, 0, 184, machine
"2009-02-23 13:49:15.810", P, H, 0, 184, machine
"2009-02-23 15:50:55.647", H, F, 8263, 9, machine
```

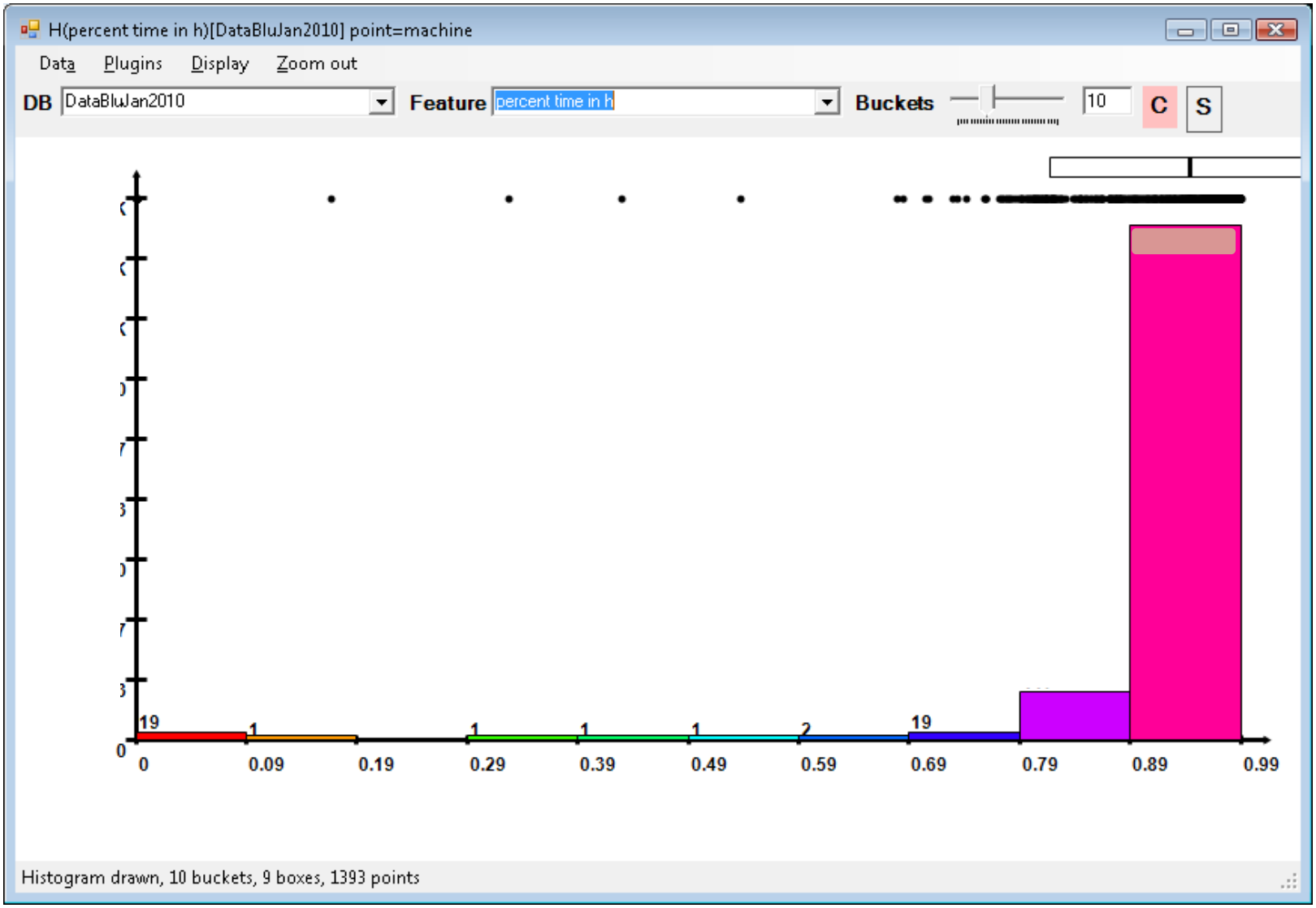
h

Reason for transition
e.g. = e8382

