INTEGRIEN

Everything you know about monitoring is wrong!

LISA Conference
7 December 2006

INTEGRITY MANAGEMENT FOR ENTERPRISE SYSTEMS™

Agenda

- The Path from Data to Information
- Why More Data Does Not Equal More Information
- Problem Solving 101
- Let's Call It Integrity Management
- A Different Approach to Problem Identification

What is Data?

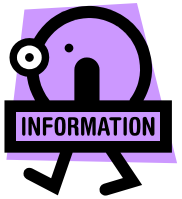
a) Something I measure, count, check,…
b) An entity occupying storage space
c) What I must feed the Sarbanes-Oxley monster
d) An android on Star Trek TNG

What is Information?

a) It's something I can do something about
b) It's what I need to do my job better
c) It's what my manager is really asking for
d) It's what I hold onto for job security

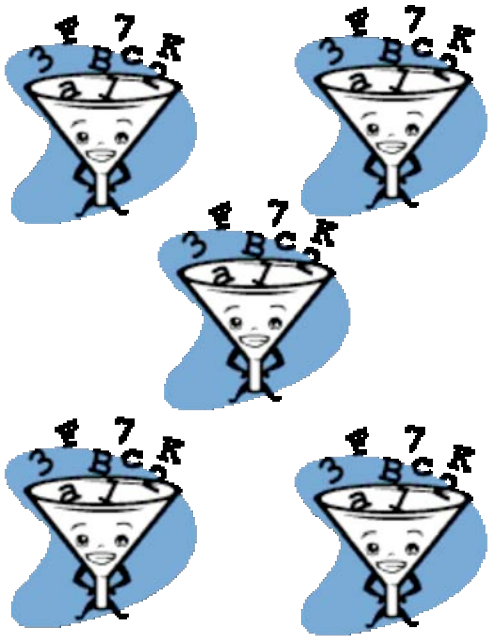
The Path from Data to Information

Manageable Data + Human Synthesizer = Information



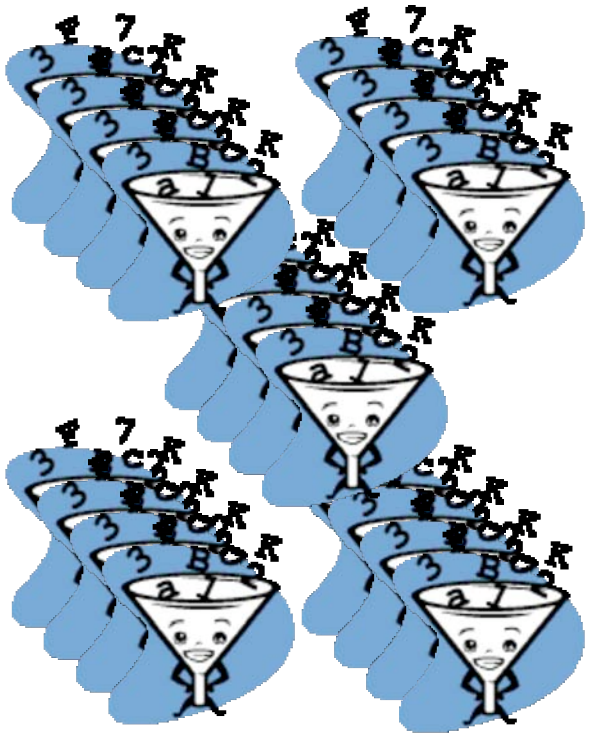
The Path from Data to Information

Lots of Data + Human Synthesizer = ???



The Path from Data to Information

Tons of Data + Computer-Aided Synthesis = ???



Why Are We Collecting More Data?

- "I have more stuff to monitor."
- "Since I don't know what to collect, let's collect everything."
- "If I collect more I'll have a better chance of detecting problems."
- "If I collect more *and* collect it more often, no problem can sneak by me. Call me Sub-second Sam."
- "We've already invested zillions of dollars in tools, let's at least get our money's worth."

The Ultimate Goal for Data Collection

- The purpose of data collection is to get a handle on problems that can impact the business.
- Handling can mean:
  - Early detection
  - Prediction before occurrence
  - Root cause determination
  - Etc.
- The question then becomes, what data to collect and how to synthesize the data

Problem Solving 101

- How can I relate data to problem?
  - A problem may be identified by grouping metric events.
  - A metric event is defined as a fault condition placed on the metric (e.g. threshold violation).
  - A problem may be identifiable via several unique groupings of metric event sequences. For example problem P1 may be identified by:
    - Group 1 = M1-M2-M3-M4-M5
    - Group 2 = M6-M1-M8-M7-M11-M18-M21
    - Group 3 = M10-M13-M14-M23-M45-M71-M19
    - Etc.
  - Define Problem DNA as any grouping of metric events

Problem Solving 101

- Due to the inherent difficulty of identifying long sequences of metric faults, we have been forced to search for the elusive one or two metrics that can identify a problem.
- This has precipitated the need to collect as many metrics as possible so the needle in the haystack can be found.
- Add this to the ever-growing complexity of today's applications and infrastructure and you get data overload.

Problem Solving 101

- Solving this problem requires a new way of thinking about data collection.
- To demonstrate: I created a simulated IT environment where metrics are collected, problems are randomly generated, and the costs of both problem identification and metric collection are considered to arrive at the optimal approach.
- The simulated environment consisted of the following:
  - **A 500 server environment with an average of 100 metrics per server available to be collected (total of 50,000 metrics).**
  - **A total of 500 problems can occur in this environment.**
  - **The occurrence of problems is governed by an exponential function (i.e. 25% of the problems occur 75% of the time).**

Getting the Most Bang for your Buck

- **An efficient operational environment needs to balance the costs of more data collection vs. speed to problem identification.**

Total Cost ( ) = Problem isolation cost ( ) + metric processing cost ( )

