The Machine Learning behind the Autonomous Database

LAD – Oracle Groundbreakers

Sandesh Rao
VP AIOps , Autonomous Database

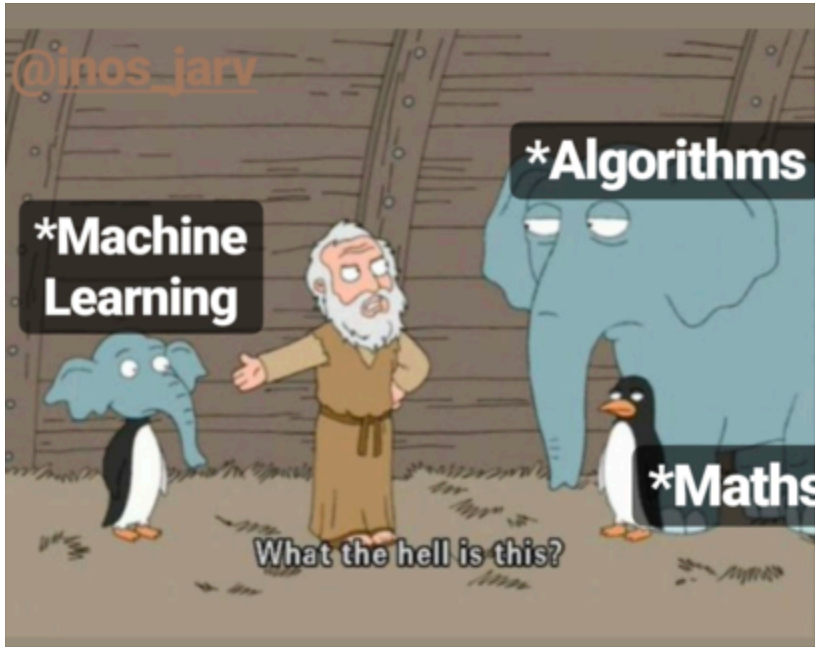
@sandeshr
https://www.linkedin.com/in/raosandesh/
https://www.slideshare.net/SandeshRao4

Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, timing, and pricing of any features or functionality described for Oracle's products may change and remains at the sole discretion of Oracle Corporation.

whoami

- Real Application Clusters - HA
- DataGuard-DR
- Machine Learning-AIOps
- Enterprise Management
- Sharding
- Big Data
- Operational Management
- Home Automation Geek



@sandeshr
https://www.linkedin.com/in/raosandesh/

Agenda

- Architecture for the AIOps platform for the Autonomous Database
- Which algorithms, tools & technologies are used?
- Oracle use cases for – AIOps in Autonomous Database
- Questions and Open Talk

Why Machine Learning for us and why now?

- Lots of Data generated as exhaust from systems
  – Cloud , different formats and interfaces , frameworks
- Machine Learning has become accessible
  – Anyone can be a Data Scientist
  – Algorithms are accessible as libraries aka scikit , keras , tensorflow ..
  – Sandbox to get started as easy as a docker init
- Business use cases
  - How to find value from the data , fewer guesses to make decisions

ORACLE

AIOps Cloud Operations – 3 Strategic Pillars

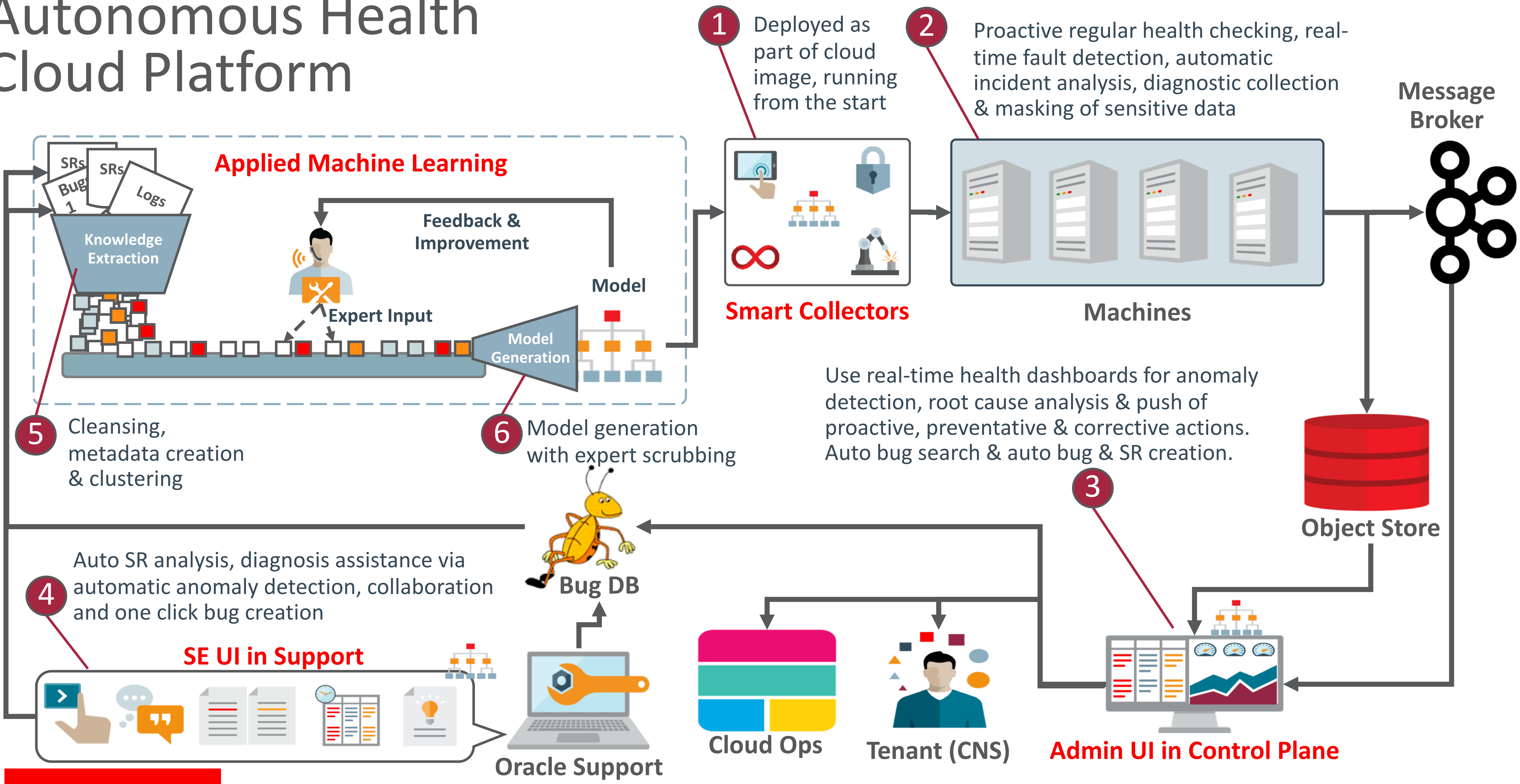
Resource Lifecycle Management
Bare-Metal thru Installation
Upgrade
Patching
Dependency Resolution
Prerequisites Resolution
Required Capabilities
Automatable
Scalable
Online (if possible)

Database Lifecycle Management
Installation
Upgrade
Patching
Dependency Resolution
Prerequisites Resolution
Workload Profile Identification
Placement determination
SLA management
Required Capabilities
Automatable
Provider Interoperable

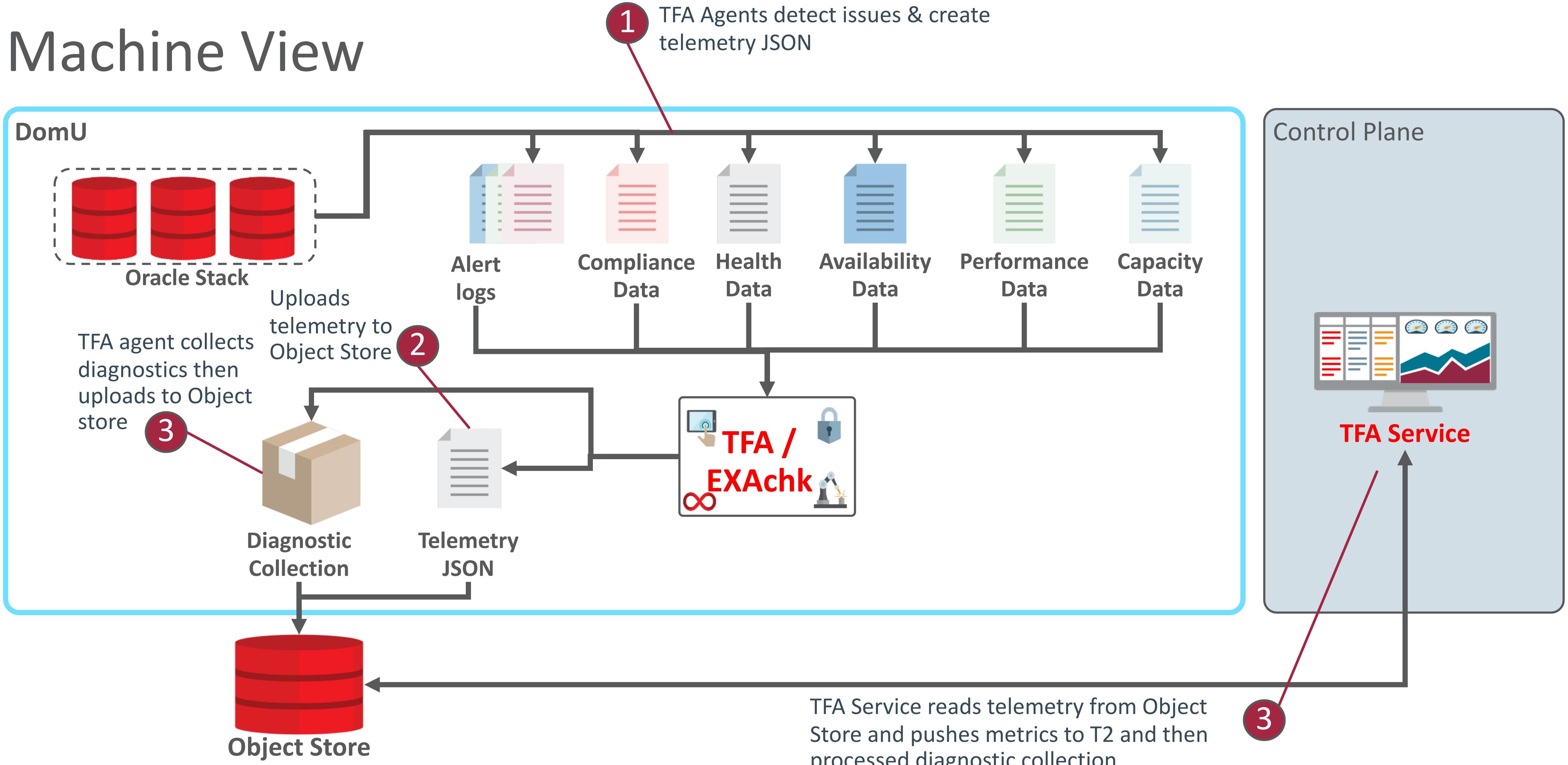
Database Autonomous Self-Repair
Detect degradations and faults
Pinpoint root cause & component
Push warnings and alerts
Push targeted corrective actions
SLA – based resource management
Real-time Health Dashboard
Required Capabilities
Continuous and frequent
Autonomous Action Enabled
OSS Integration Enabled
Management Interoperable

ORACLE

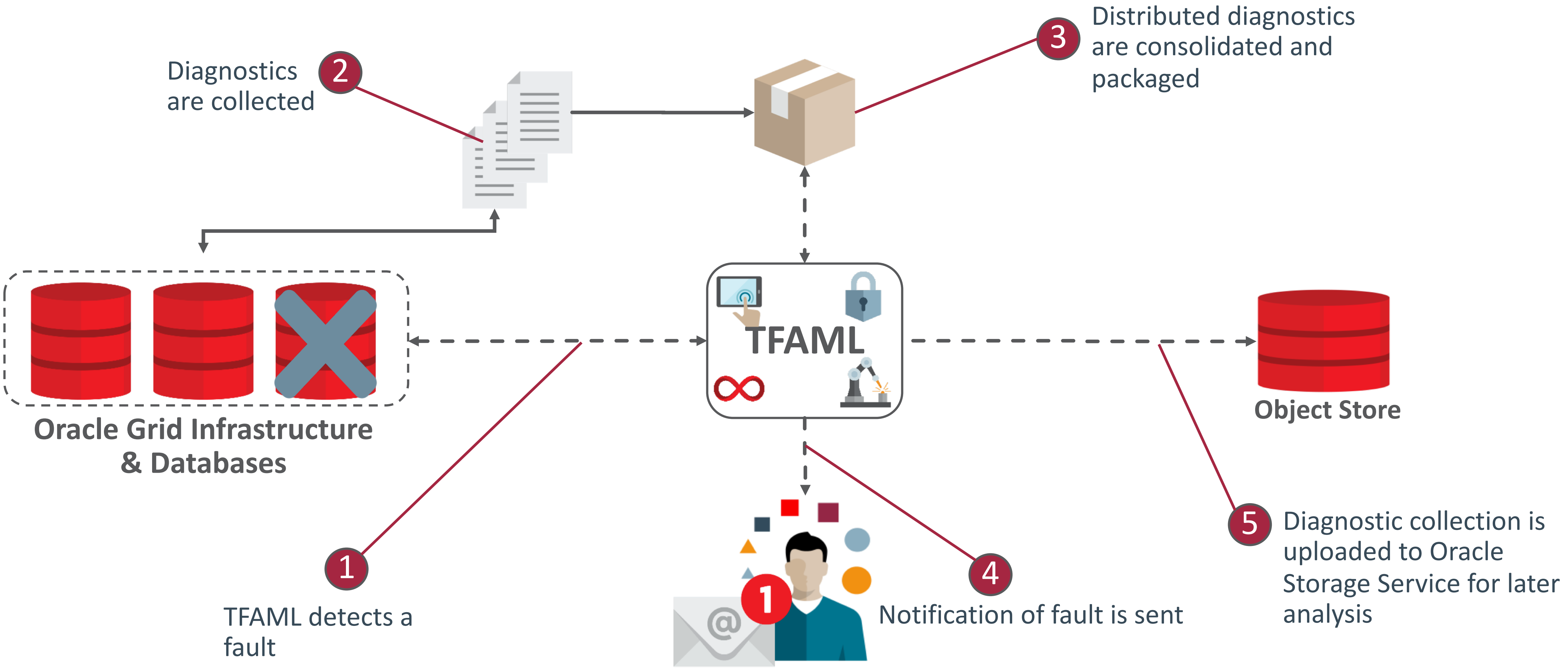
Autonomous Health Cloud Platform



Machine View



SRDCs (Service Request Diagnostic Collection)