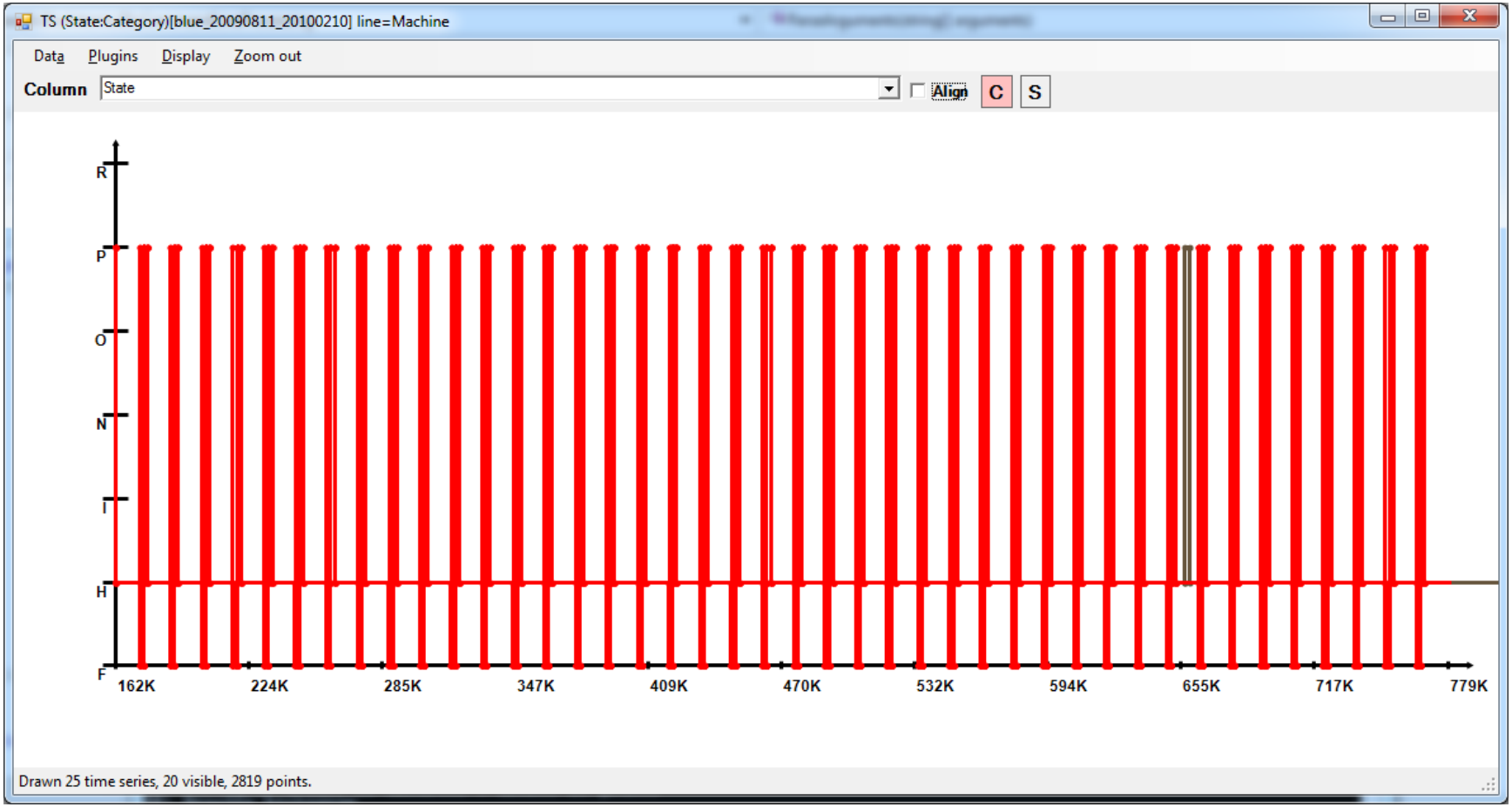
Time of the event
2009-02-21 02:09:07

f

Percentage of time on “Healthy”



Real Data: 20 Machines Oscillate



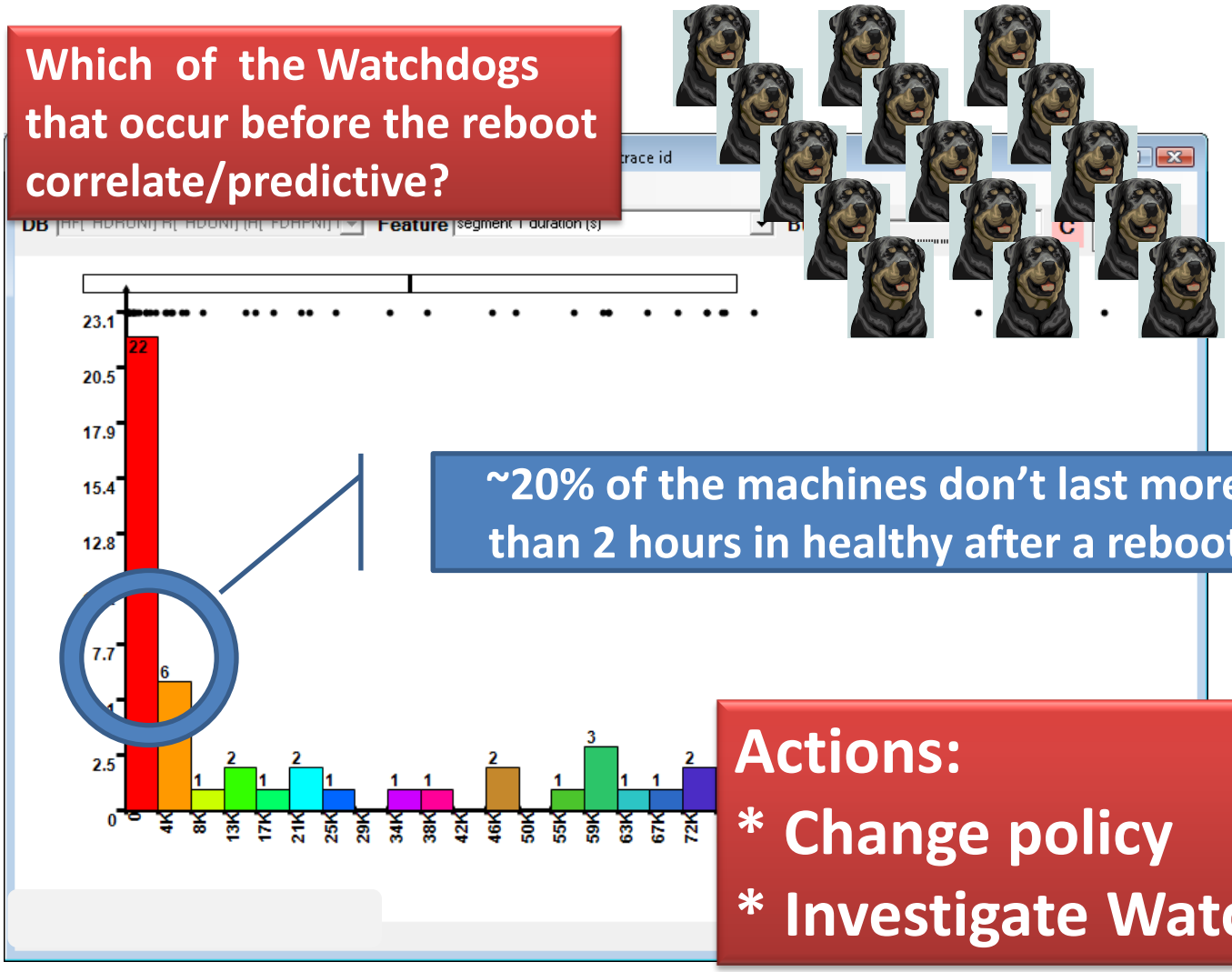
More complex queries….