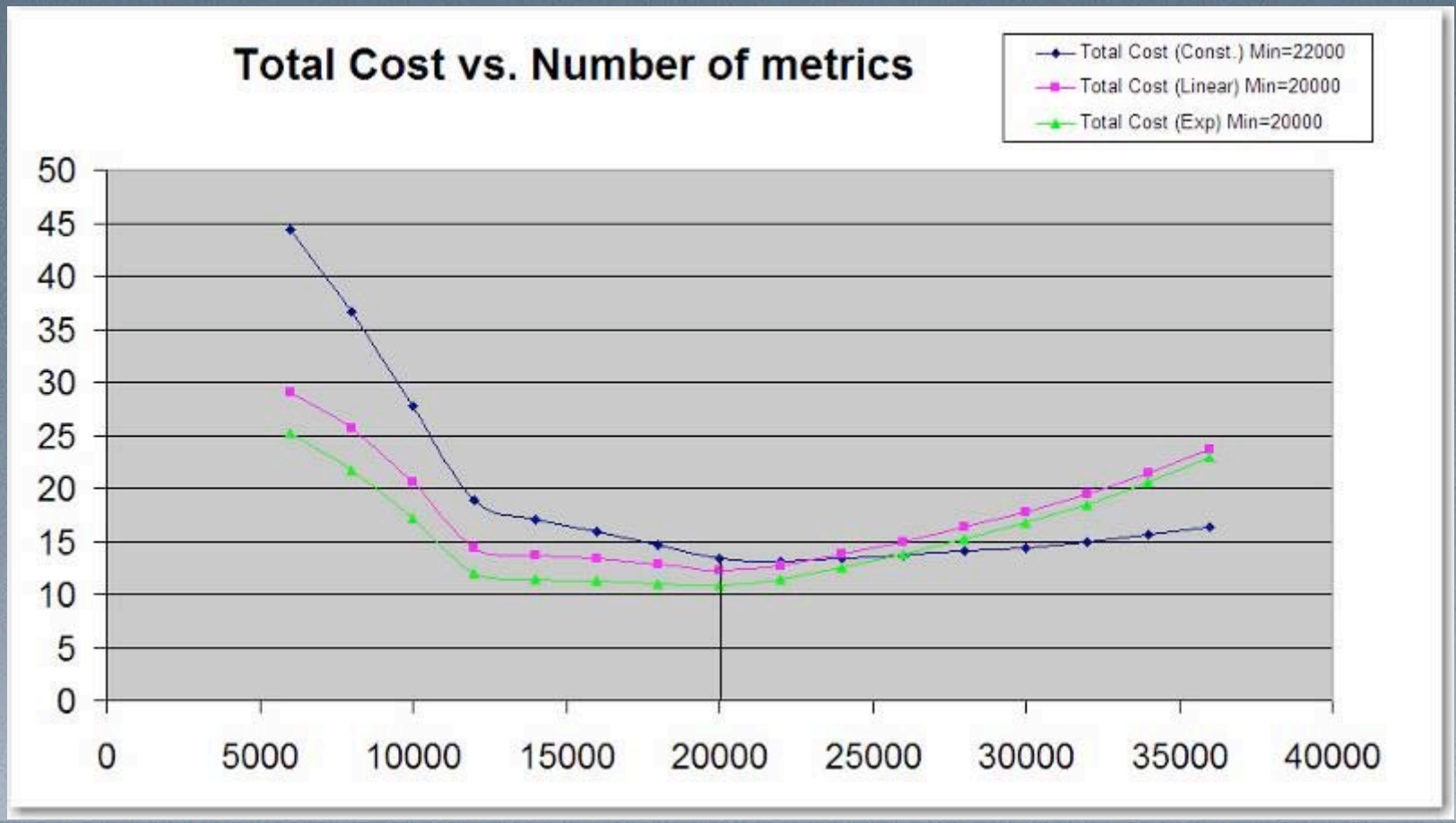
Where E is the cost to process one problem and F is the cost to process one alert and one metric. Three different models for E and F:

1-

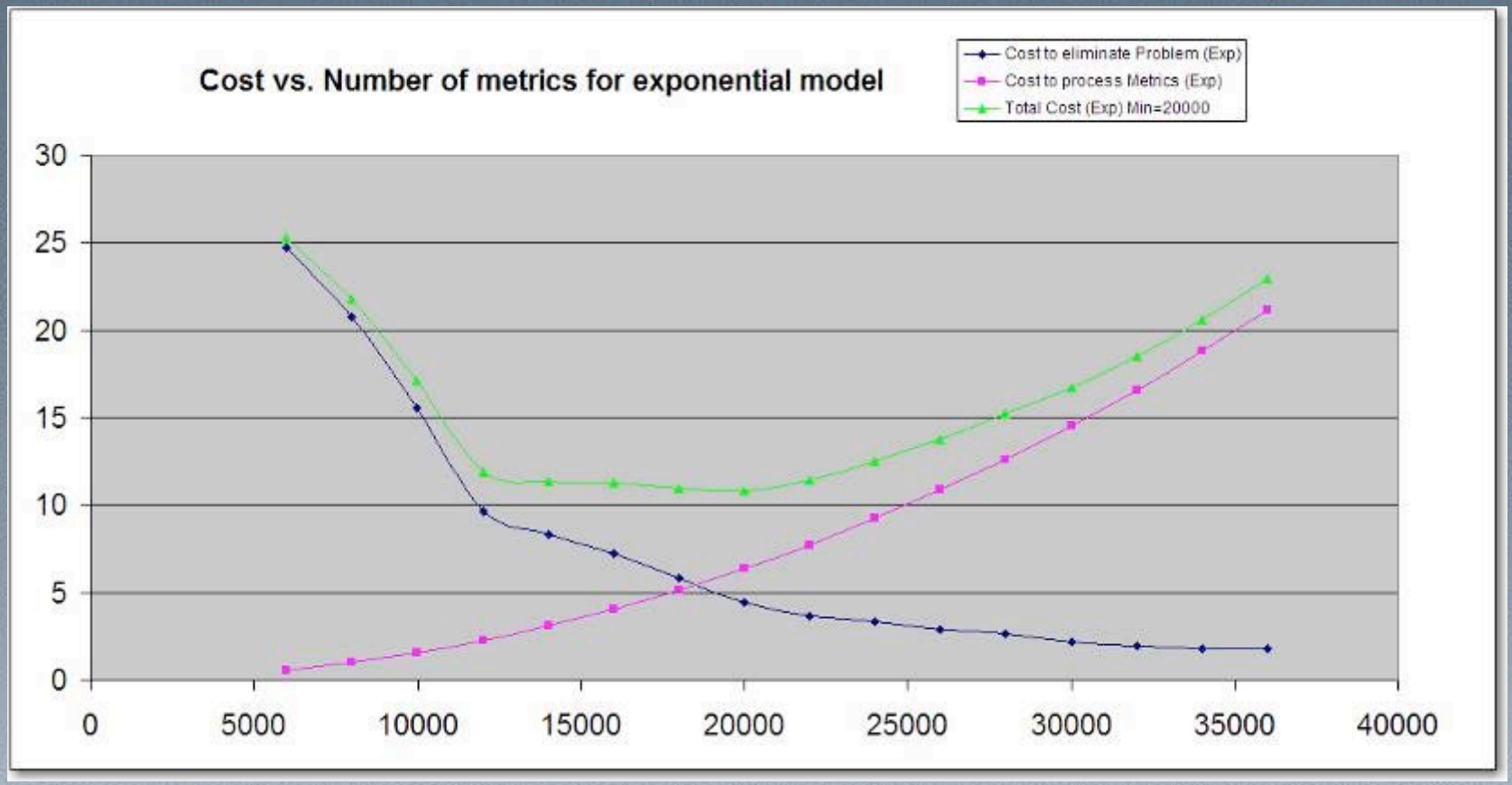
2-

3-

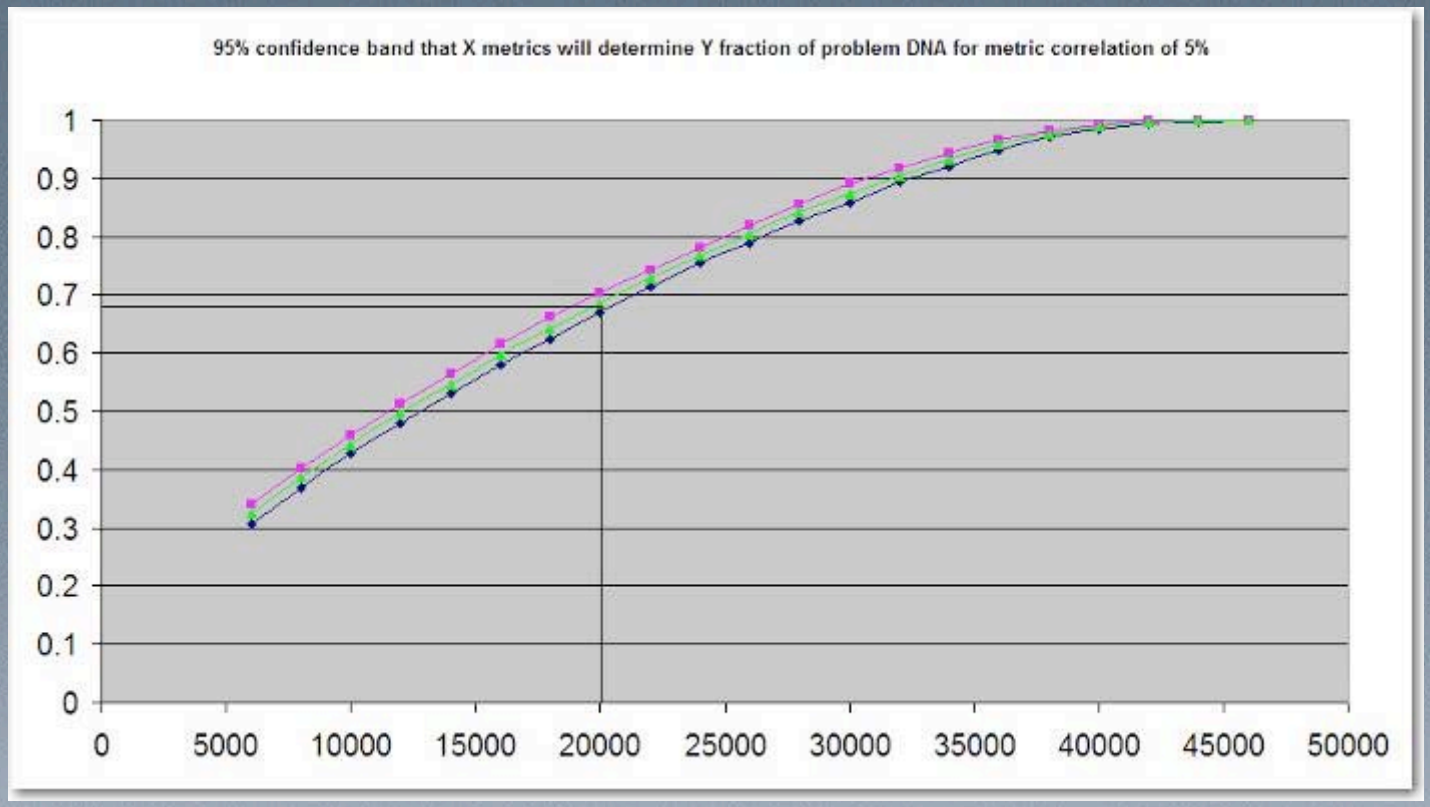
Cost of Finding Problems



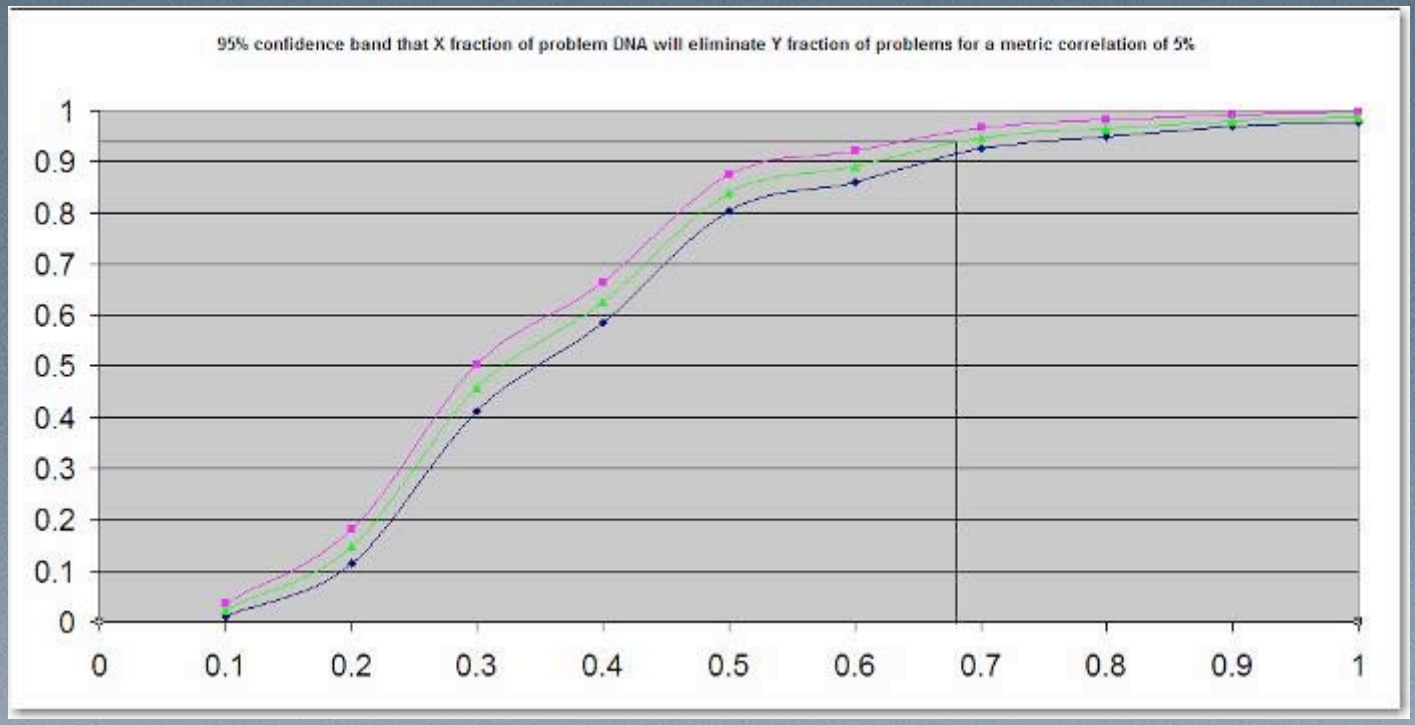
Cost of Finding Problems For Exponential Model



So, What's the Answer?



So, What's the Answer?

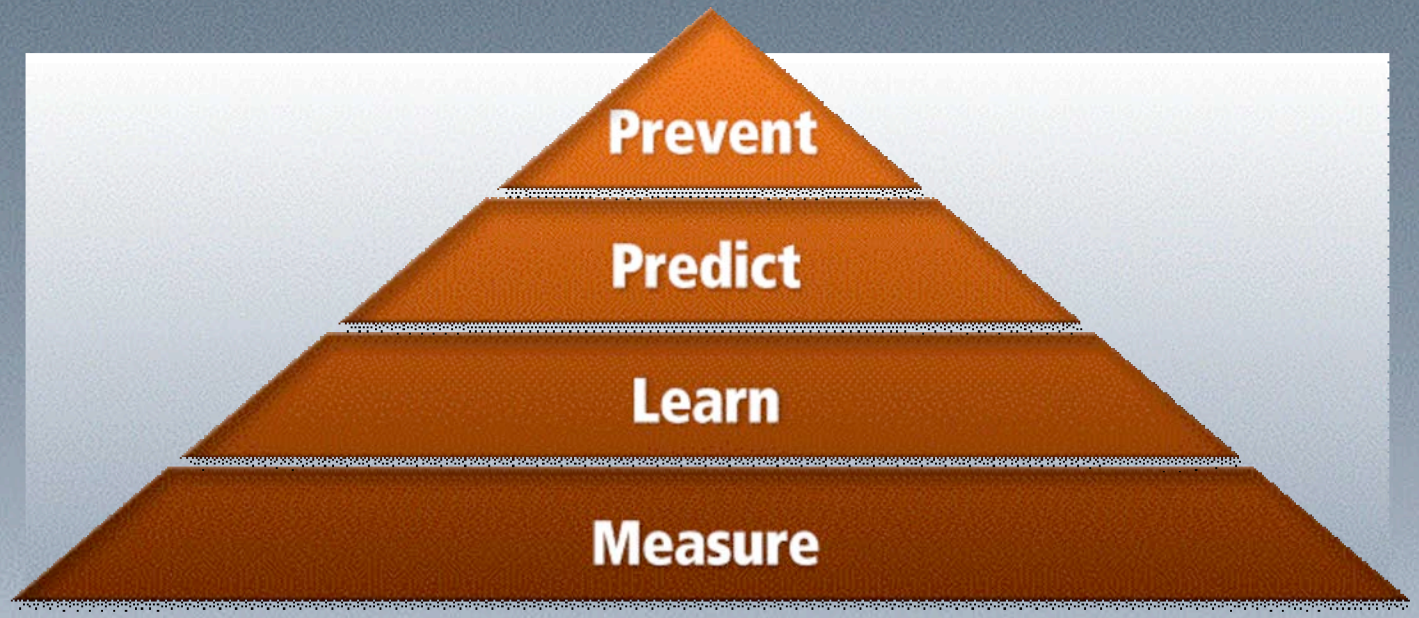


What Was that Again?