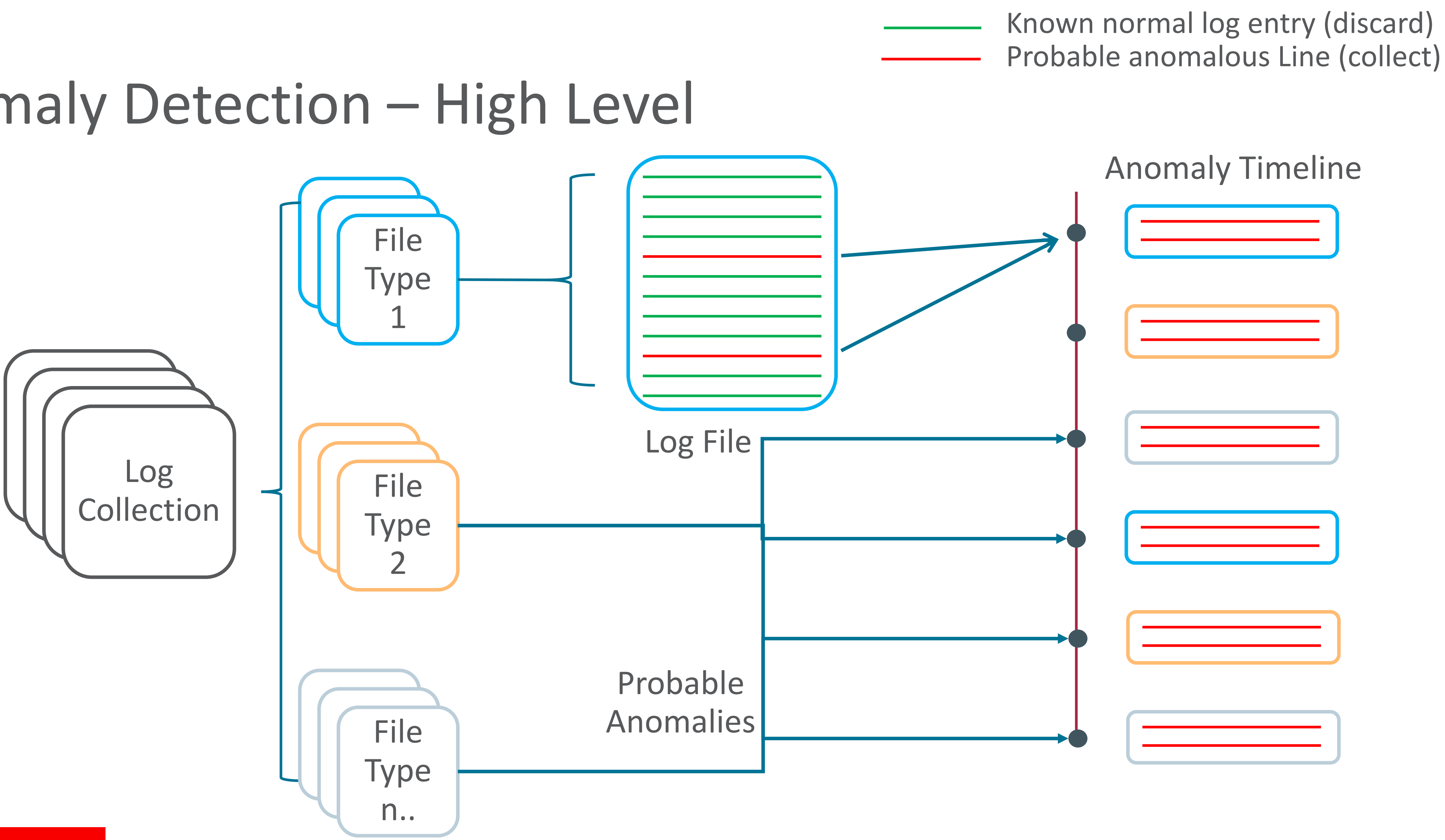


Autonomous Database Health - Anomaly Timeline

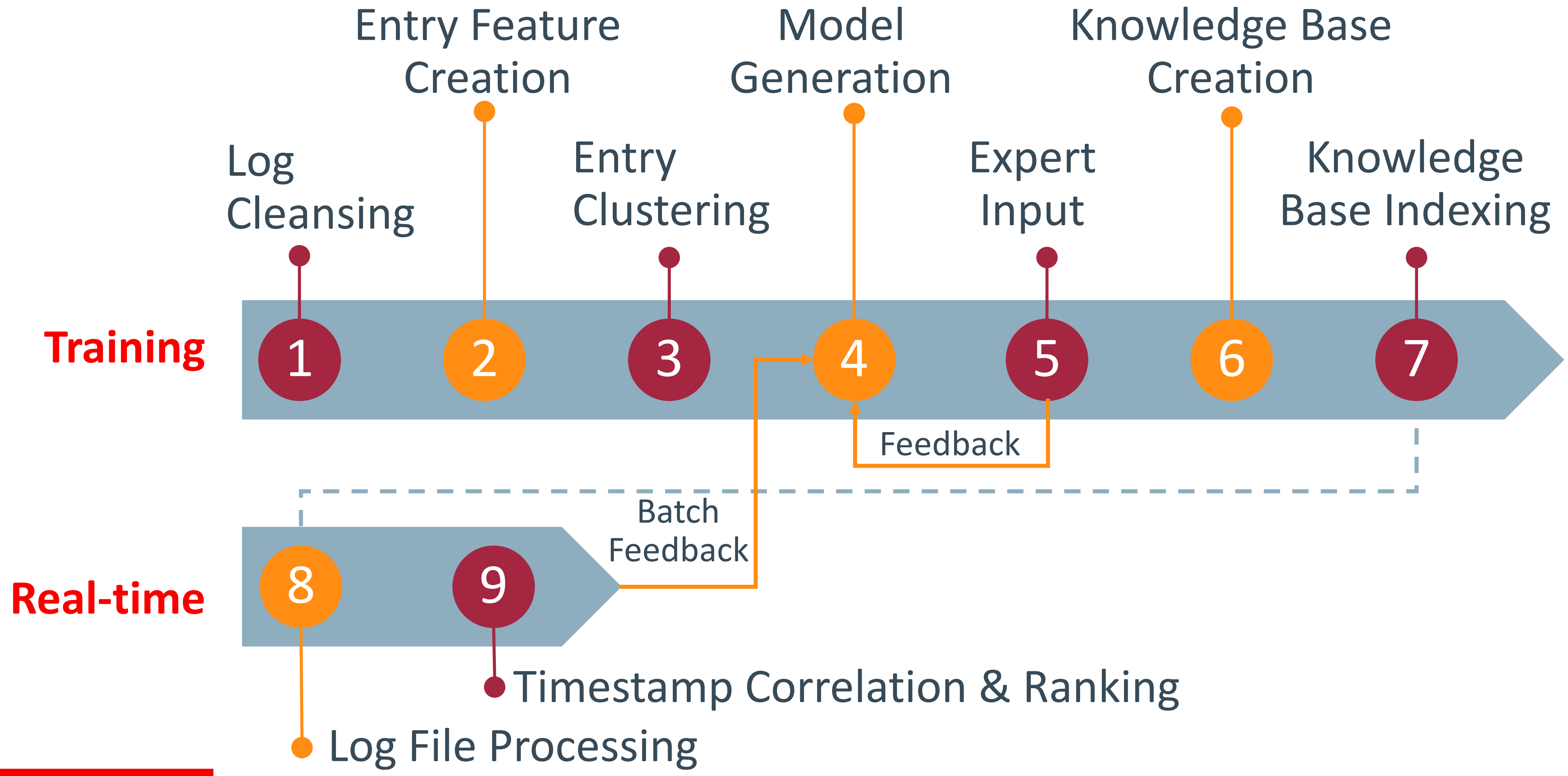
Remove clutter from log files to find the most important events to enable root cause analysis

ORACLE

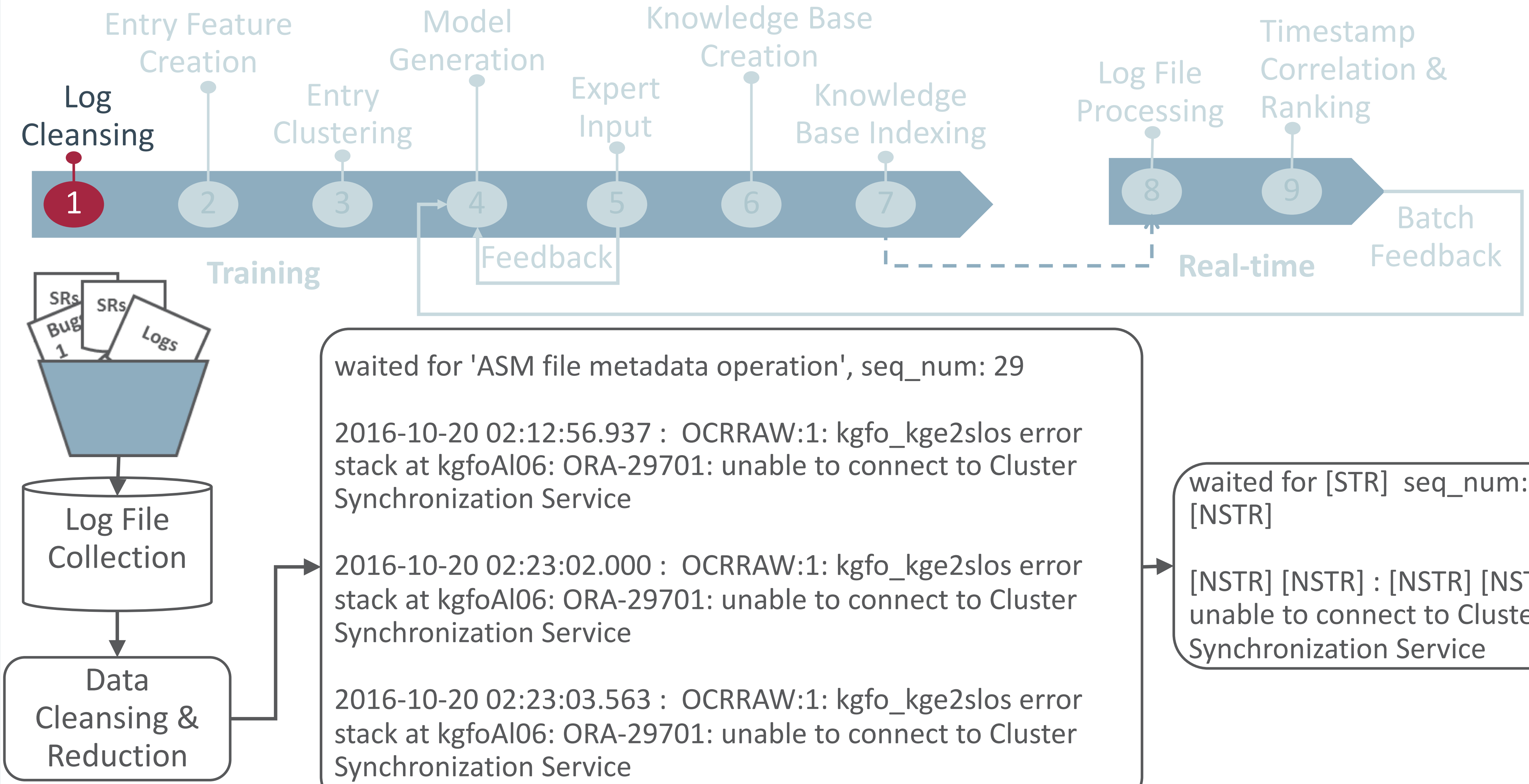
Anomaly Detection – High Level



Trace File Analyzer – High Level Anomaly Detection Flow



ORACLE



waited for 'ASM file metadata operation', seq_num: 29

2016-10-20 02:12:56.937 : OCRRAW:1: kgfo_kge2slos error stack at kgfoAl06: ORA-29701: unable to connect to Cluster Synchronization Service