Refine the policy? Find faulty watchdogs?
- Which watchdogs "predict" a(n) (un)successful action?

How effective are human repairs?
- What is the probability that a machine will be "available" for at least 8 hours after human intervention?

Which watchdogs predict failure?

Which of the Watchdogs that occur before the reboot correlate/predictive?

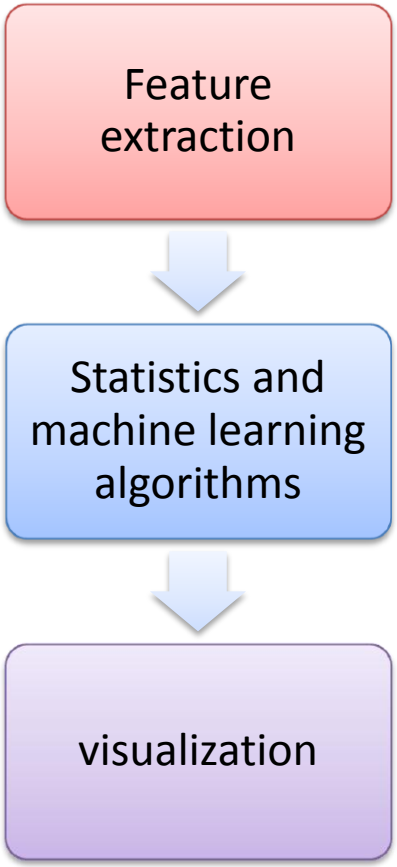


~20% of the machines don't last more than 2 hours in healthy after a reboot

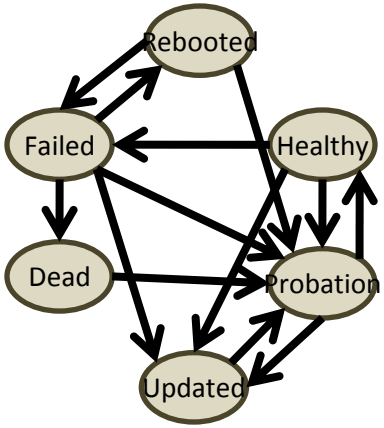
Actions:

*** Change policy**

*** Investigate Watchdogs**


Feature extraction
Statistics and machine learning algorithms
visualization

From traces to features



Step 1 → segment extraction

…RPHFUFRPHFPHPHRPH…

With time stamps….

Query:

How much time does to get to the ***Healthy*** state after a ***Reboot*** ?