- Without knowing which metrics, the most cost-effective way to solve problems is to randomly select 40% of metrics that are collected.
- By collecting only 40% of the metrics, on the average 68% of problem DNA's can be identified for the most commonly occurring problems.
- By collecting 40% of the metrics, 94% of the most commonly occurring problems can be eliminated.
- The question then becomes how can I identify and relate these long chained groups of metric events? The answer is Integrity Management.

What is Integrity Management

1. View and Measure Service
2. Learn What's Normal
3. Capture Problem Patterns for Prediction
4. Detect Recurring Problem Patterns for Prevention

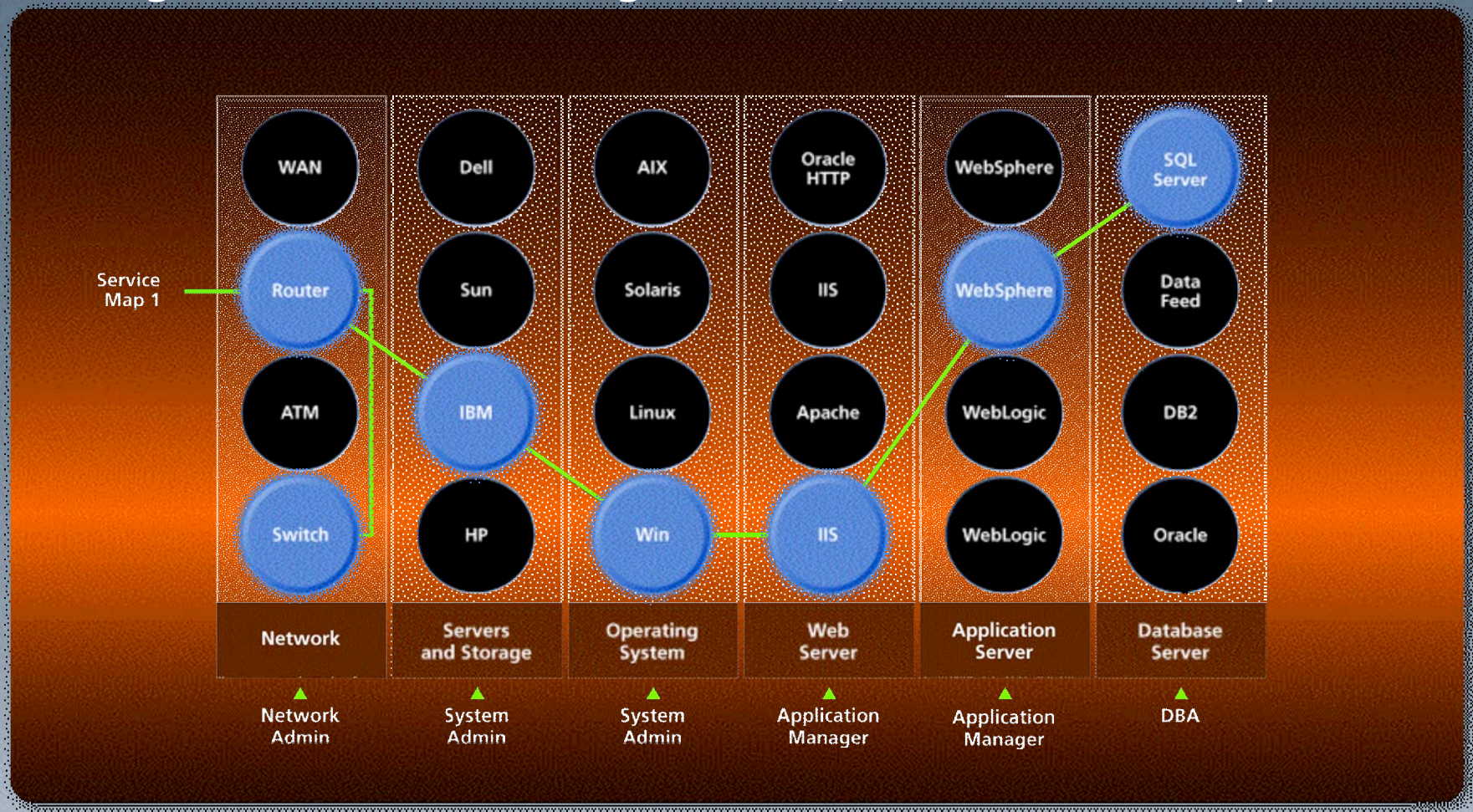


The Problem Integrity Management Solves

- Faster Resolution of App Slowdowns and Outages by collecting fewer metrics and using learning based system for problem identification.
- Contrary to ITIL, this should not be a post-mortem exercise
- Benefits
  - 1st occurrence: Post-mortem, but significantly reduced duration
  - Recurrences: Near zero duration of impact
  - Troubleshooting: Significantly reduced IT labor and disruption
  - Bottom line impact: What's your cost of slowdowns/downtime?