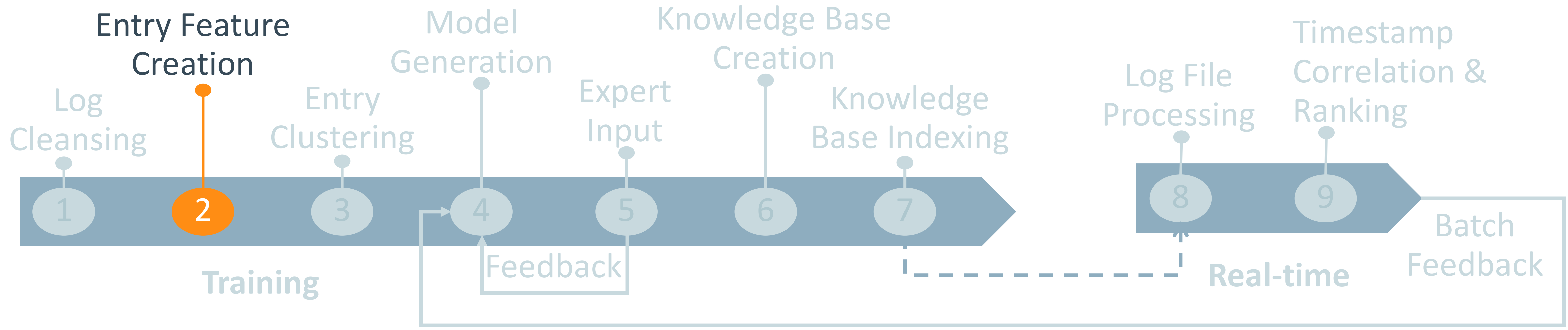
2016-10-20 02:23:02.000 : OCRRAW:1: kgfo_kge2slos error stack at kgfoAl06: ORA-29701: unable to connect to Cluster Synchronization Service

2016-10-20 02:23:03.563 : OCRRAW:1: kgfo_kge2slos error stack at kgfoAl06: ORA-29701: unable to connect to Cluster Synchronization Service

waited for [STR] seq_num: [NSTR]

[NSTR] [NSTR] : [NSTR] [NSTR] unable to connect to Cluster Synchronization Service

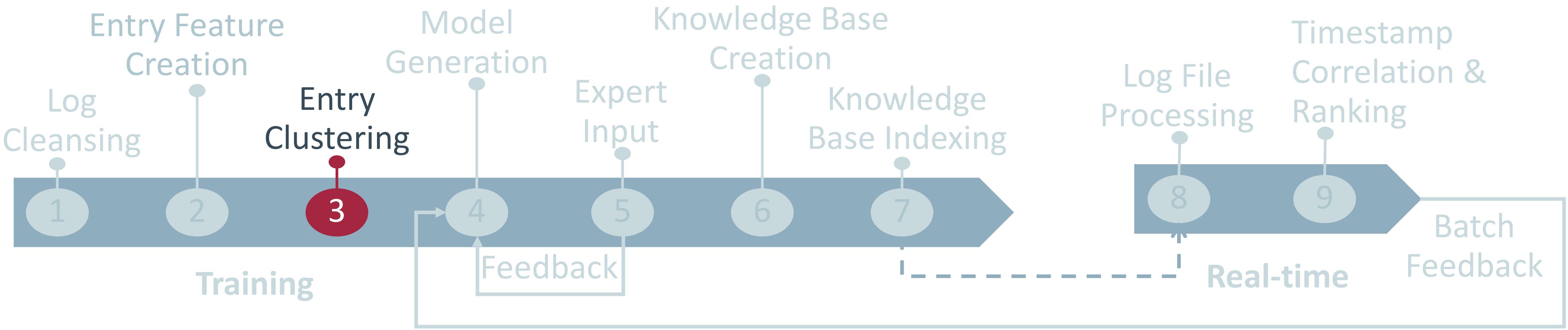
ORACLE



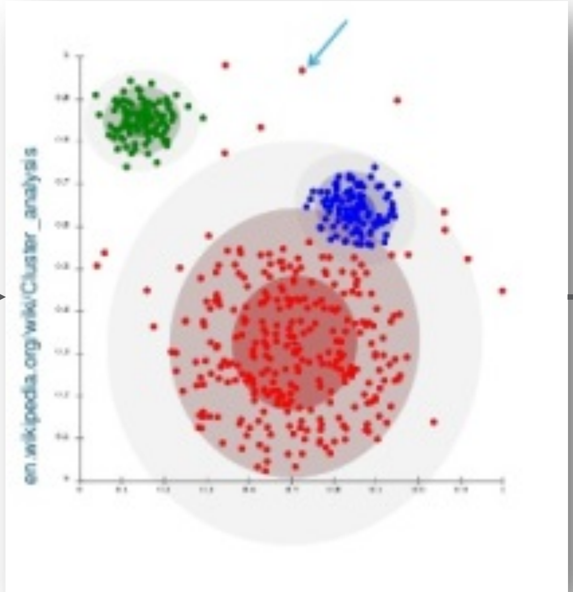
Feature Extraction → waited for [STR] seq_num: [NSTR]

[NSTR] [NSTR] : [NSTR] [NSTR] unable to connect to Cluster Synchronization Service →

..	Seen in Bugs	Total Bugs Seen	Seen in Files	Total Files Seen	Total Count	..
..	13	40	1440	5088	2890	..

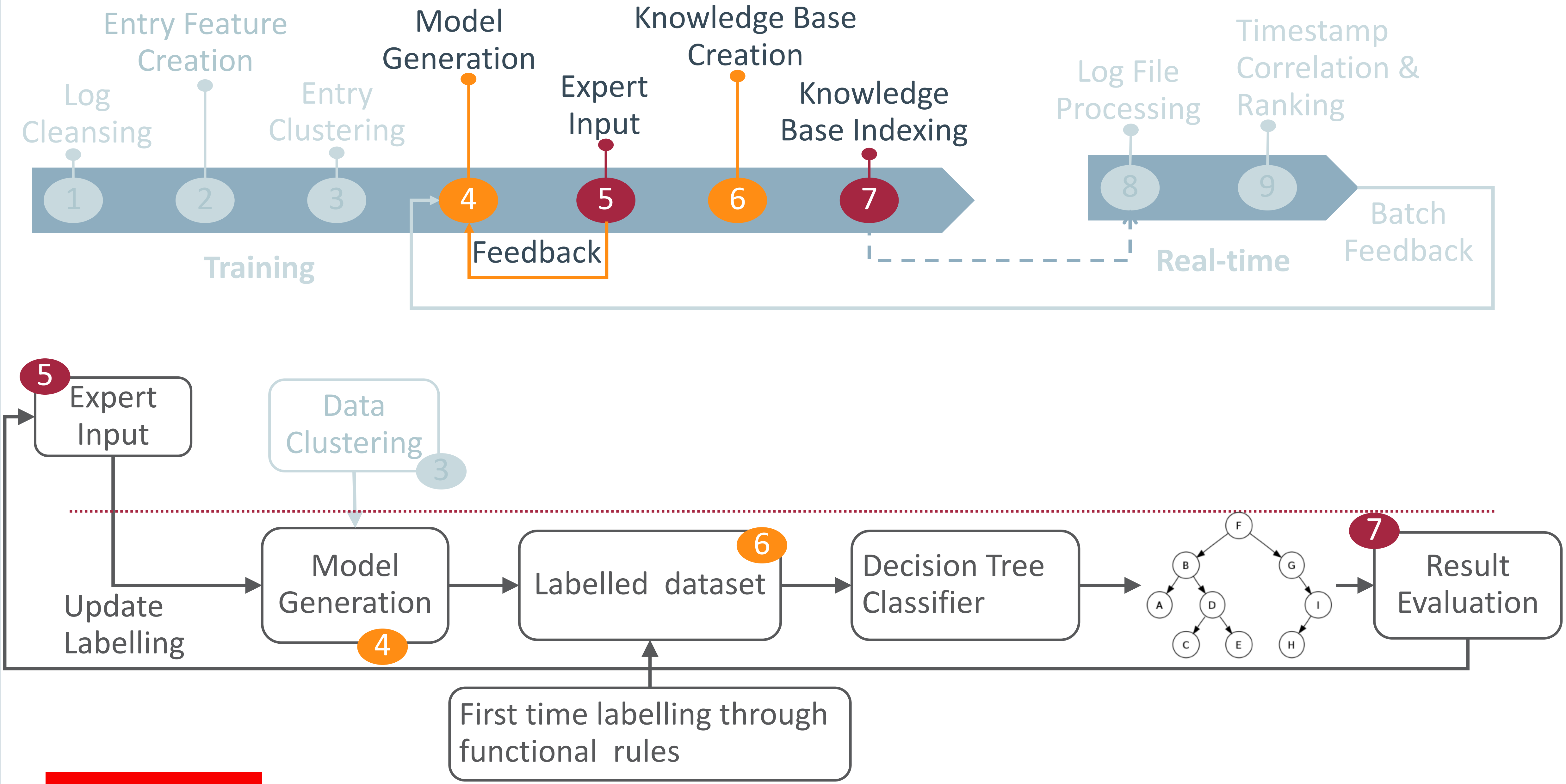


Data Clustering



Record Merging and feature aggregation for records belonging to same log signature

ORACLE


Entry Feature Creation
Model Generation
Knowledge Base Creation
Timestamp Correlation & Ranking
Log Cleansing
Entry Clustering
Expert Input
Knowledge Base Indexing
Log File Processing
1
2
3
4
5
6
7
8
9
Feedback
Training
Real-time
Batch Feedback
5
Expert Input
Data Clustering
3
Model Generation
4
Labelled dataset
6
Decision Tree Classifier
F
B
G
A
D
I
C
E
H
7
Result Evaluation
Update Labelling
First time labelling through functional rules

Autonomous Database Health - Maintenance Slot Identification

Find the next best window when maintenance can be performed with minimal service impact

Autonomous Database Health - Maintenance Slot Identification

Model Generation and Training Flow

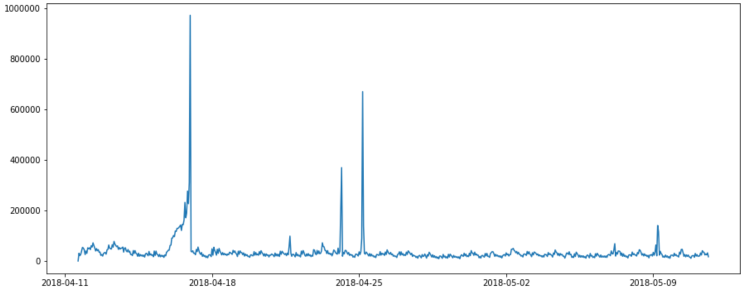
- Identify Relevant Workload Metrics
  - Ex: Average Active Sessions, CPU/Mem/IO Utilization
- Time Series Decomposition
  - Trend
  - Seasonality
  - Residual
- Workload Seasonality Determination Locating Minimas
- Optimum Window Identification and Validation

Autonomous Database Health - Maintenance Slot Identification

Seasonality Determination to Window Identification Flow

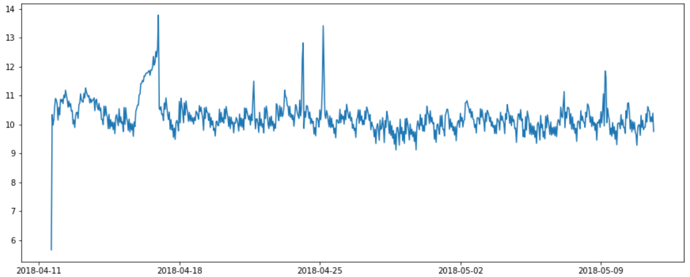
1 Original observation data

START_TIME	CNT
2018-04-11 15:00:00	290
2018-04-11 16:00:00	31120
2018-04-11 17:00:00	21530
2018-04-11 18:00:00	26240
2018-04-11 19:00:00	40520
2018-04-11 20:00:00	54270
2018-04-11 21:00:00	51460
2018-04-11 22:00:00	44310
2018-04-11 23:00:00	25690



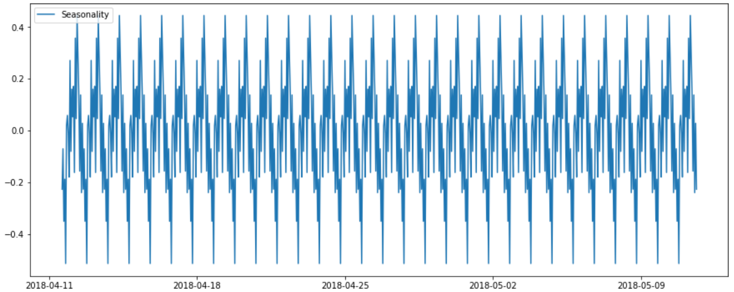
2 Apply convolution filter & average

START_TIME	
2018-04-11 15:00:00	5.669881
2018-04-11 16:00:00	10.345606
2018-04-11 17:00:00	9.977203
2018-04-11 18:00:00	10.175040
2018-04-11 19:00:00	10.609551
2018-04-11 20:00:00	10.901727
2018-04-11 21:00:00	10.848560
2018-04-11 22:00:00	10.698966
2018-04-11 23:00:00	10.153857



3 Calculate seasonality

START_TIME	
2018-04-11 15:00:00	-0.226098
2018-04-11 16:00:00	-0.069821
2018-04-11 17:00:00	-0.350088
2018-04-11 18:00:00	-0.187483
2018-04-11 19:00:00	-0.513240
2018-04-11 20:00:00	0.019737
2018-04-11 21:00:00	0.059213
2018-04-11 22:00:00	-0.011312
2018-04-11 23:00:00	-0.179156



4 Use seasonality to predict best maintenance window

```
Current Date : 2018-05-12 15:00:00
Current Position in Seasonality : -0.2260982974253585
Best Maintenance Period in next Cycle : 2018-05-12 19:00:00
Worst Maintenance Period in next Cycle : 2018-05-13 08:00:00
```