1. Extract segments that match -- R[^HR]*H

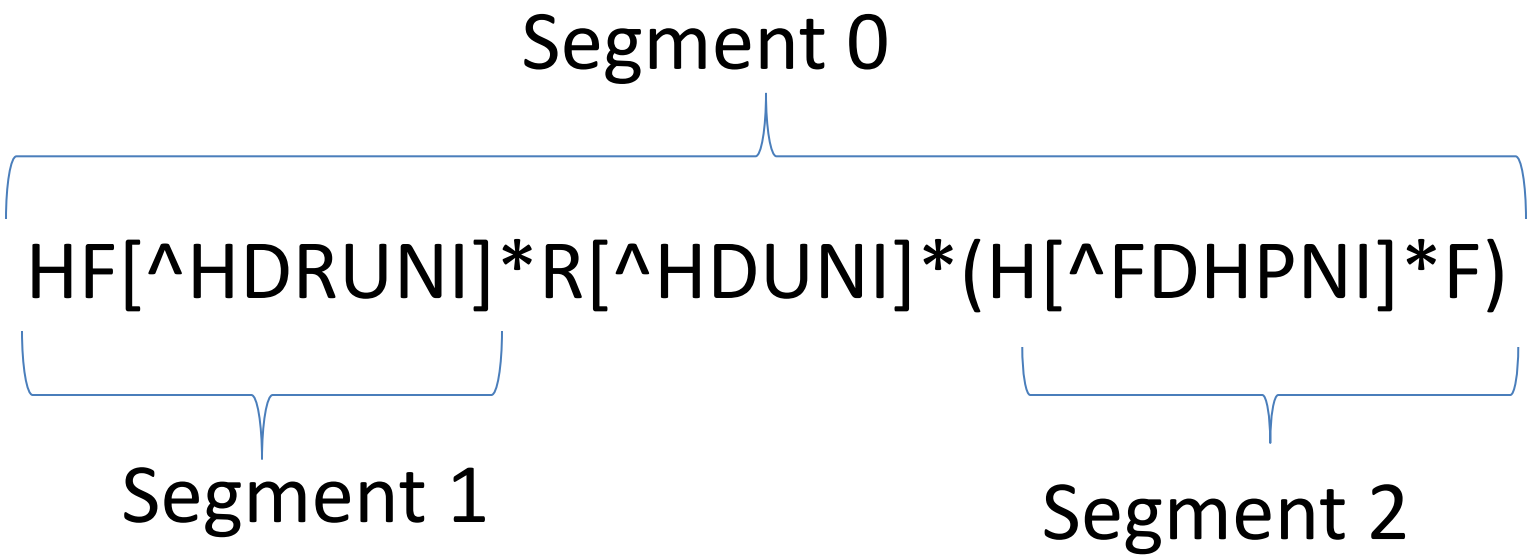
Machine1 …RPHFUFRPHFPHPHRPH…

Machine2 …PHRUPRUPHPHPHPHPRDPH…

2. Compute time differences → (end – beg) of segment

From traces to features

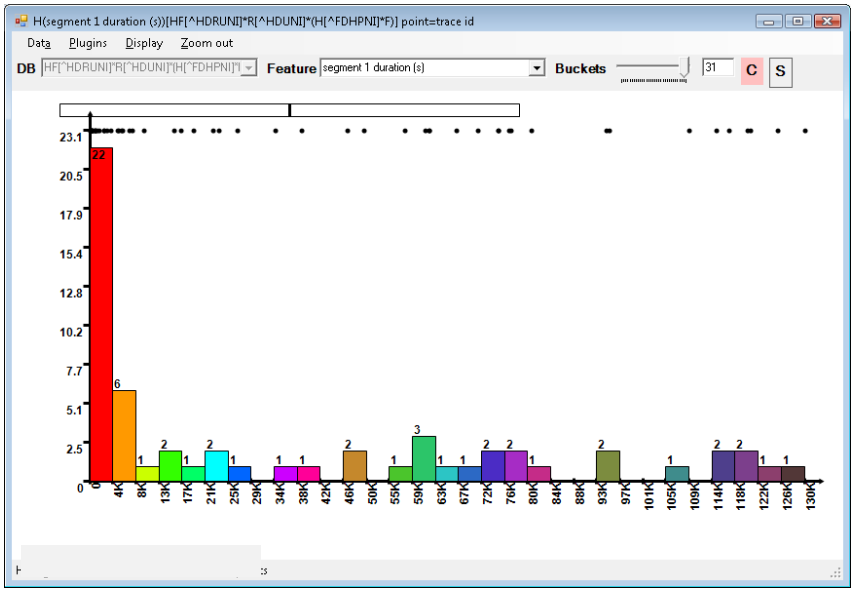
Step 2 → Aggregator



Extract (by segment):
Number of states
Number of times each watchdog sends a signal
Time duration

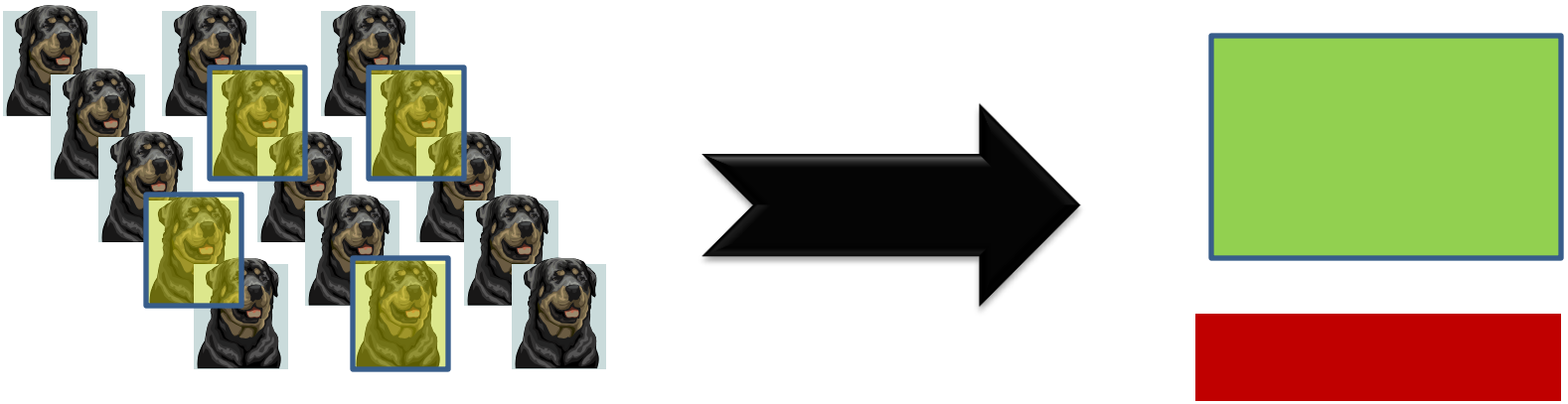
Feature extraction

(HF[^HDRUNI])*R[^HDUNI]*(H[^FDHPNI]*F)



A machine learning approach

Pattern classification: Automatically find a function from watchdogs to the class of machines that last less than two hours in the healthy state



…with feature selection

Logistic Regression with L1