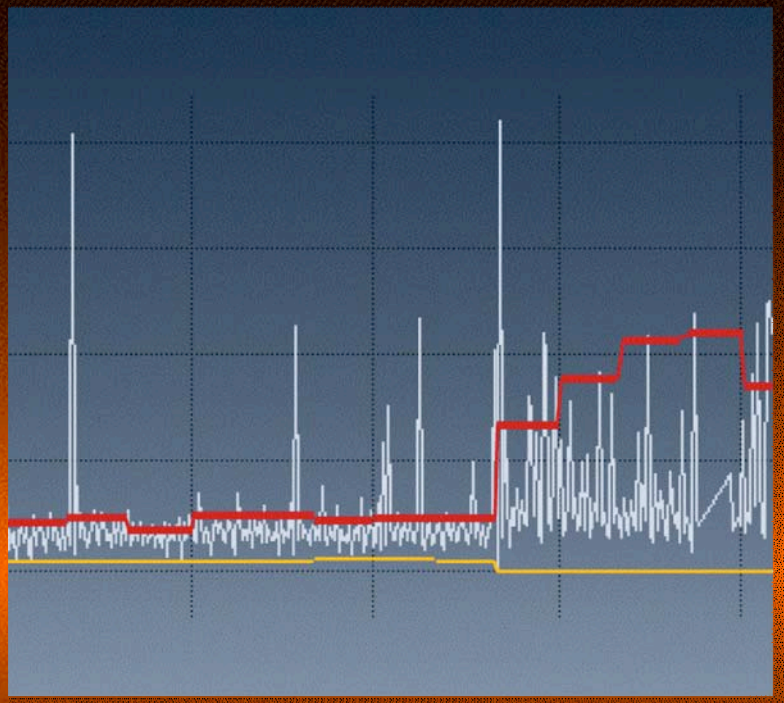
1. View and Measure Service

Right device metrics gathered, measured as application

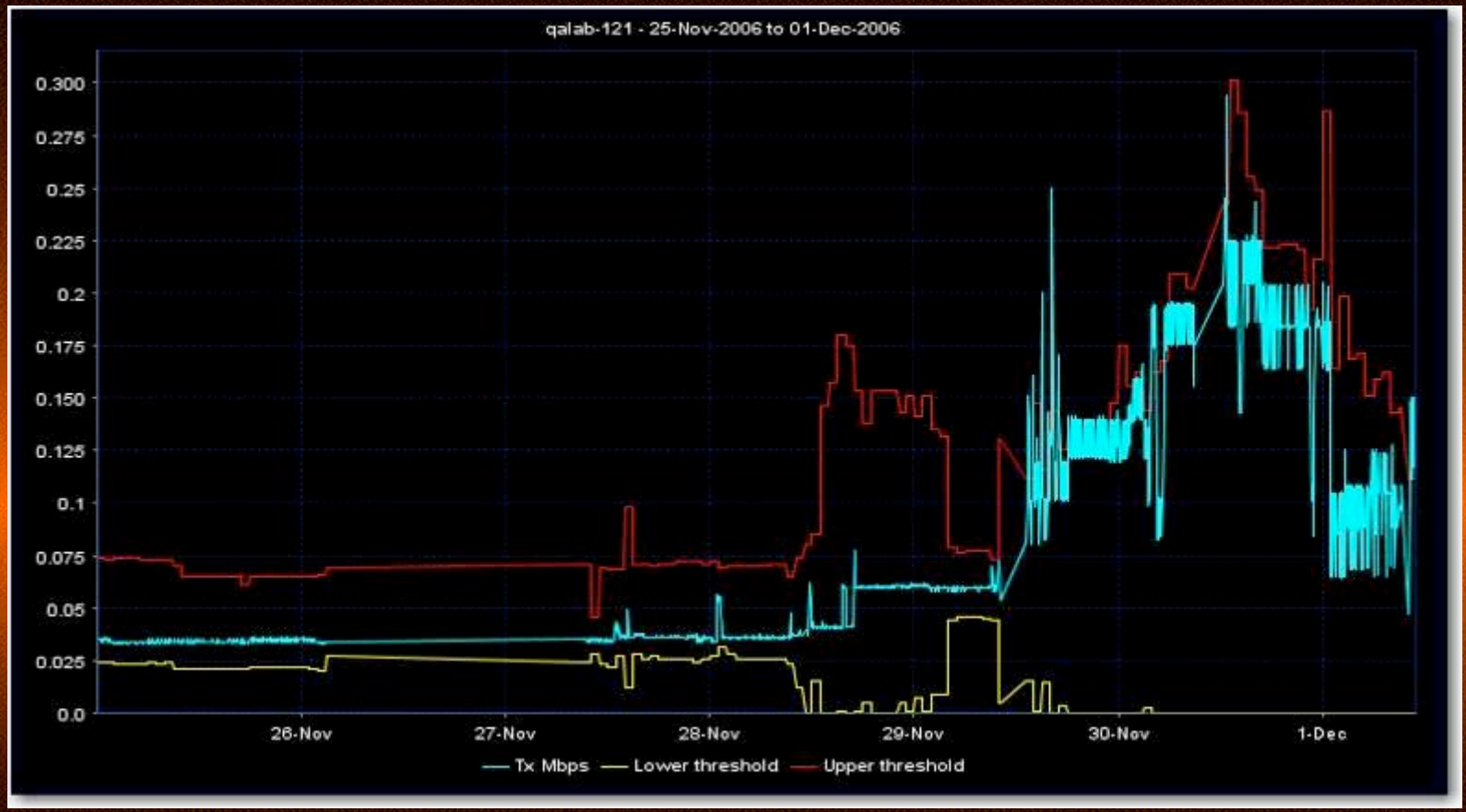


2: Learn Normal
Dynamic Thresholds of all metrics

- Dynamic thresholding to learn normal metric levels to day of week and hour of day
  - You don't have to guesstimate thresholds
  - Sustained departure from normal generates a blip
- Abnormal behavior "blips" precede problems
  - Most "blips" require no action but can guide application tuning
- Detect cracks in the dam, not just water over the top