ORACLE

Autonomous Database Health - Maintenance Slot Identification

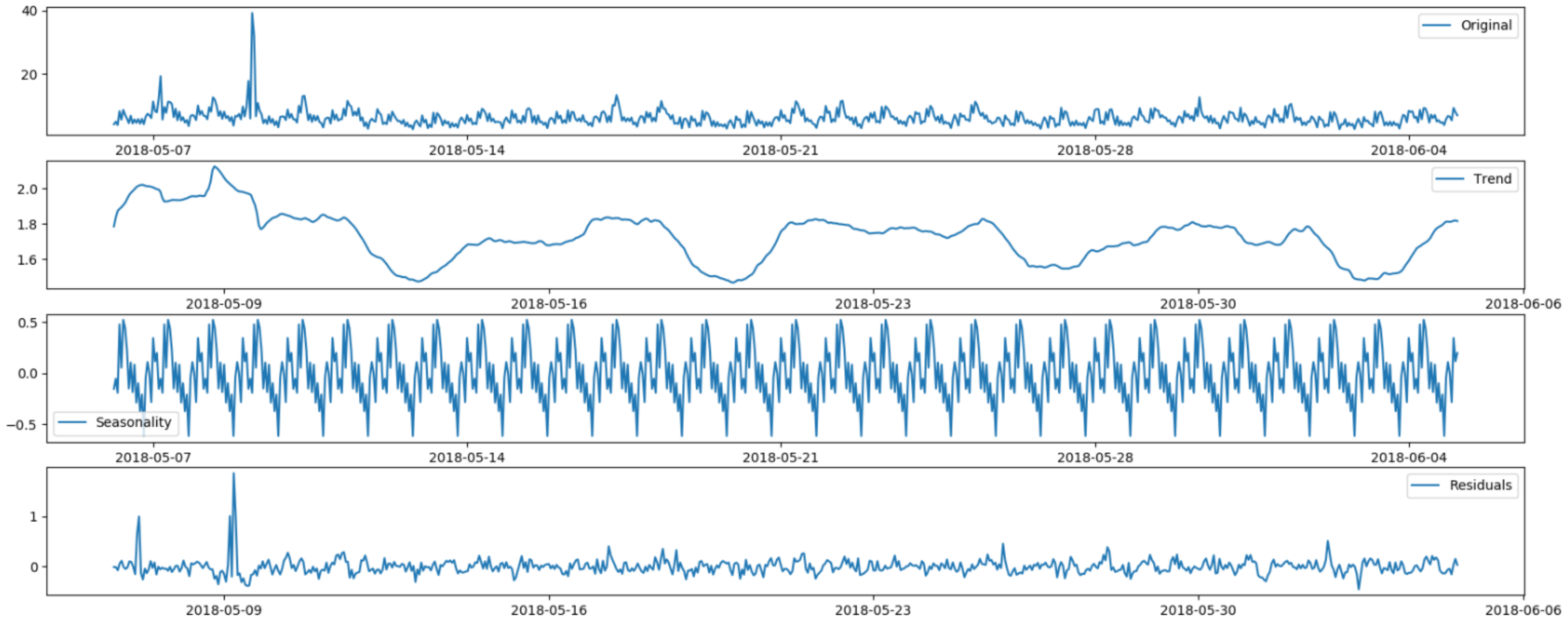
Validating Performance Against Random or Periodic Window Selection

Maintainance Window

Snapshot Time	Next Best Maintainance Window	#DataPoints (Total)	Down Time Window (TS Interval)	Detection Mechanism	Frequency	Transformation
2018-06-07 09:16:28.494471	2018-06-05 19:00:00	2171	60m	Seasonal Decomposition	24	Log

Seasonal Decomposition (Over Timeseries of Average Active Session)

Report for Historical Predictions - Precision - 83.3333% relative to Random Selection



Detect Metric Anomalies

Find combinations of unusual OS metrics to enable root cause analysis

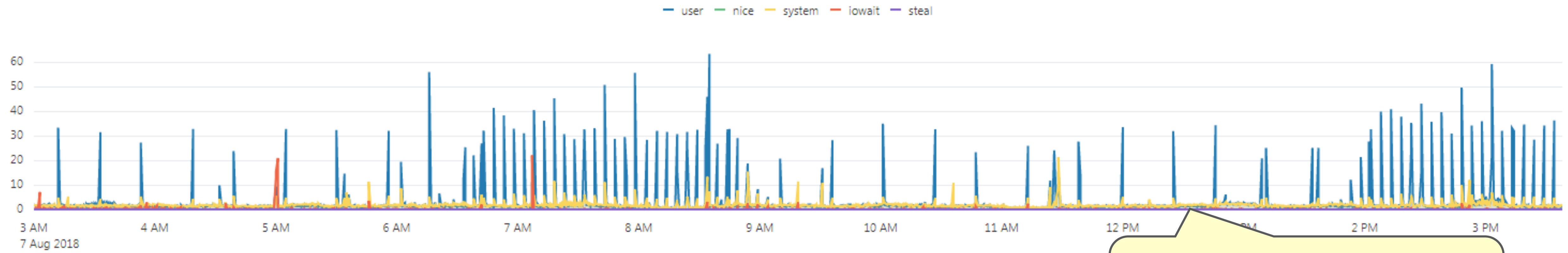
Use of Z-Score

- Z-Score gives us a measurement of standard deviation from the mean
- Allows us to compare the relative “unusualness” of different types of incomparable metrics like CPU usage vs IO waittime
- We multiple the Z-Score by a common factor, for ease of graphing and zooming

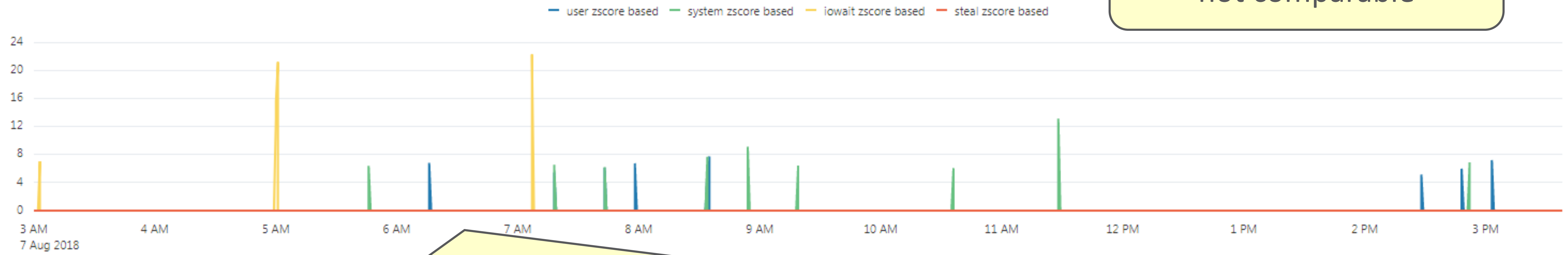
Dashboard SR Notes OS Analysis ✕

mysite.COM - CPU Utilization Distribution

user: Show the percentage of CPU utilization that occurred while executing at the user (application) level. nice: Show the percentage of CPU utilization that occurred while executing at the user level with nice priority. system: Show the percentage of CPU utilization that occurred while executing at the system (kernel) level. iowait: Show the percentage of time that the CPU or CPUs were idle during which the system had an outstanding disk I/O request. steal: Show the percentage of time spent in involuntary wait by the virtual CPU or CPUs while the hypervisor was servicing another virtual processor.



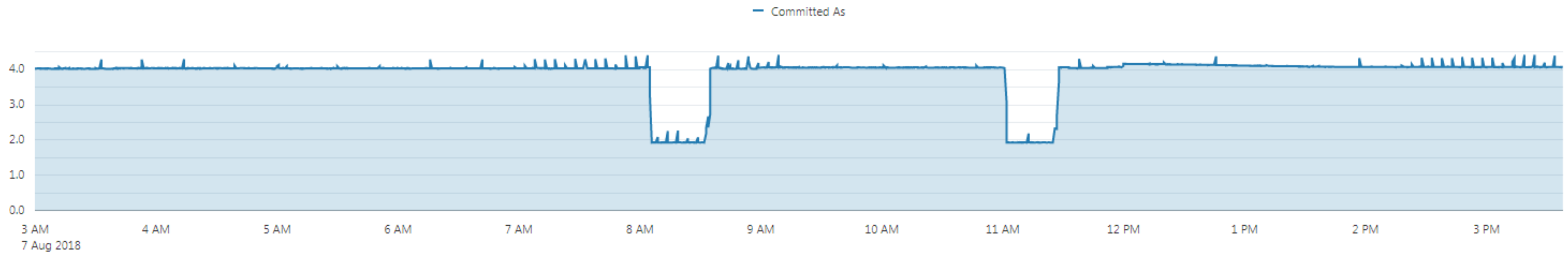
Original metric values are not comparable

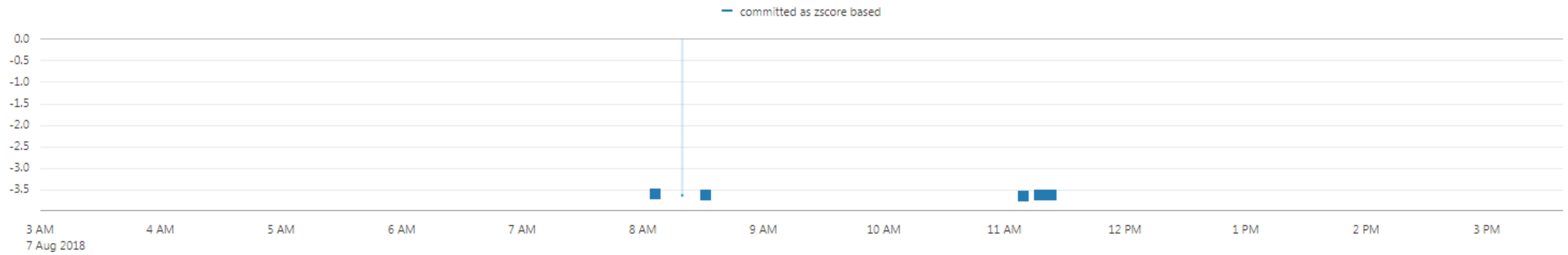


Z-Score factored values are now comparable
Larger spikes show more unusual values