$$\log\left(\frac{P(c^0 \mid wd_1,\ldots,wd_n)}{P(c^1 \mid wd_1,\ldots,wd_n)}\right) = \sum_i \beta_i \times wd_i + I$$

L1 regularization: $\sum_i \beta_i < \lambda$

Advantages of LR+L1

- Shown (empirically) to work well even in cases where #dimensions ~ #samples
- Easy interpretation of the model (linear function)

Model examples

selected signals: 9
CV BA: 0.872
CV confusion matrix:

	below	above
pred below	89	14
pred above	11	71

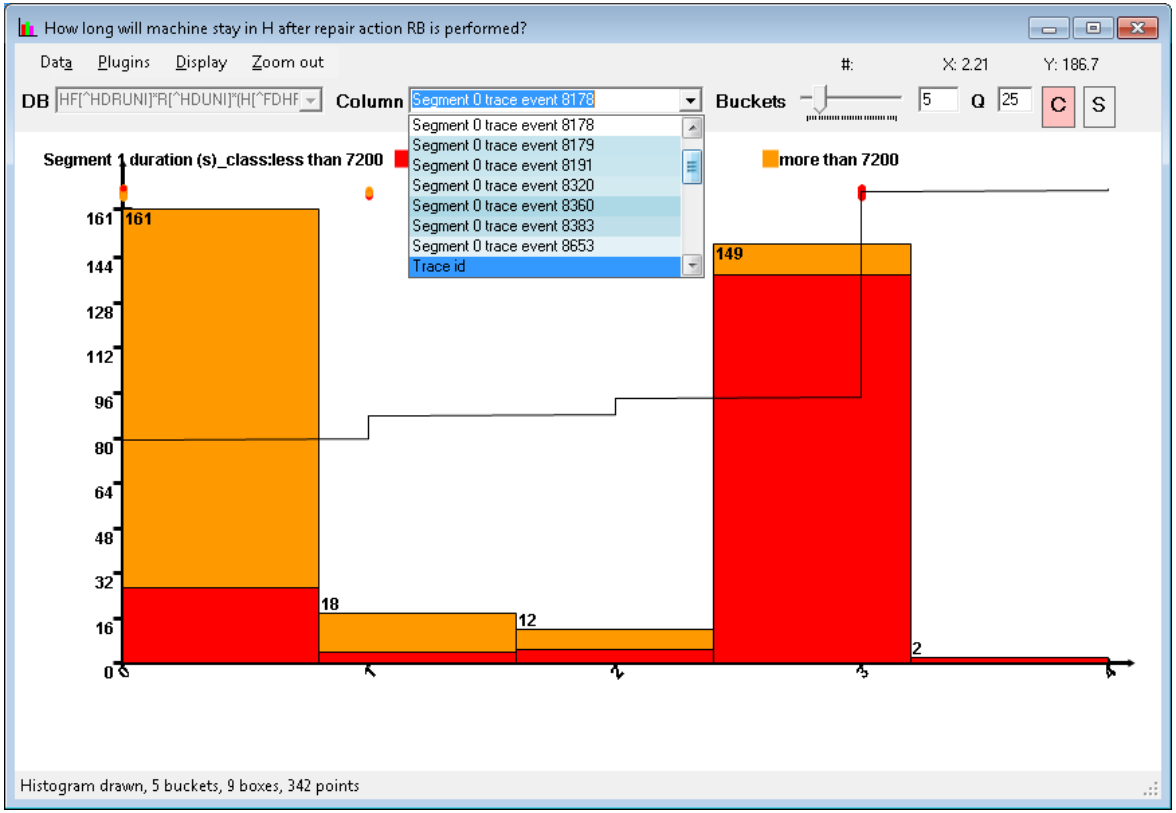
	coeffs	ind	threshold
e50202	-0.79	0.965	0.00
e8240	-0.89	0.942	0.00
e8383	0.31	0.692	1.00
e8506	-0.84	0.861	0.00

185 samples with 42 signals

Visualization

What (events) predict failure to reboot successfully?

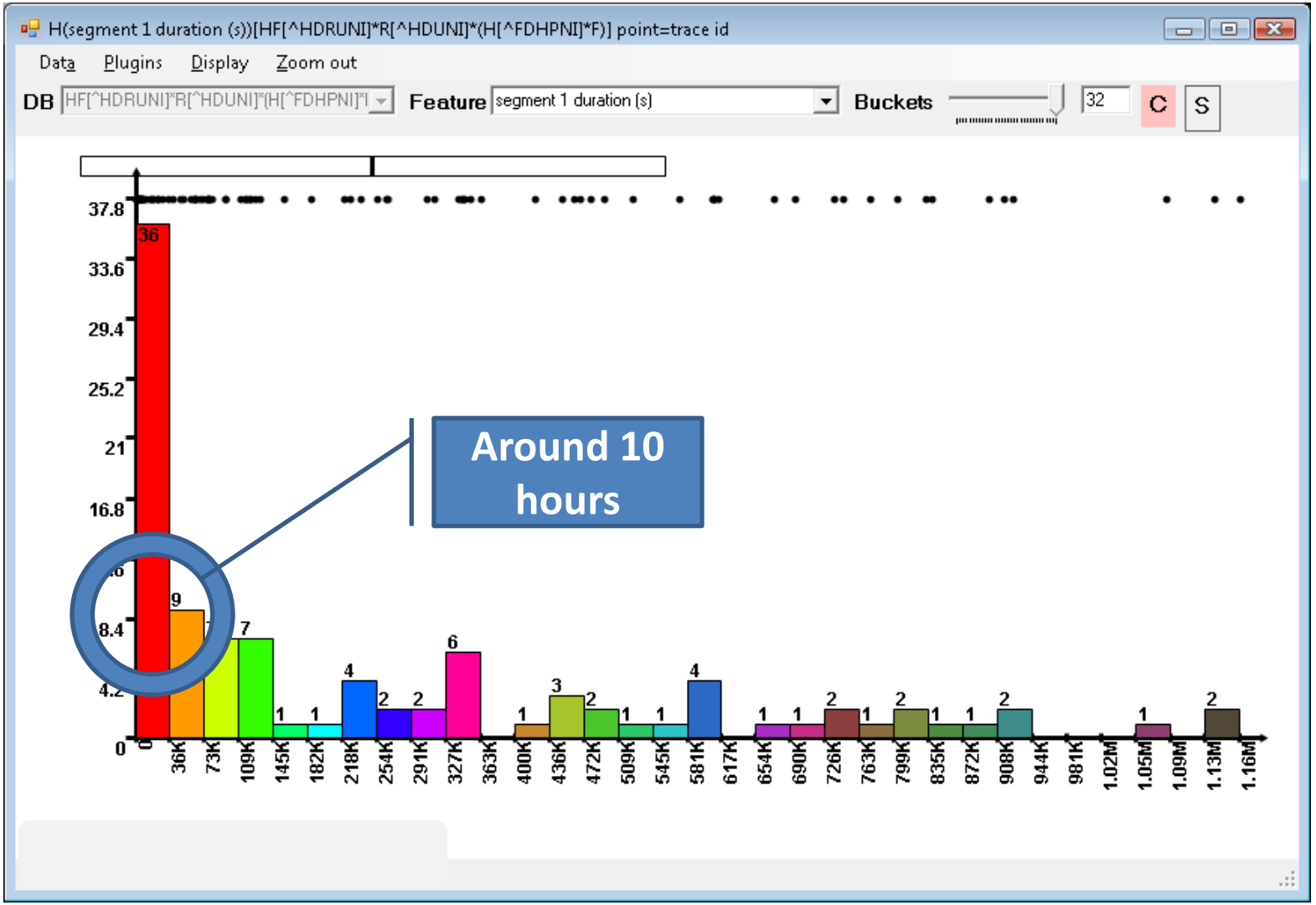
HF[^HDRUNI]*R[^HDUNI]*(H[^FDHPNI]*F)



Enabled to focus attention on a handful of watchdogs.