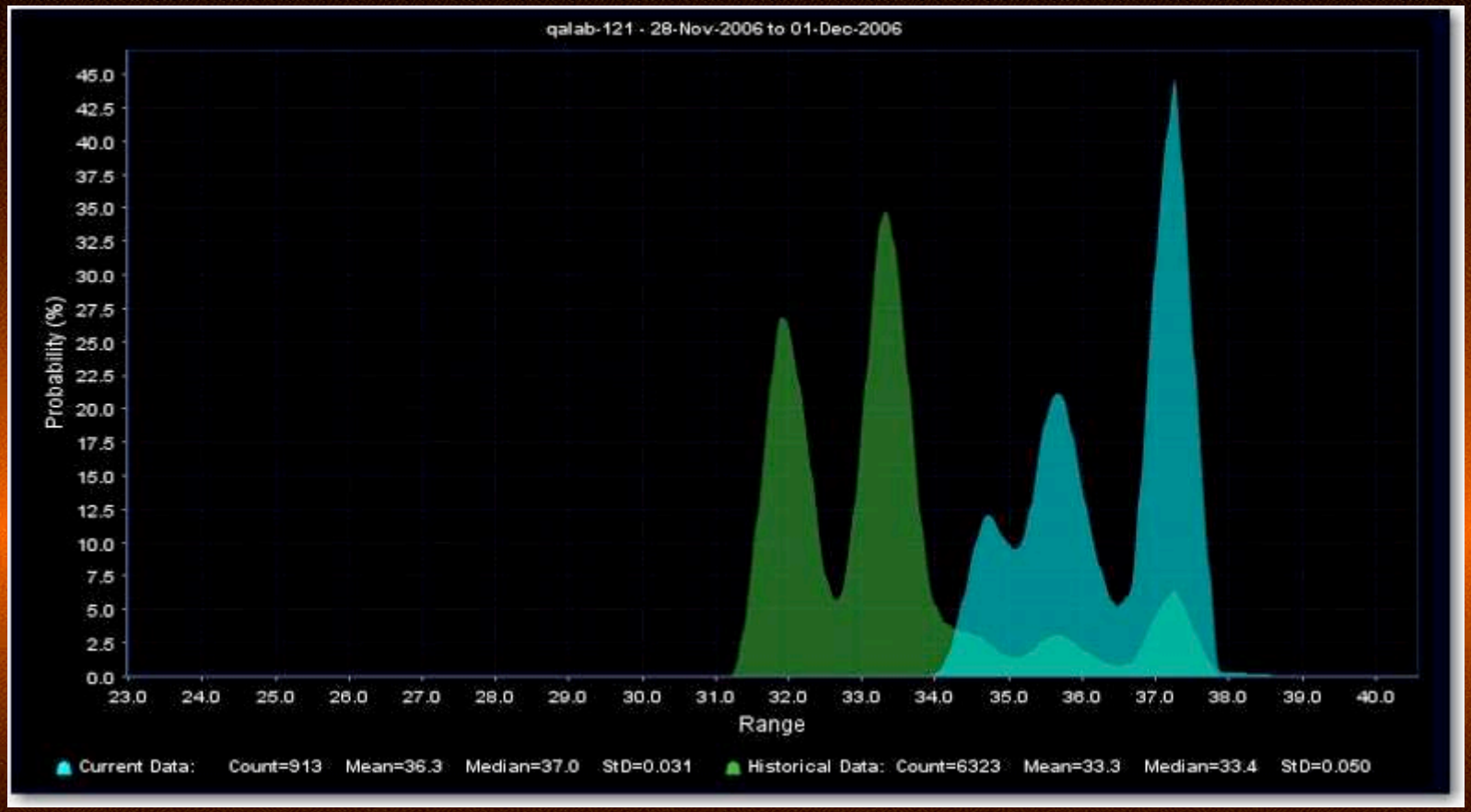


2a: Learn Normal
The Futility of Setting Thresholds



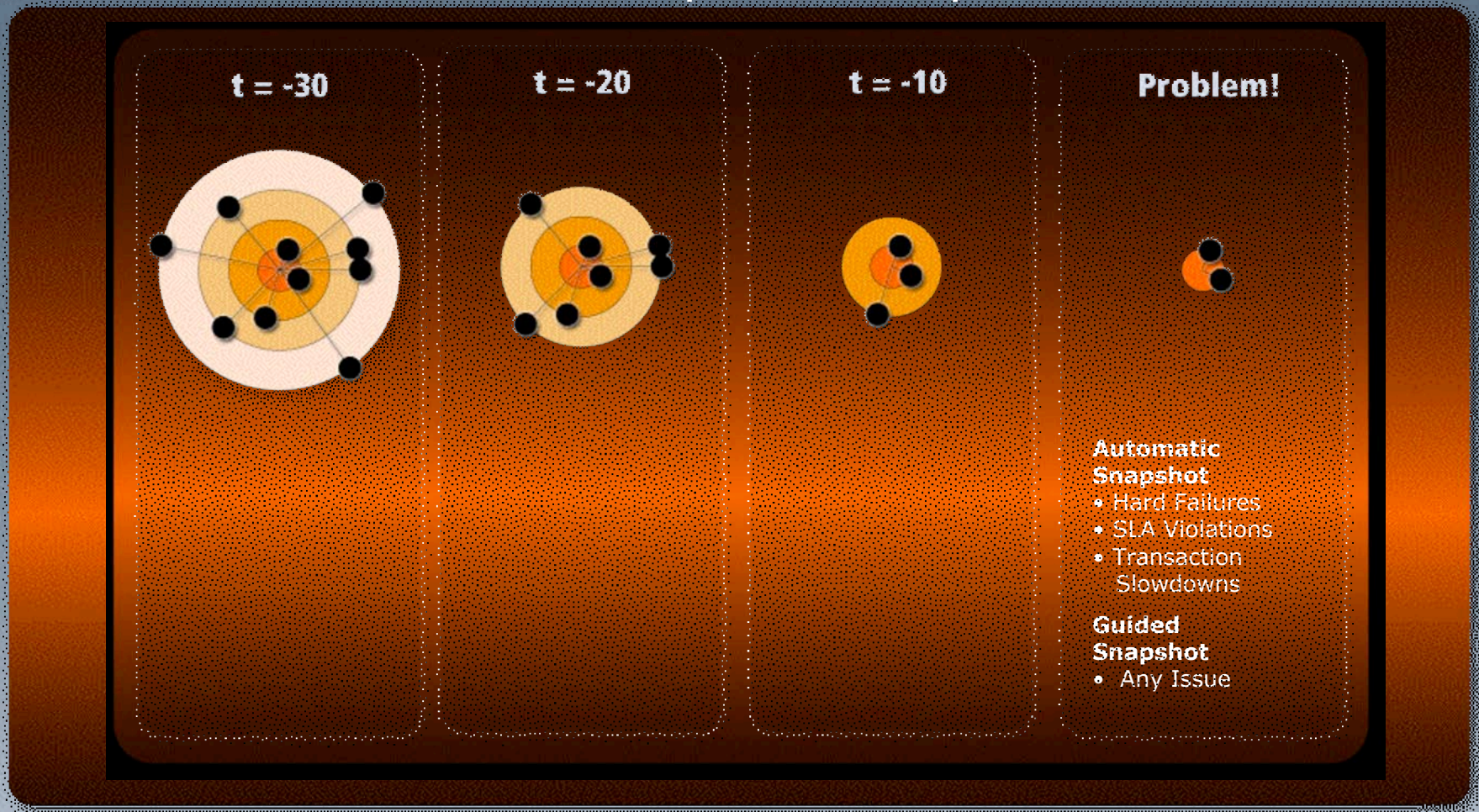
2b: Learn Normal

The Value of Knowing How Things Change

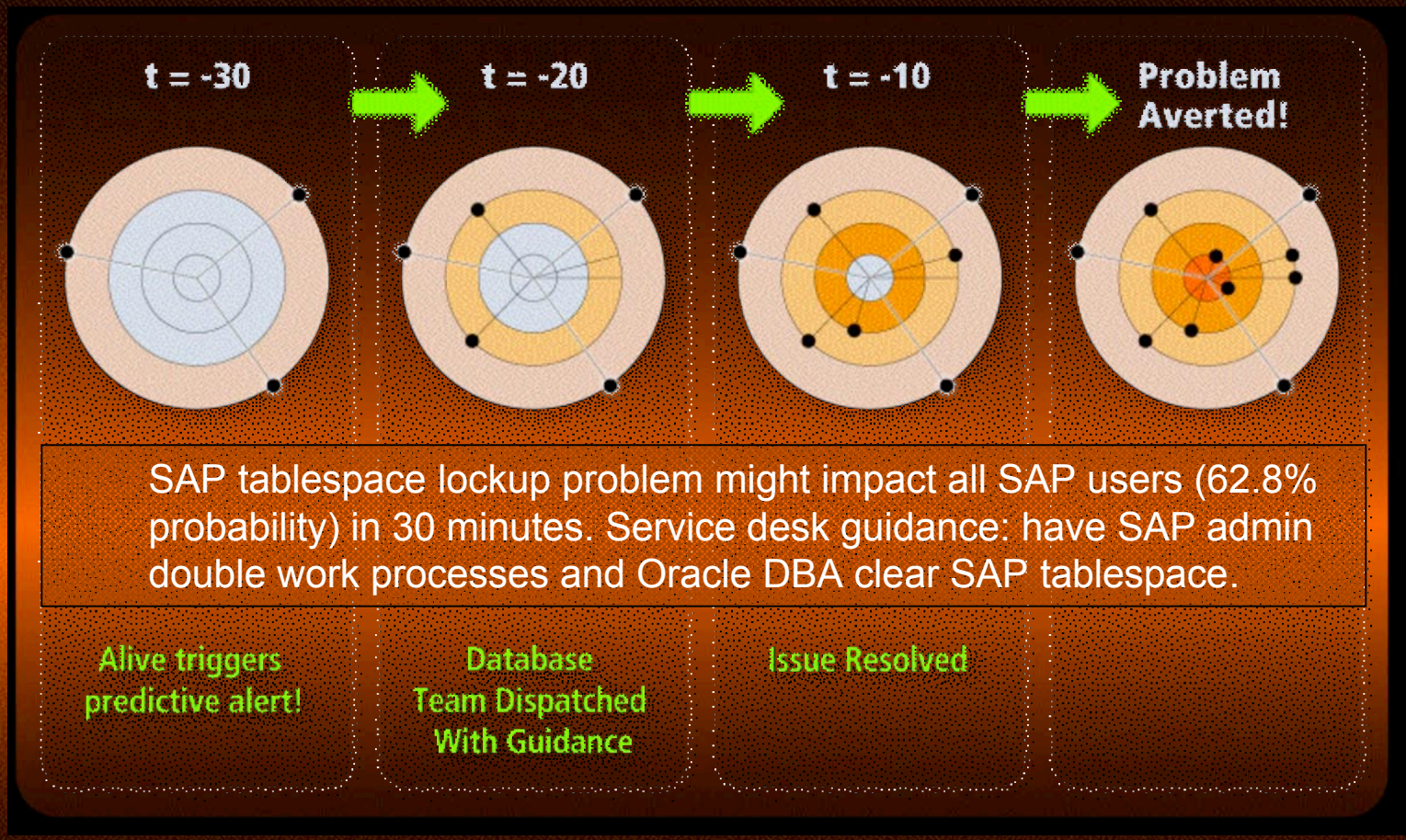


3: Learn to Identify Problem Patterns

"*Problem Precursor*" Capture = *Rapid Resolution*

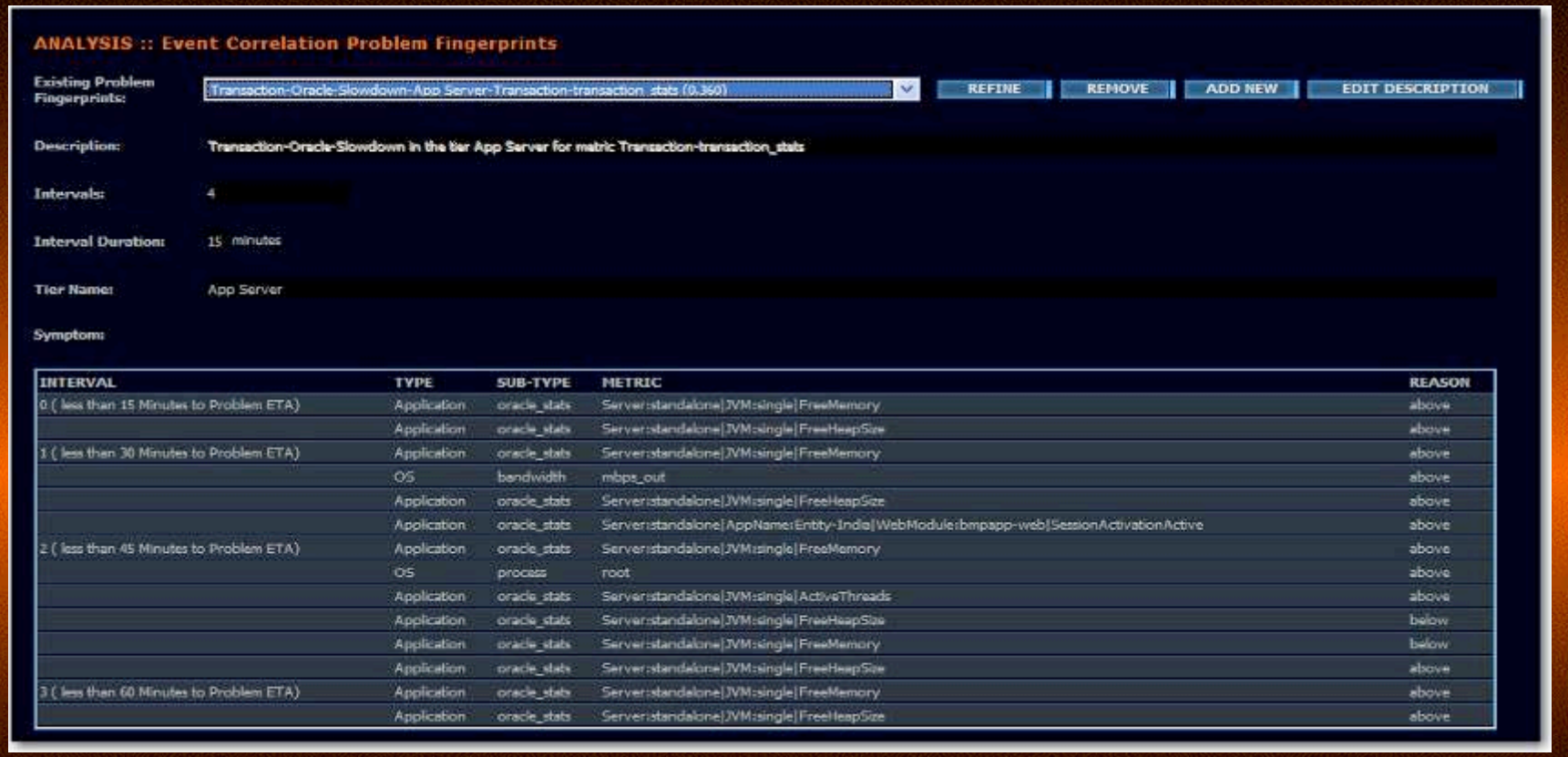


4: Predict and Prevent
Precursor Pattern Detection *predicts problems*



4a: Predict and Prevent