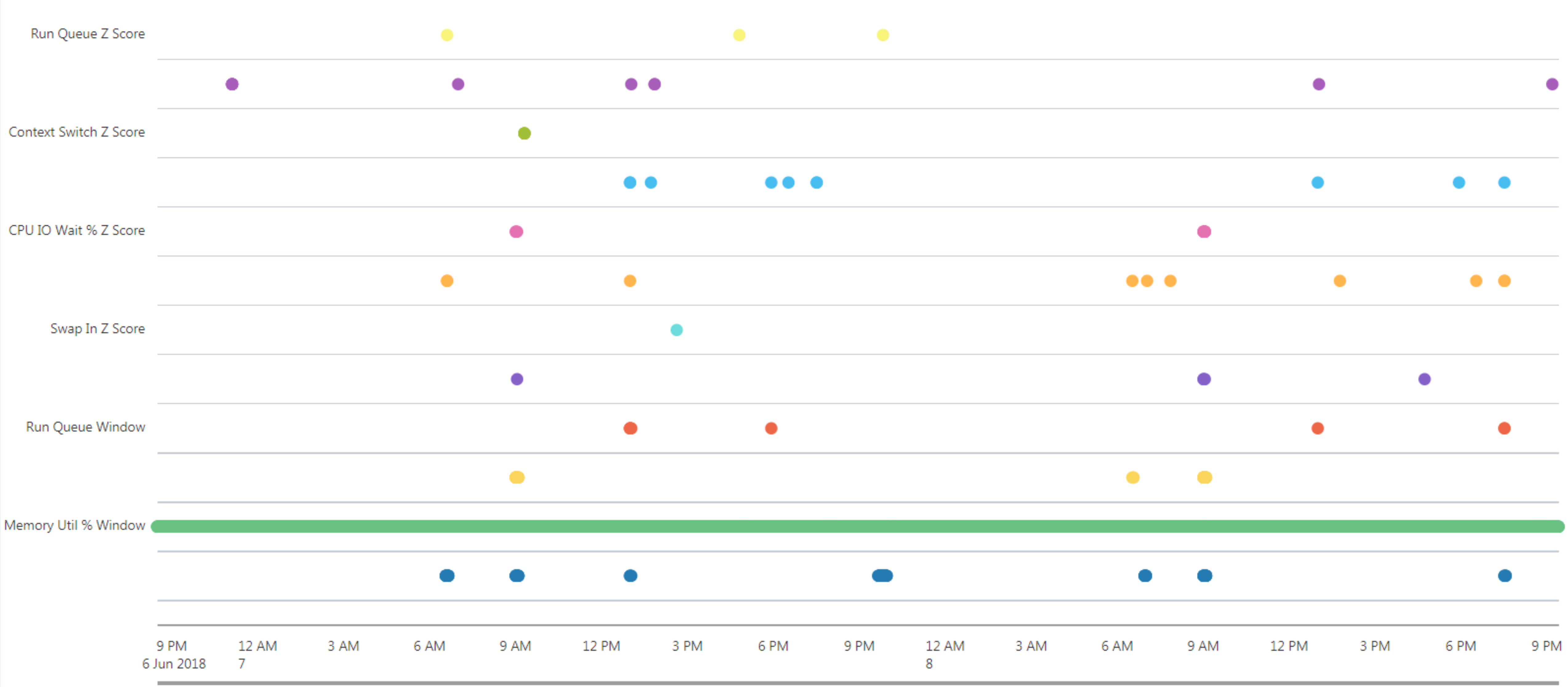
mysite.COM - Committed AS

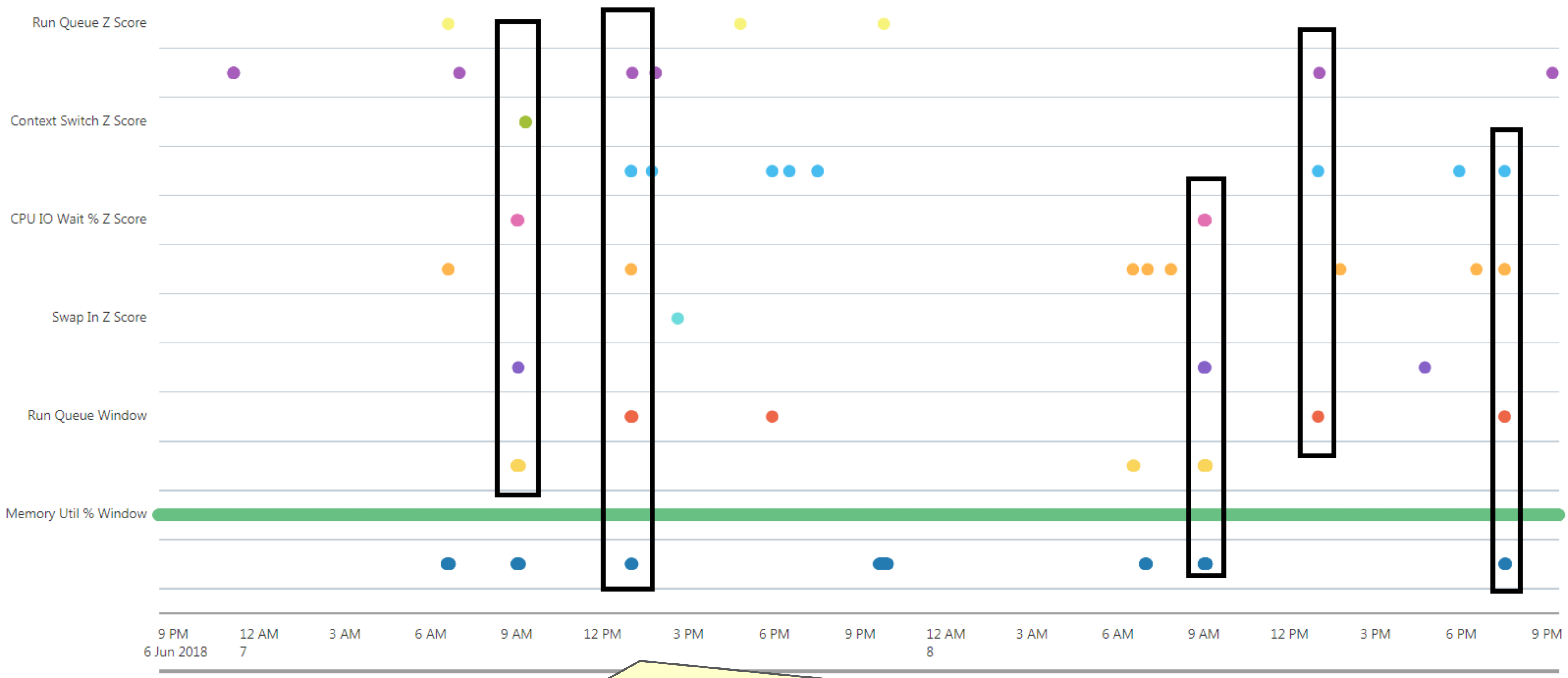
Committed Memory, is the sum of all memory which has been allocated by processes.






procs waiting for runtime
procs in uninterruptible sleep
0
30
60
90
120
150
3 PM
6 Jun 2018
9 PM
12 AM
7
3 AM
6 AM
9 AM
12 PM
6 PM
8
user
nice
iowait
0
10
20
30
40
50
60
70
procs waiting for runtime
procs in uninterruptible sleep
0
10
20
30
40

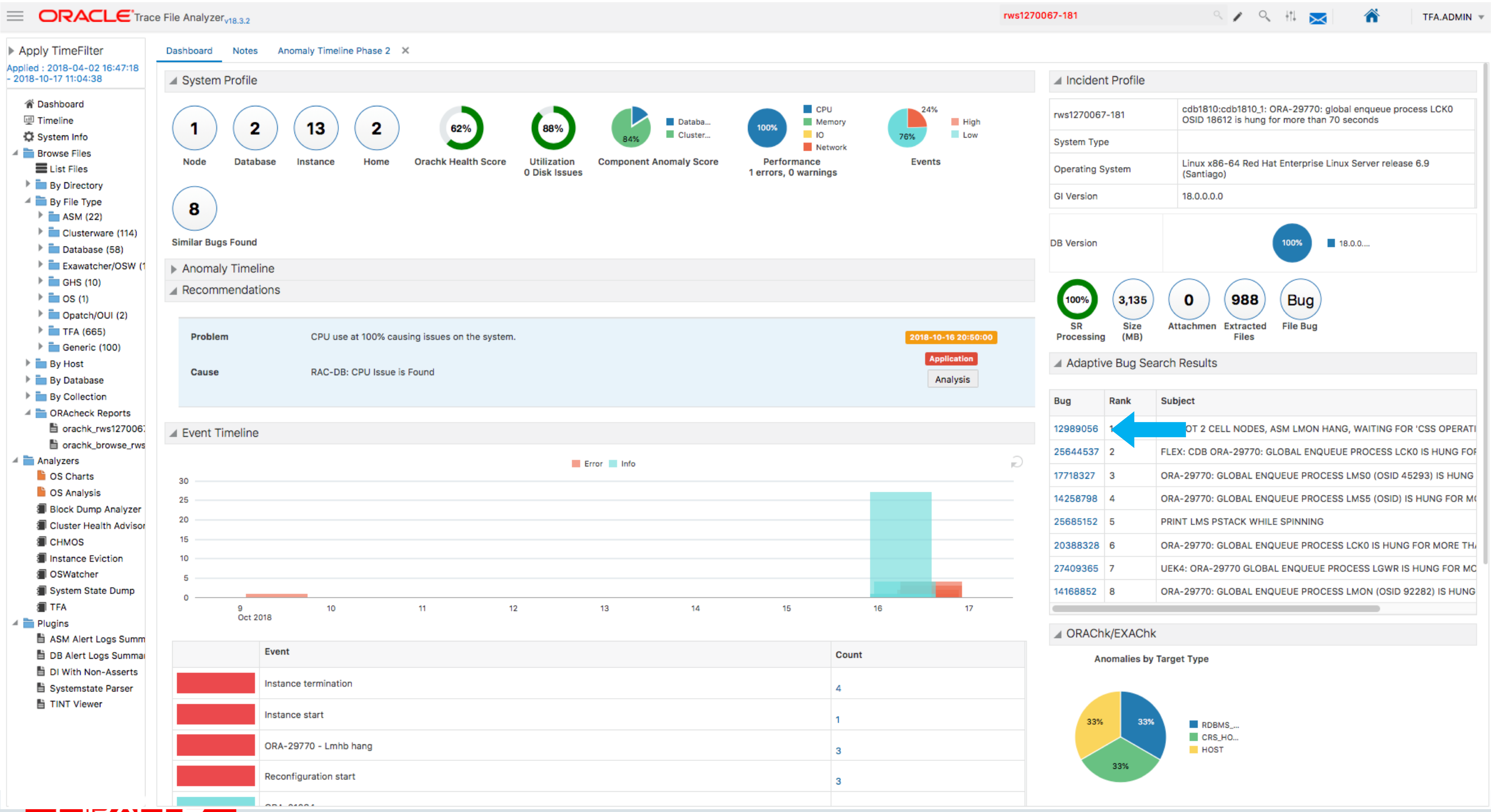

Run Queue Z Score
Context Switch Z Score
CPU IO Wait % Z Score
Swap In Z Score
Run Queue Window
Memory Util % Window
9 PM
6 Jun 2018
12 AM
7
3 AM
6 AM
9 AM
12 PM
3 PM
6 PM
9 PM
12 AM
8
3 AM
6 AM
9 AM
12 PM
3 PM
6 PM
9 PM



Identifying time periods with high z-score events across multiple metrics

Autonomous Health - Bug Duplicate Identification

Discovers Duplicate Bugs, Correlated Issues and Prioritizes Based Upon Customer Impact

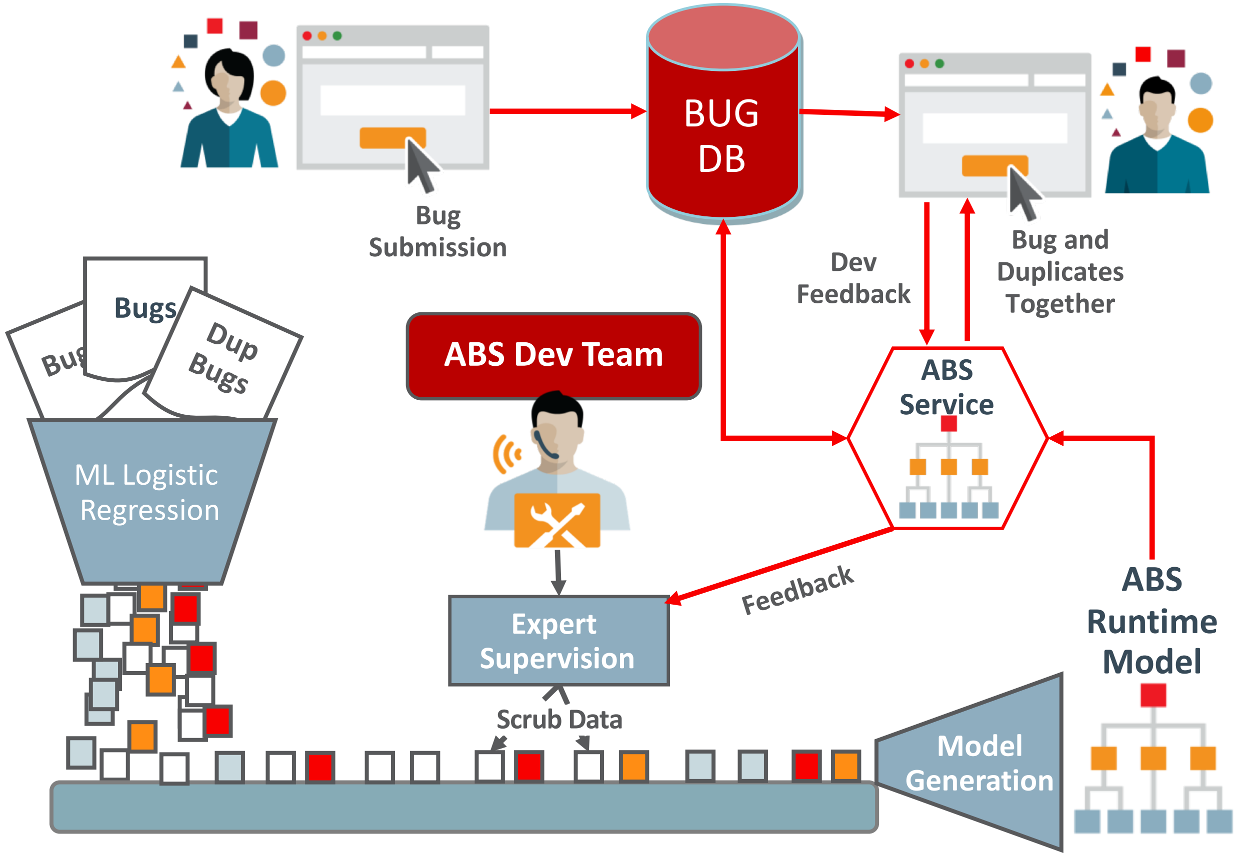


ORACLE Trace File Analyzer v18.3.2
rws1270067-181
TFA.ADMIN
Apply TimeFilter
Applied : 2018-04-02 16:47:18 - 2018-10-17 11:04:38
Dashboard
Timeline
System Info
Browse Files
List Files
By Directory
By File Type
ASM (22)
Clusterware (114)
Database (58)
GHS (10)
OS (1)
Opatch/OUI (2)
TFA (665)
Generic (100)
By Host
By Database
By Collection
ORAcheck Reports
Analyzers
OS Charts
OS Analysis
Block Dump Analyzer
CHMOS
Instance Eviction
OSWatcher
System State Dump
TFA
Plugins
DI With Non-Asserts
Systemstate Parser
TINT Viewer
Dashboard
Notes
Anomaly Timeline Phase 2
System Profile
1 Node
2 Database
13 Instance
2 Home
62% Orachk Health Score
88% Utilization 0 Disk Issues
84% Component Anomaly Score
100% Performance 1 errors, 0 warnings
CPU
Memory
IO
Network
76% 24% Events
High
Low
8 Similar Bugs Found
Anomaly Timeline
Recommendations
Problem CPU use at 100% causing issues on the system.
Cause RAC-DB: CPU Issue is Found
2018-10-16 20:50:00
Application
Analysis
Event Timeline
Error
Info
Oct 2018
Event
Count
Instance termination 4
Instance start 1
ORA-29770 - Lmhb hang 3
Reconfiguration start 3
Incident Profile
rws1270067-181
cdb1810:cdb1810_1: ORA-29770: global enqueue process LCK0 OSID 18612 is hung for more than 70 seconds
System Type
Operating System
Linux x86-64 Red Hat Enterprise Linux Server release 6.9 (Santiago)
GI Version 18.0.0.0.0
DB Version
100% SR Processing
3,135 Size (MB)
0 Attachmen
988 Extracted Files
Bug File Bug
Adaptive Bug Search Results
Bug
Rank
Subject
12989056 1
25644537 2 FLEX: CDB ORA-29770: GLOBAL ENQUEUE PROCESS LCK0 IS HUNG FOR
17718327 3 ORA-29770: GLOBAL ENQUEUE PROCESS LMS0 (OSID 45293) IS HUNG
14258798 4
25685152 5 PRINT LMS PSTACK WHILE SPINNING
20388328 6
27409365 7
14168852 8 ORA-29770: GLOBAL ENQUEUE PROCESS LMON (OSID 92282) IS HUNG
ORAChk/EXAChk
Anomalies by Target Type
33%
33%
33%
HOST

Adaptive Bug Search – Applied Machine Learning

Discovers Duplicate Bugs and Correlated Issues

- Bugs are submitted from over 400 Oracle products
- Performs ML Logistic Regression on training set of bugs to generate model
- Displays up to 8 possible duplicates per bug or SR
- Feedback improves model accuracy
  - Direct from developers
  - Indirect from bug updates