Why 5 times?
Why that particular one?

Change threshold

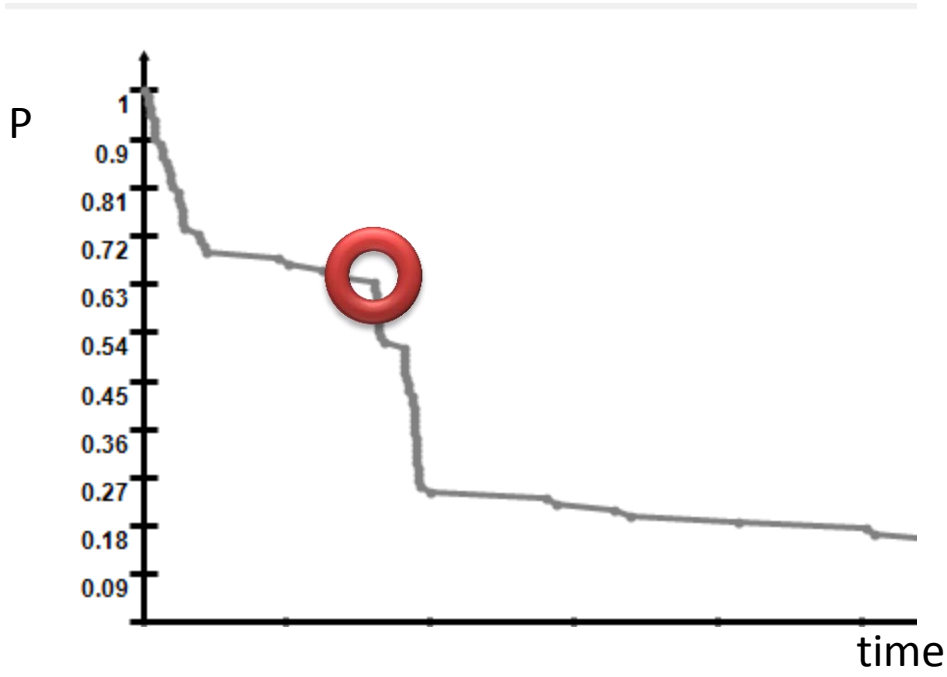


Results

- Faulty Watchdogs
  - Watchdog that checks the service is alive
    - Wrong time loop
  - Watchdog that checks disk controllers came up
    - Wrong time loop
  - Watchdog that relays the disk controllers fault signals
    - Vendor provided the same signal for two different problems one
- Faulty manual repair
- Policy changes
- Highlight differences between datacenters (hardware)

Effectiveness

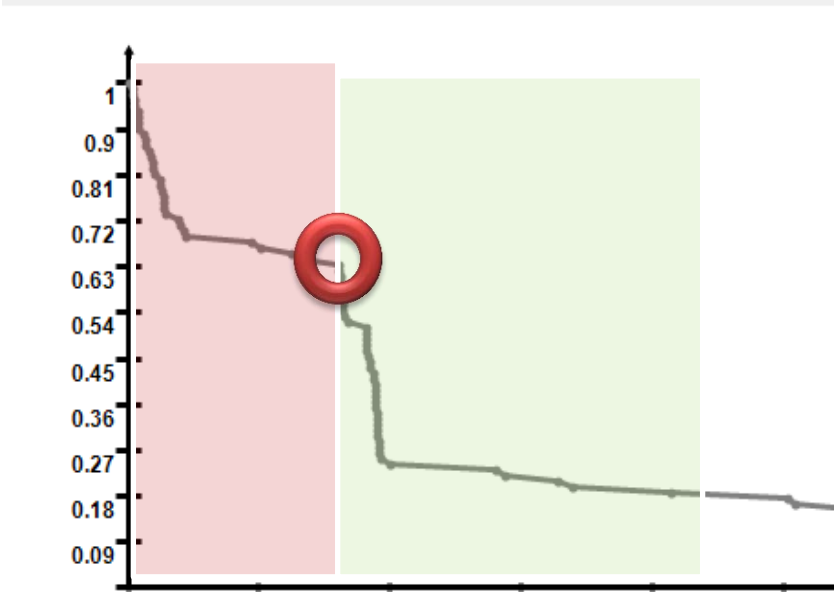
- Effectiveness → time that a machine is 'usable'
- Estimate the survival curve of the repair action



duration2	event2
19	0
134	0
277	1
555	1
572	0
632	1
722	1
827	0
929	1
1429	1
2594	1
2754	1
2828	1
3169	1
3446	1
3937	1

Modeling interesting regions

Automatically find a function from watchdog-signals to regions



Logistic regression with L1 regularization

Who's watching your watchdogs?