What would one of these things look like?



ANALYSIS :: Event Correlation Problem Fingerprints

Existing Problem Fingerprints: Transaction-Oracle-Slowdown-App Server-Transaction-transaction_stats (0.360) | REFINE | REMOVE | ADD NEW | EDIT DESCRIPTION

Description: Transaction-Oracle-Slowdown in the tier App Server for matric Transaction-transaction_stats

Intervals: 4

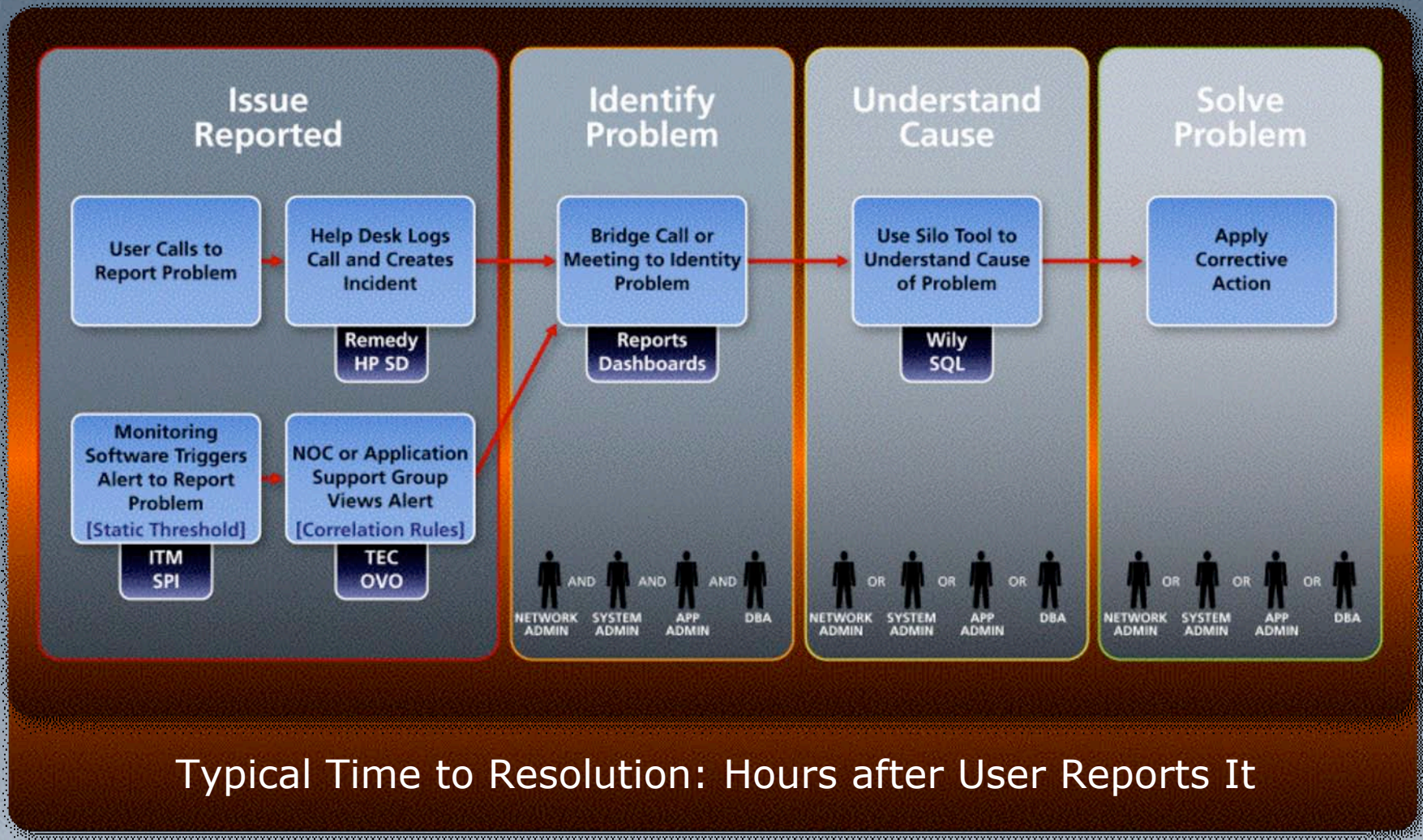
Interval Duration: 15 minutes

Tier Name: App Server

Symptom:

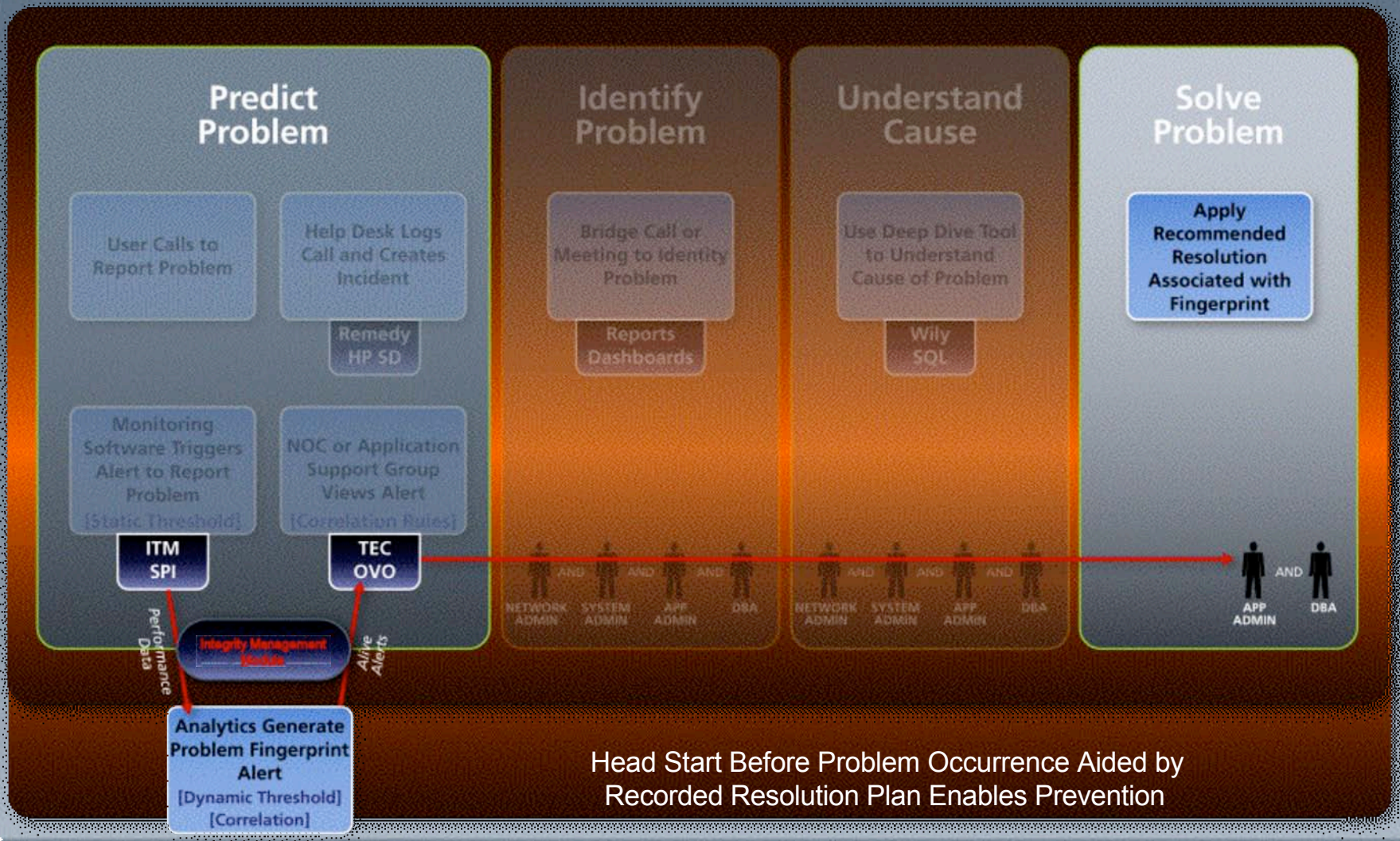
INTERVAL	TYPE	SUB-TYPE	METRIC	REASON
0 (less than 15 Minutes to Problem ETA)	Application	oracle_stats	Server:standalone|JVM:single|FreeMemory	above
	Application	oracle_stats	Server:standalone|JVM:single|FreeHeapSize	above
1 (less than 30 Minutes to Problem ETA)	Application	oracle_stats	Server:standalone|JVM:single|FreeMemory	above
	OS	bandwidth	mbps_out	above
	Application	oracle_stats	Server:standalone|JVM:single|FreeHeapSize	above
	Application	oracle_stats	Server:standalone|AppName:Entity-India|WebModule:bmpapp-web|SessionActivation|Active	above
2 (less than 45 Minutes to Problem ETA)	Application	oracle_stats	Server:standalone|JVM:single|FreeMemory	above
	OS	process	root	above
	Application	oracle_stats	Server:standalone|JVM:single|ActiveThreads	above
	Application	oracle_stats	Server:standalone|JVM:single|FreeHeapSize	below
	Application	oracle_stats	Server:standalone|JVM:single|FreeMemory	below
	Application	oracle_stats	Server:standalone|JVM:single|FreeHeapSize	above
3 (less than 60 Minutes to Problem ETA)	Application	oracle_stats	Server:standalone|JVM:single|FreeMemory	above
	Application	oracle_stats	Server:standalone|JVM:single|FreeHeapSize	above

Problem Resolution Process Today



Typical Time to Resolution: Hours after User Reports It

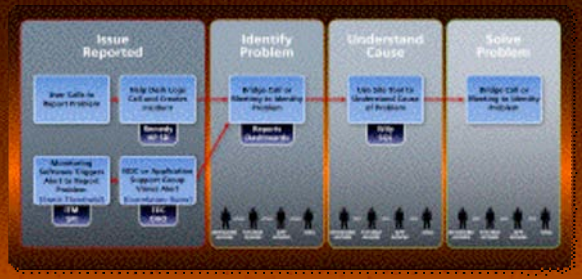
Problem Resolution With Integrity Managment



Before and After Integrity Management

Before: Manual Firefighting

Help desk asks application support team to troubleshoot <u>after</u> users call



Bridge call ***<u>after</u>*** problem occurrence

- Individual silos look good, reporting 99.9% uptime
  - So how come the application is down?
- Early warnings missed
  - Or detected as a meaningless alert storm
- Difficult, after-the-fact root cause analysis
  - No shared view of "just what's involved"
  - Any "automation" involves labor-intensive rules
- Heavy cost of slowdown and downtime
  - IT productivity, business impact

Integrity Management: Automated Prevention

Early warning with recommendation "Have SAP admin double work processes, DBA clear SAP tablespace"

Problem predicted, staff dispatched

- Application-focused Dashboard
  - Health of application at a glance, SLA tracking
- Early warning of problems
  - Dynamic thresholds pick up abnormal behavior
- Problem Precursor pattern match
  - Operations desk gets early warning
  - Location, symptom, ETA, your resolution plan
- Minimized business impact on:
  - Productivity, Revenue, Brand Value

Integrity Management Demands A New Approach To IT Measurement

- Less is More
- Measure Devices in the Context of a Service
- Measure for Deviations from Normal
- Align Measurements in Time
- Statistical Sophistication Required

Questions?