Autonomous Database Health – Adaptive Bug Search (ABS)

High Level Flow

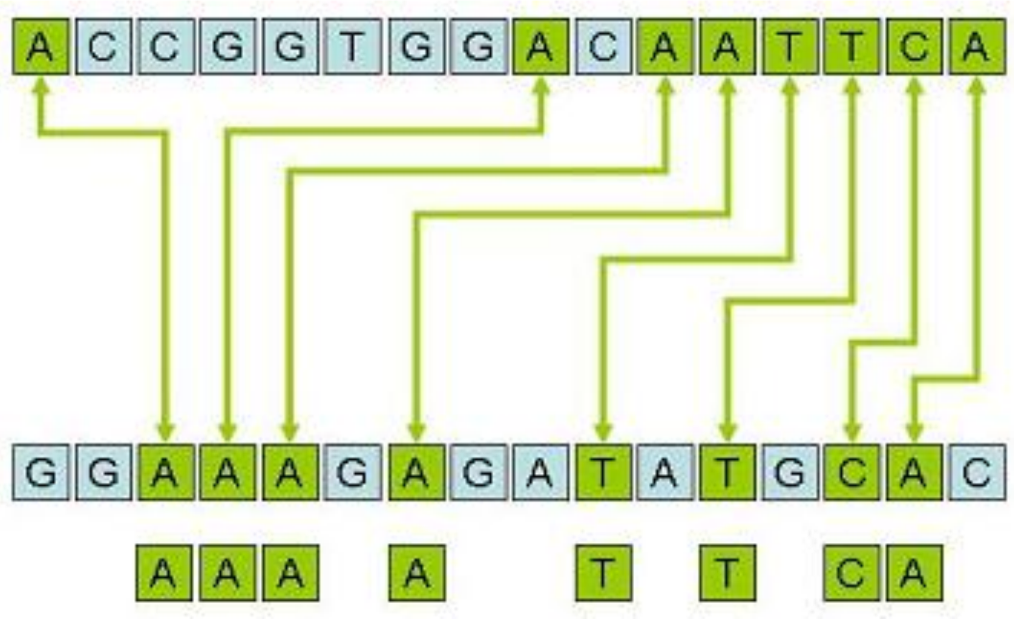
- Issues parsed into different features
  - Error stack, Trace data, Problem description, etc.
- Issues represented as a cluster of features
  - i.e. All bugs in a bug tree contribute towards the feature set
- Logistic Regression applied to build a model
  - Model defines the significance of each feature
- Similarity between issues computed using the model
  - Identifies the root of the cluster (aka bug tree)
- Feedback used to improve the model
  - Feedback is automatically derived based on how the bug gets closed

Autonomous Database Health - Anomaly Analysis

Identify a series of events as connected and representing the signature of a problem

Longest Common Subsequence of Anomalous Entries

1. Start by classifying a problem such as an important ORA or CRS error
2. Find occurrences of the problem across many different log files
3. Identify anomalous entries and lifecycle events in chronological order within a predefined time window around the occurrence of the problem in all the logs
4. Compare the repeating anomalous / lifecycle entries to identify the longest common subsequence of anomalous entries



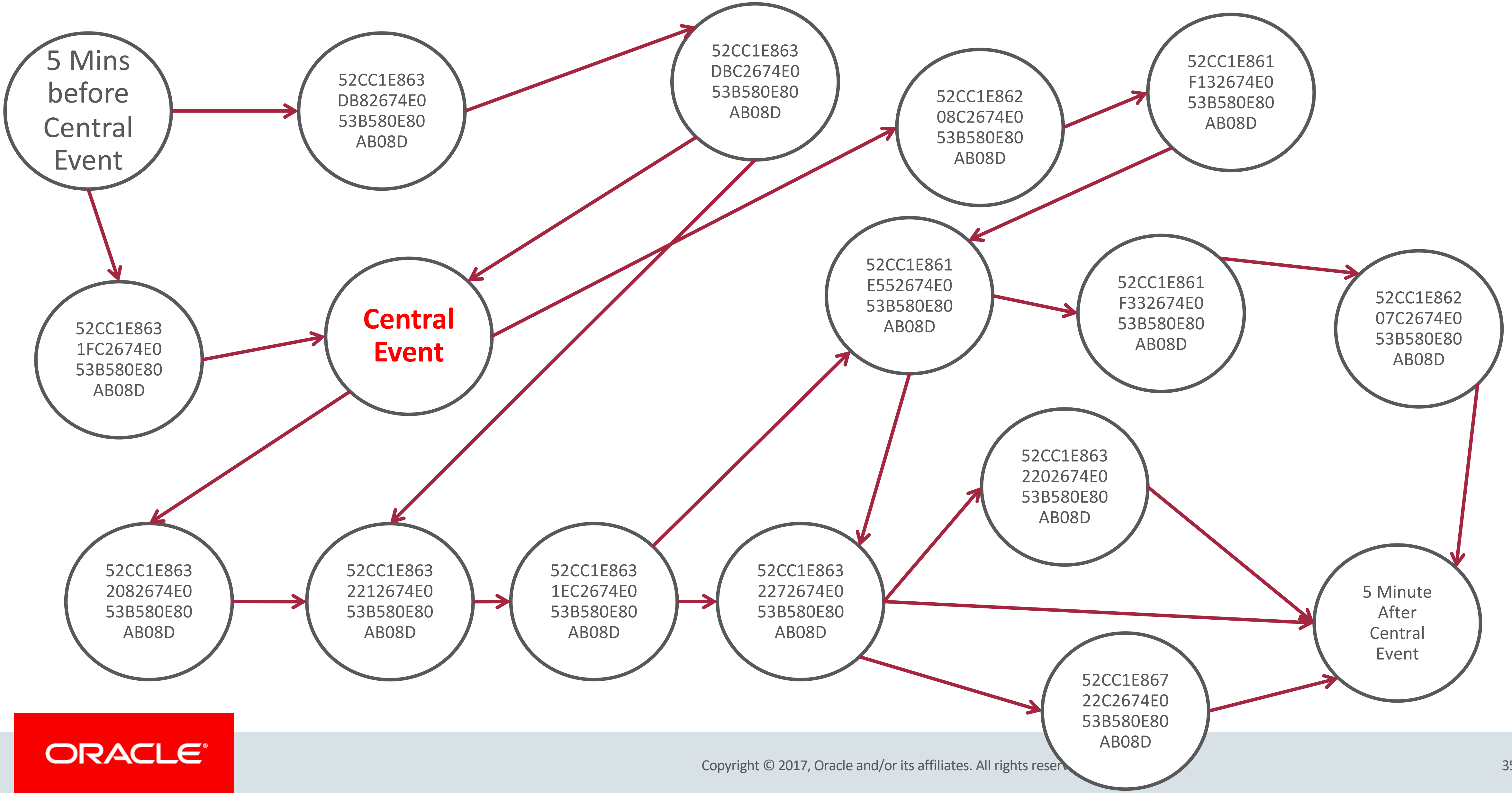
Find the Finite State Automata(FSA)

Example signatures and their analysis

***Sample Central Event** : 2017-01-19 16:51:20.562 [OCSSD(24862)]**CRS-1656**: The CSS daemon is terminating due to a fatal error; Details at (:CSSSC00012:) in /tools/list/grid/orabase/diag/crs/ur102ora3502c/crs/trace/ocssd.trc*

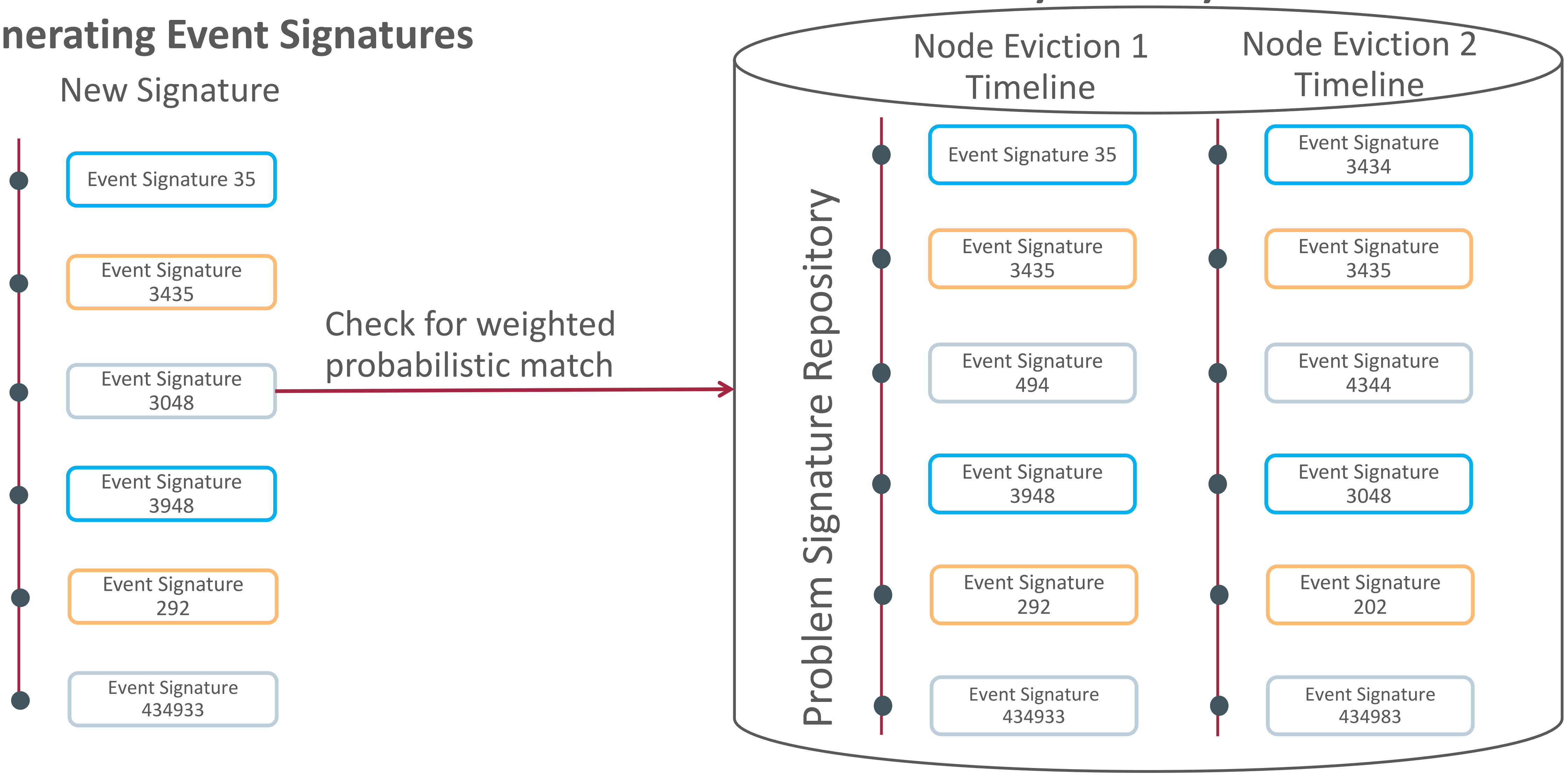
Knowledge Id	***Sample Line (States in FSA for central event)***
52CC1E8631FC2674E053B580E80AB08D	2016-10-16 21:22:36.520+CRS-5008: Invalid attribute value: en4 for the network interface
52CC1E8632082674E053B580E80AB08D	2016-10-16 21:25:11.516 [OCSSD(6816354)]CRS-1608: This node was evicted by node 3, rwsbs03; details at (:CSSNM00005:) in /u01/app/crsusr/diag/crs/rwsbs02/crs/trace/ocssd.trc.
52CC1E8632212674E053B580E80AB08D	2016-10-16 21:25:17.927 [OCSSD(18219406)]CRS-1654: Clean up of CRSD resources finished successfully.
52CC1E8631EC2674E053B580E80AB08D	2016-10-16 21:25:17.927 [OCSSD(18219406)]CRS-1655: CSSD on node rwsbs01 detected a problem and started to shutdown.
52CC1E8632272674E053B580E80AB08D	2016-10-16 21:25:19.431 [OCSSD(18219406)]CRS-8503: Oracle Clusterware process OCSSD with operating system process ID 18219406 experienced fatal signal or exception code 6.
52CC1E8632202674E053B580E80AB08D	2016-10-16 21:25:21.788 [CRSD(44696012)]CRS-0805: Cluster Ready Service aborted due to failure to communicate with Cluster Synchronization Service with error [3]. Details at (:CRSD00109:) in /u01/app/crsusr/diag/crs/rwsbs01/crs/trace/crsd.trc.
52CC1E86208C2674E053B580E80AB08D	2016-10-18 02:02:00.835 : CSSD:6684: (:CSSSC00012:)clssscExit: A fatal error occurred and the CSS daemon is terminating abnormally
52CC1E861F132674E053B580E80AB08D	CLSB:6684: Oracle Clusterware infrastructure error in OCSSD (OS PID 12452524): Fatal signal 6 has occurred in program ocssd thread 6684; nested signal count is 1
52CC1E861E552674E053B580E80AB08D	Incident 393 created, dump file: /u01/app/crsusr/diag/crs/rwsbs02/crs/incident/incdir_393/ocssd_i393.trc
52CC1E861F332674E053B580E80AB08D	2016-10-18 02:02:07.113 : SKGFD:5655: ERROR: -9(Error 27041, OS Error (IBM AIX RISC System/6000 Error: 47: Write-protected media
52CC1E86207C2674E053B580E80AB08D	2016-10-18 02:02:07.774 : CSSD:5655: clssnmvDiskCreate: Cluster guid ea34893b9442ef79ff642d70699aff9d found in voting disk /dev/rbs01_100G_asm1 does not match with the cluster guid 7b63590c34fa5f44bf6944aefa4ee85d obtained from the GPnP profile
52CC1E863DB82674E053B580E80AB08D	2017-01-19 16:48:01.057 [OCSSD(24862)]CRS-1649: An I/O error occurred for voting file: /dev/rdsk/c1d16; details at (:CSSNM00059:) in /tools/list/grid/orabase/diag/crs/ur102ora3502c/crs/trace/ocssd.trc.
52CC1E863DBC2674E053B580E80AB08D	2017-01-19 16:49:40.550 [OCSSD(24862)]CRS-1615: No I/O has completed after 50% of the maximum interval. Voting file /dev/rdsk/c1d16 will be considered not functional in 99508 milliseconds