- Basic statistics and graphics
  - Histograms of time in healthy state
  - How long for a machine to come to healthy from a failed state
  - What are unhealthy machines doing?
- Model fitting
  - What is the probability that a machine will stay healthy for eight hours after a reboot
  - Which events predict that a machine will not survive for two hours after a repair action
  - Correlated failures
- High level repair
  - Identify faulty watchdogs
  - Identify faulty repair actions
  - Identify changes in repair policy

Currently…

Availability

Correlate violations of QoS with
- Event traces
- Watchdogs
- Vitals

Interesting work on **clustering** time series

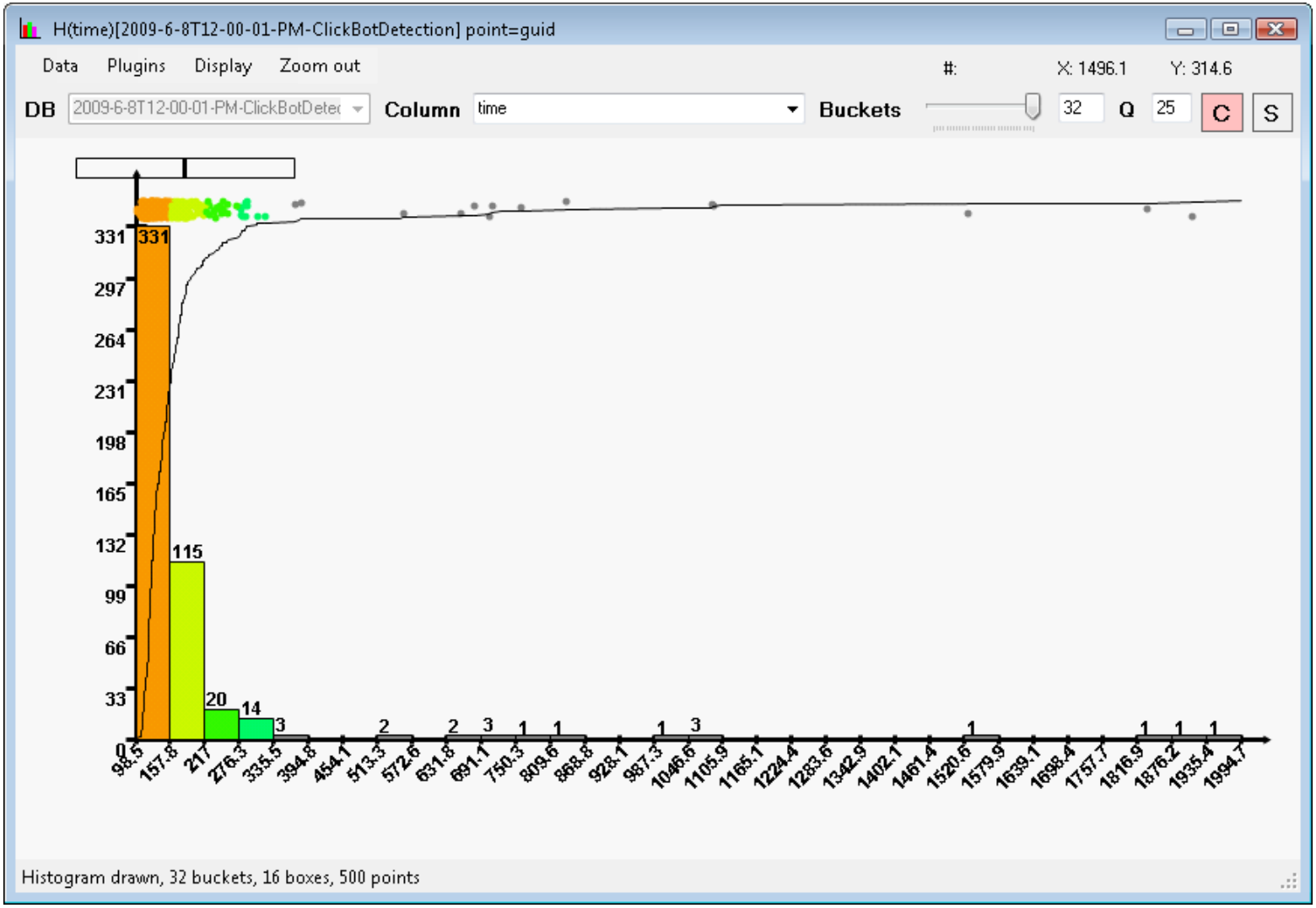
Performance

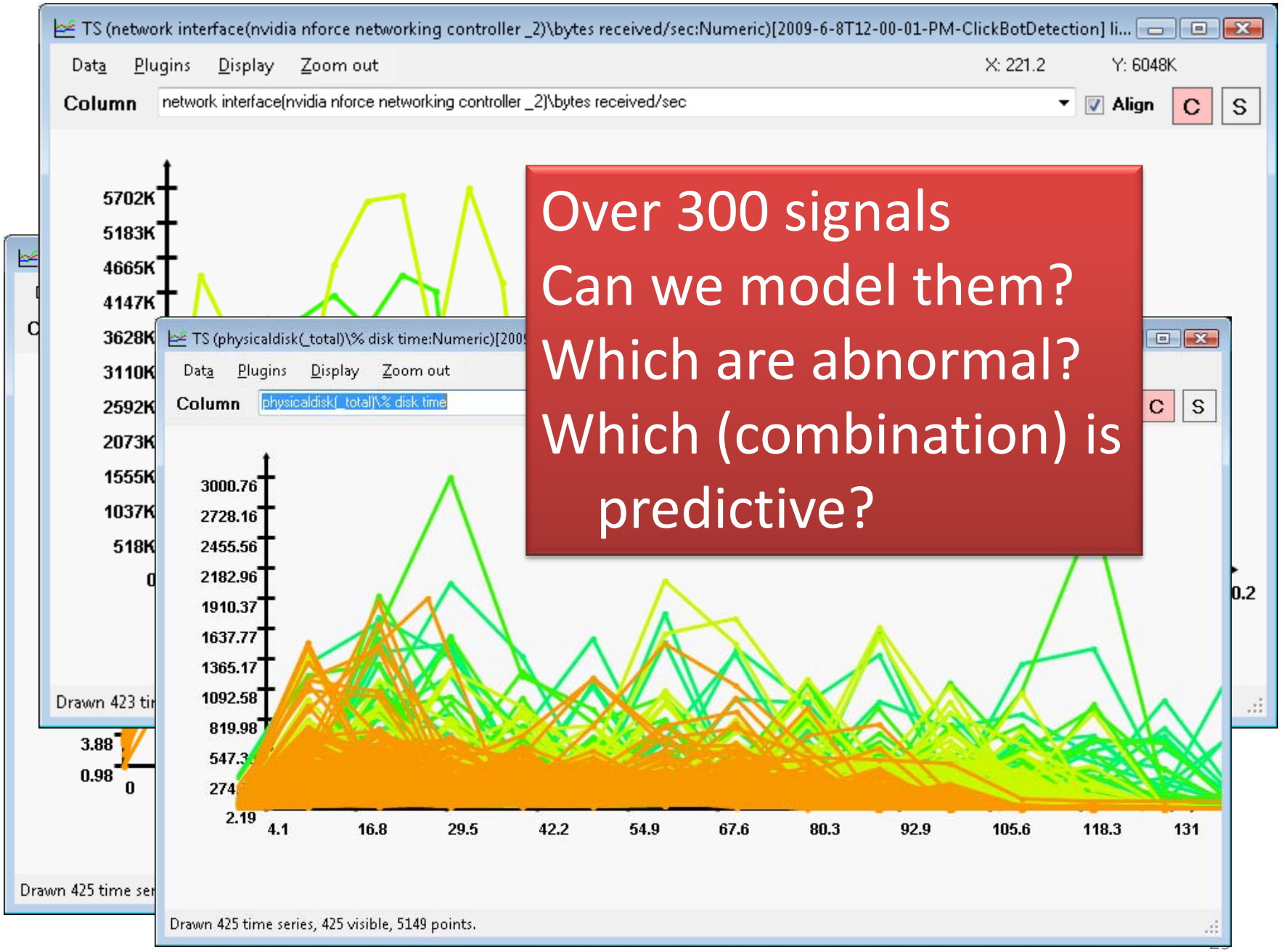
Which of the 300+ signals can predict the running time

On Dryad(LINQ)

Interesting work on **modeling** time series

Why the difference in processing time?




TS (network interface(nvidia nforce networking controller _2)\bytes received/sec:Numeric)[2009-6-8T12-00-01-PM-ClickBotDetection] li...
Data
Plugins
Display
Zoom out
X: 221.2
Y: 6048K
Column
network interface(nvidia nforce networking controller _2)\bytes received/sec
Align
C
S
5702K
5183K
4665K
4147K
3628K
3110K
2592K
2073K
1555K
1037K
518K
0
Drawn 423 ti
3.88
0.98
0
TS (physicaldisk(_total)\% disk time:Numeric)[200