Example signatures and their analysis



Autonomous Database Health - Anomaly Analysis

Generating Event Signatures



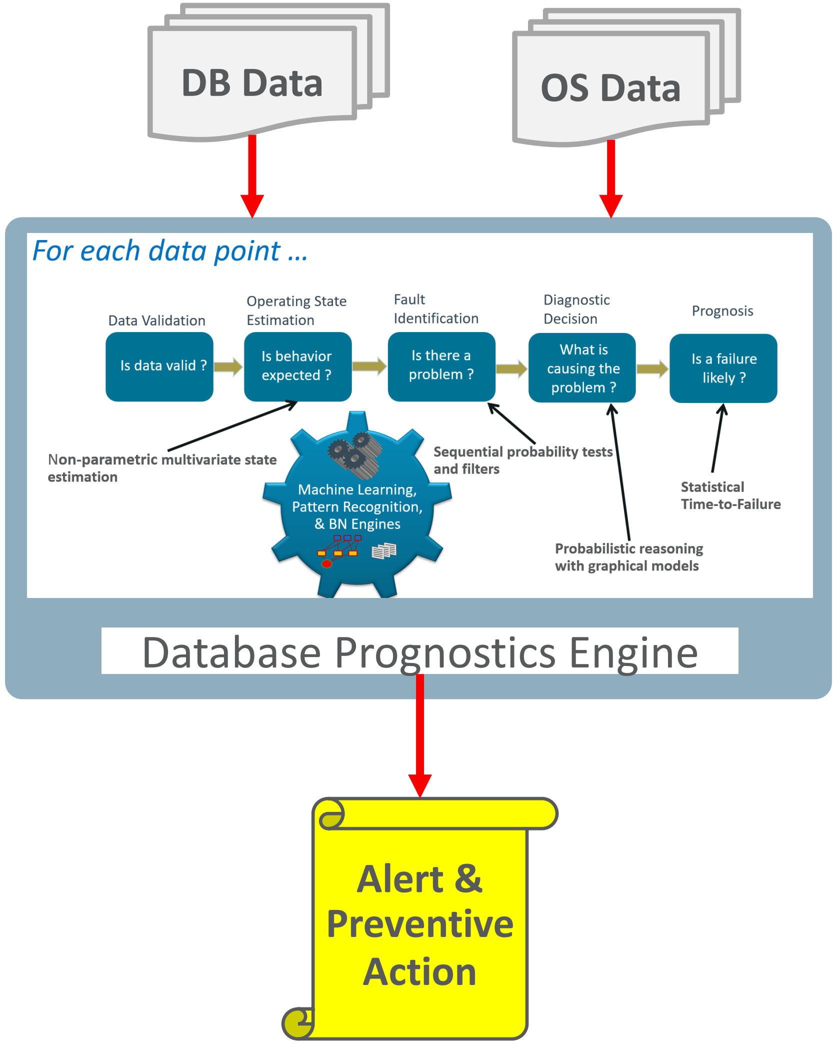
Autonomous Database Health - Database Performance

Preserving instance performance when database resources are constrained

Autonomous Database Health – Database Performance

Database Data Flow Overview

- Reads OS and DB Performance data directly from memory
- Uses Machine Learning models and data to perform prognostics
- Detects common RAC database problems
- Performs root cause analysis
- Sends alerts and preventative actions to Cloud Ops per target



Autonomous Database Health – Database Performance

Data Sources and Data Points

A *Data Point* contains > 150 signals (statistics and events) from *multiple sources*

OS, ASM , Network → ← **DB (SH, AWR session, system and PDB statistics)**

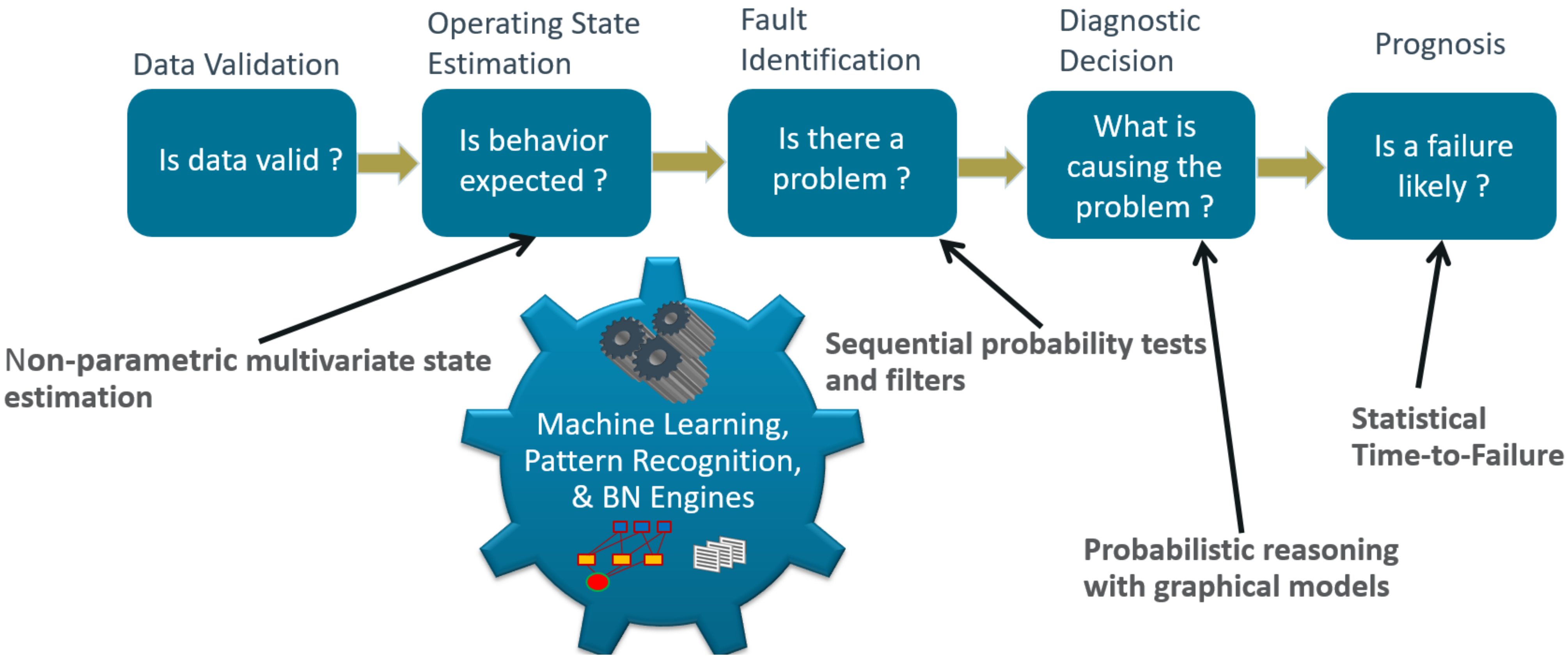
Time	CPU	ASM IOPS	Network % util	Network_ Packets Dropped	Log file sync	Log file parallel write	GC CR request	GC current request	GC current block 2-way	GC current block busy	Enq: CF - contention	…
15:16:00	0.90	4100	13%	0	2 ms	600 us	0	0	300 us	1.5 ms	0	

Statistics are collected at a ***1 second internal sampling*** rate , synchronized, smoothed and aggregated to a Data Point ***every 5 seconds***

Autonomous Database Health – Database Performance

Data Flow Overview

For each data point …



Autonomous Database Health – Database Performance

Inline and Immediate Fault Detection and Diagnostic Inference

Input : Data Point at Time *t*

Time	CPU	ASM IOPS	Network % util	Network_ Packets Dropped	Log file sync	Log file parallel write	GC CR request	GC current request	GC current block 2-way	GC current block busy	Enq: CF - contention	…
15:16:00	0.90	4100	88%	105	2 ms	600 us	504 ms	513 ms	2 ms	5.9 ms	0	

Fault Detection and Classification

15:16:00	OK	OK	HIGH 1	HIGH 2	OK	OK	HIGH 3	HIGH 3	HIGH 4	HIGH 4	OK	

Diagnostic Inference

Symptoms

15:16:00

1. Network Bandwidth Utilization
2. Network Packet Loss
3. Global Cache Requests Incomplete
4. Global Cache Message Latency

→ **Diagnostic Inference Engine** →

Root Cause
(Target of Corrective Action)

Network Bandwidth Utilization

ORACLE

Autonomous Database Health Platform ML Technologies

Real-time Prevention

- Data Ingestion
  - Kernel Smoothing and Moving Average
  - Interpolation and Imputation
- Prediction and Pattern Recognition
  - Multivariate and Auto-Associative Regression
  - Clustering, Similarity Operators and Bayes Networks
- Fault and Anomaly Detection
  - Sequential Probability Ratio Tests
  - Conditional Probability Filters & Hidden Markov Models
- Prognosis and Diagnosis
  - Bayesian Belief Networks and Probabilistic Inference
  - Remaining Useful Life Regression and GPM Models

Rapid Recovery

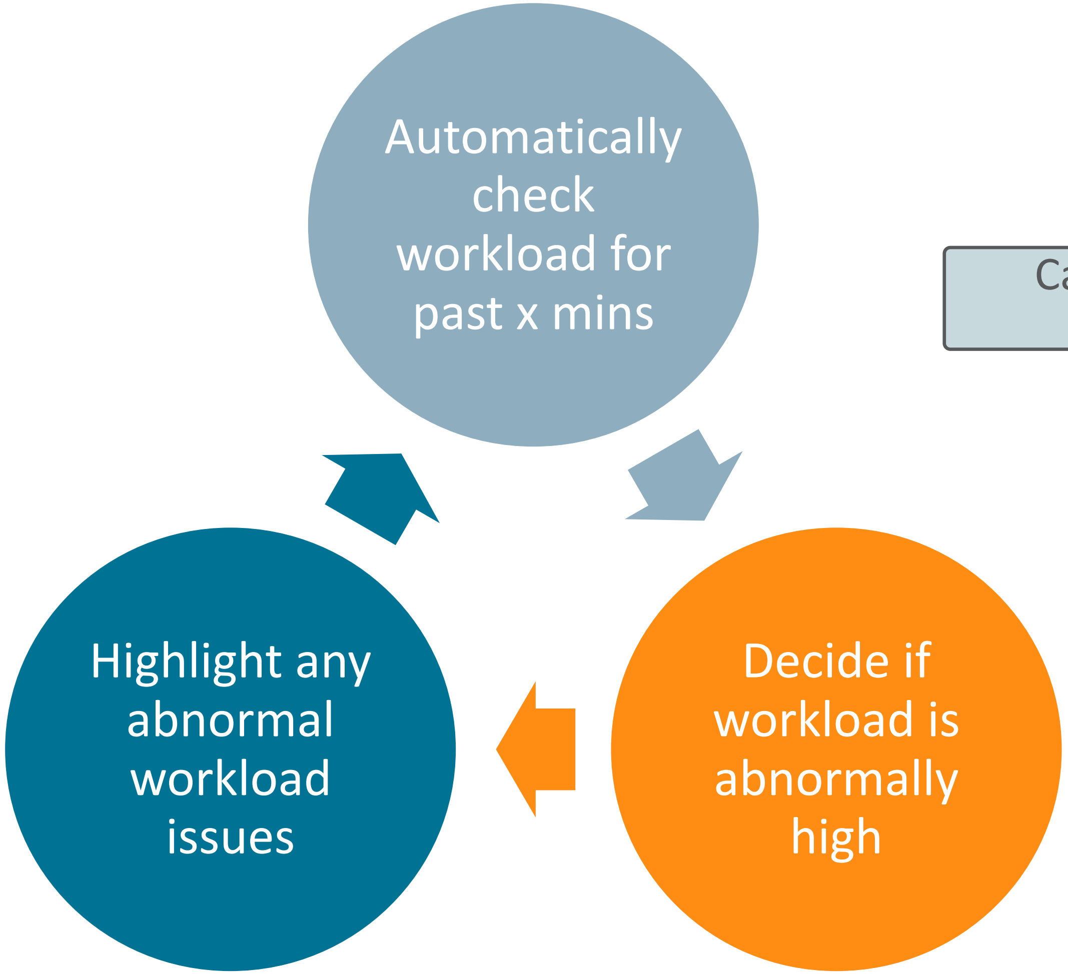
- Data Ingestion
  - ELK
  - Lucene
- Prediction and Pattern Recognition
  - TF-IDF and Bag-of-Words modelling
  - Sequence Matcher
  - K-nearest Neighbour
- Fault and Anomaly Detection
  - Decision Trees and Random Forest
  - Sequential Pattern Mining
- Prognosis and Diagnosis
  - Recurrent neural Network
  - Long short-term memory Predictive Analysis

ORACLE

Autonomous Database Health - Database Performance

Workload Determination and deviation and when to scale the load or look for problems

What is Workload



Calculated via machine learning

Optionally run on demand

Optionally snooze checking of a component

ORACLE

Workload Process

Adaptive Learning

1 CAPTURE

Every 5 Mins

Captures metrics for key performance dimensions across 5 X 1 minute time windows

2 TRAIN

Every Week

Using semi-supervised learning via SME threshold rules, the following models are retrained :

- Isolation Forest
- One-Class Support Vector Machine
- Local Outlier Factor

Model with highest confidence becomes the primary, if confidence is high enough

3 PREDICT

Every 5 Mins

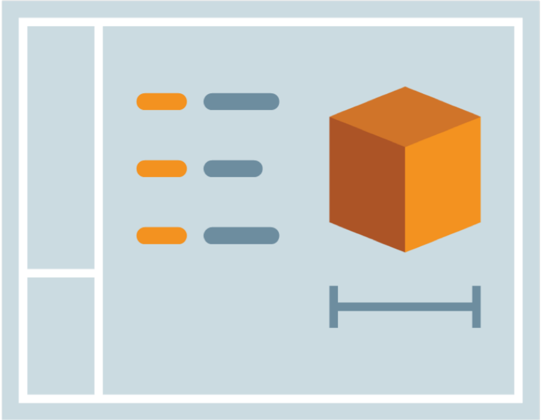
Straight after capture, the primary model is used to predict anomalies.