Data
Plugins
Display
Zoom out
Column
physicaldisk(_total)\% disk time
C
S
3000.76
2728.16
2455.56
2182.96
1910.37
1637.77
1365.17
1092.58
819.98
547.3
274
2.19
4.1
16.8
29.5
42.2
54.9
67.6
80.3
92.9
105.6
118.3
131
0.2
Drawn 425 time ser
Drawn 425 time series, 425 visible, 5149 points.
Over 300 signals
Can we model them?
Which are abnormal?
Which (combination) is predictive?

Final Remarks

- With automated repair systems -- diagnosis is changing….
  - (??) Really(??)
- Have data - we need actionable information
- Power of sophisticated statistical analysis coupled with visualization

1. D. Woodard and M. Goldszmidt, Model-Based Clustering for Online Crisis Identification in Distributed Computing, no. MSR-TR-2009-131, (submitted for publication 2010)
2. P. Bodik, M. Goldszmidt, A. Fox, D. Woodard, and H. Andersen, Fingerprinting the datacenter: Automated classification of performance crises, Eurosys 2010
3. M. Goldszmidt, M. Budiu, Y. Zhang, and M. Pechuk, Toward Automatic Policy Refinement in Repair Services for Large Distributed Systems, in *The 3rd ACM SIGOPS International Workshop on Large Scale Distributed Systems and Middleware*, 17 September 2009
4. G. Cretu, M. Budiu, and M. Goldszmidt, Hunting for problems with Artemis, in *USENIX Workshop on the Analysis of System Logs (WASL)*, USENIX, December 2008