Where anomalies are identified, metrics are compared to SME threshold rules to identify the type of anomaly

ORACLE

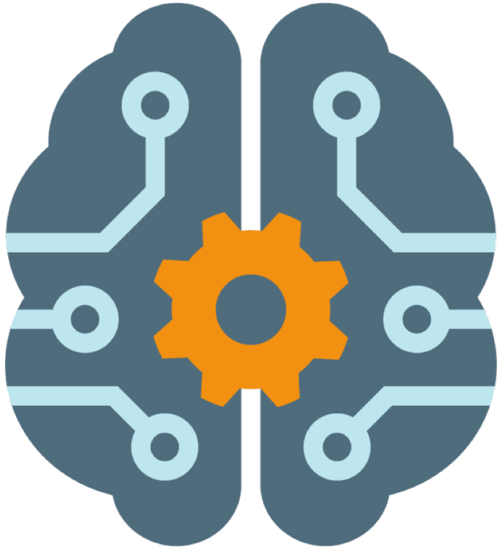
Capture

- Initial one-time setup defines configuration for scope of CDBs, PDBs & Services
- Every 5 minutes capture metrics for key performance dimensions:
- Other performance related dimensions can be used in the future
- Capture gets ASH data for later analysis



Train

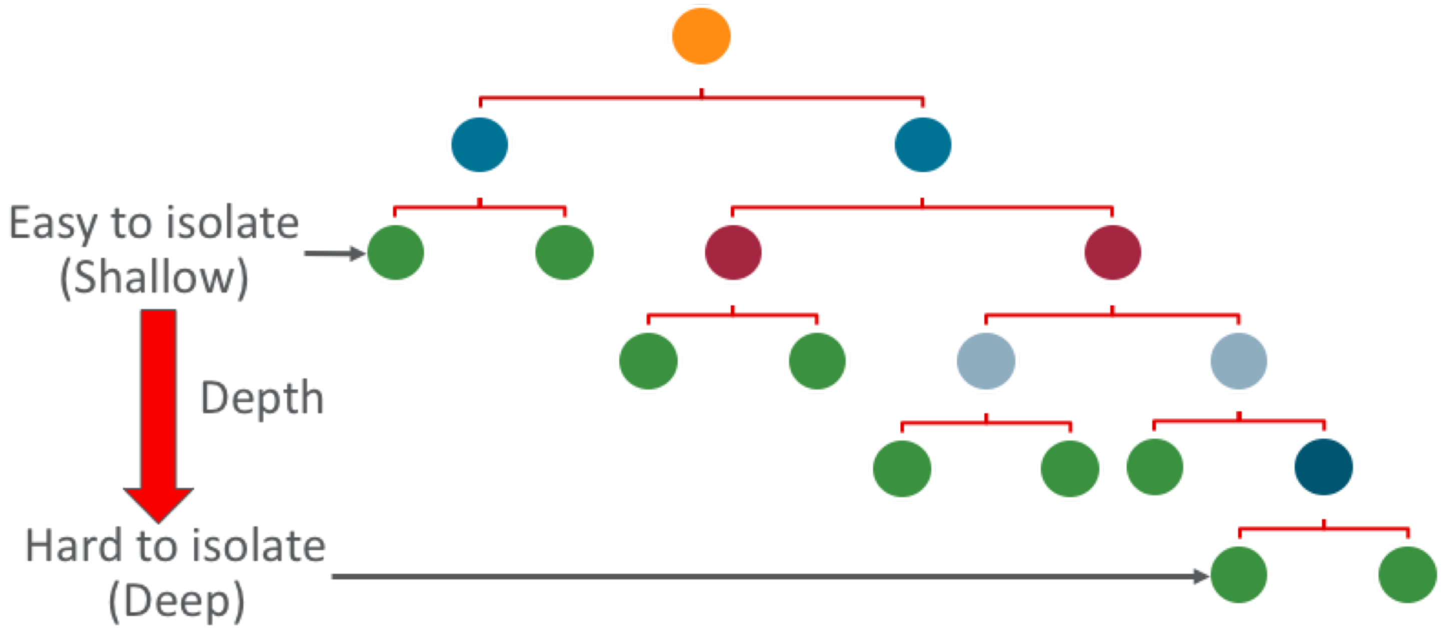
- The following models are retrained to identify anomalies in the metrics
  1. Isolation Forest
  2. One-Class Support Vector Machine
  3. Local Outlier Factor
- Each model is evaluated using 5 test accuracy scores
- Model with the highest confidence becomes the primary and is used for prediction until next training iteration, as long as confidence is > 95%
- Testing has shown minimum of 7 days data collection is required
- Maintain a rolling window of 30 days of data to account for seasonality within a month & provide better predictability



ORACLE

Isolation Forest Overview

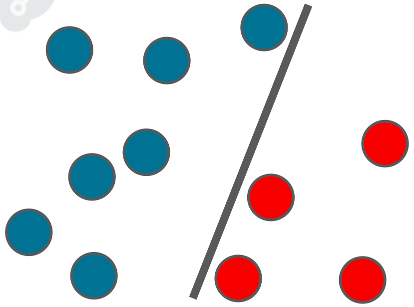
- Used to explicitly identify outliers (anomalies) rather than profiling normal data points
- Outliers are less frequent than regular observations
- Outliers lie further away from the regular observations
- Randomly separated decision trees are used because outliers will be found by identifying observations closer to the root of the tree with fewer splits



ORACLE

One-Class Support Vector Machine

1

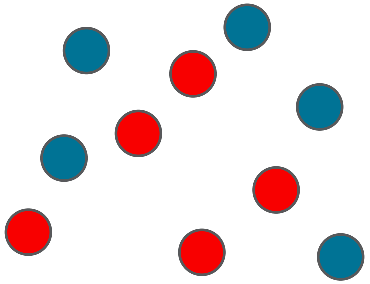


Learn to classify observations as similar or different to a training set

Define a straight line (hyperplane) for data-point classification

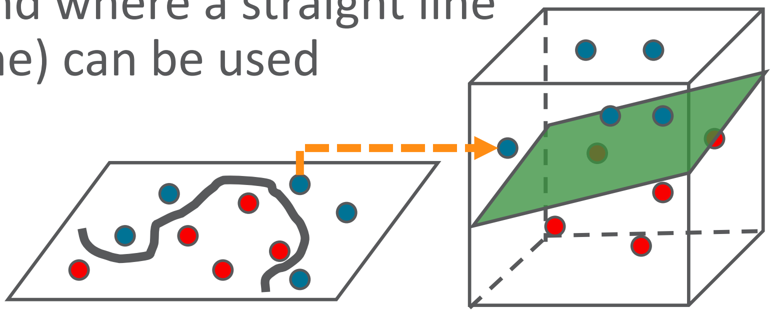
2

Sometimes a straight line is not possible with the current dimensions



3

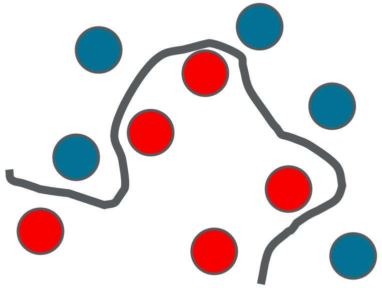
Include another dimension (kernel) our data uses Radial Basis Function (RBF) to find where a straight line (hyperplane) can be used



2 dimensions 3 dimensions

4

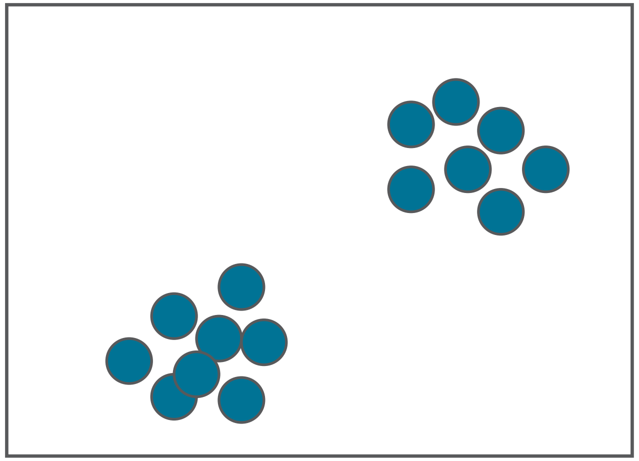
Data-points can now be classified



One-Class Support Vector Machine

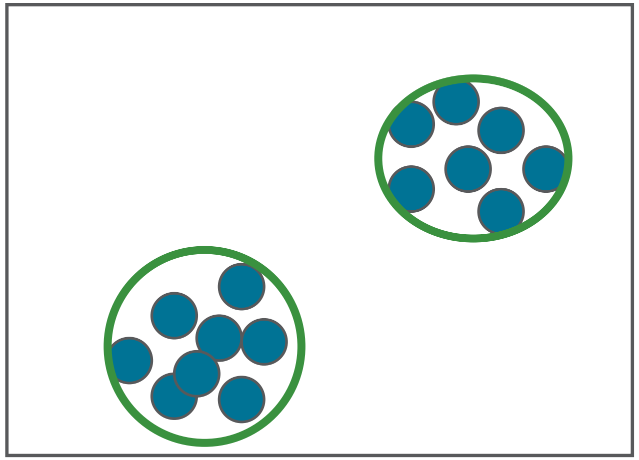
1

Train the model using normal workload data



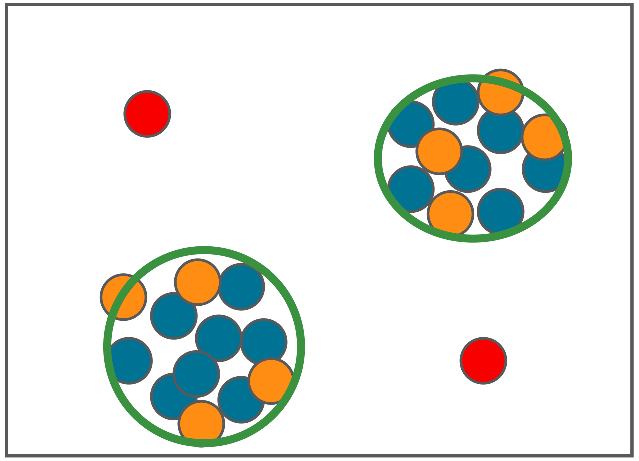
2

Model determines how to classify normal observations based on the combination of performance metrics across key dimensions



3

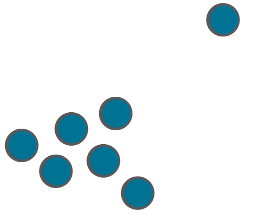
New observations can be classified as anomalies if combination of the metrics fall out of normal classification



ORACLE

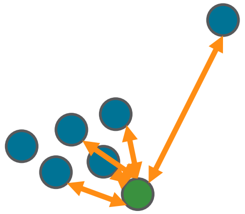
Local Outlier Factor

1



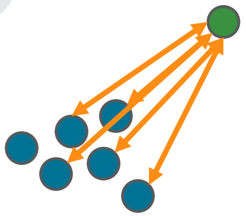
Anomalous data points are further away from the center of all data points & more isolated than the other data points

2



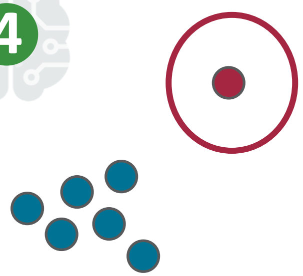
The distance between a single data point and it’s closest neighbours can be measured

3



Anomalous data points will have greater distance to their closest neighbours than other data points

4

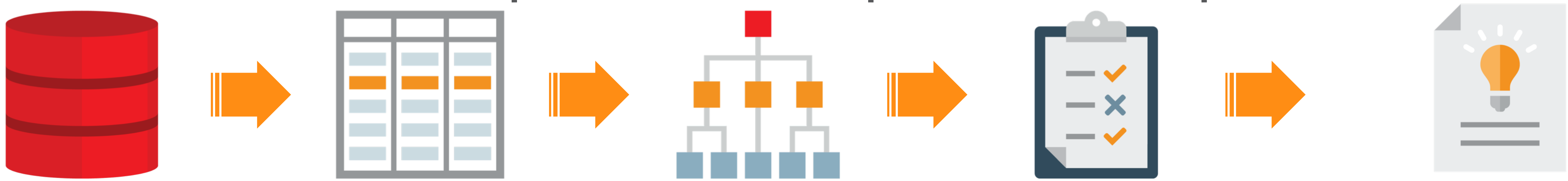


Data points that have significantly greater distances than other data points can be identified as **anomalous**

Prediction (Every 5 minutes)

5 X 1 min metrics captured for each dimension & ASH report captured for later analysis

Each anomaly is compared against the SME rules to determine which dimension it applies to

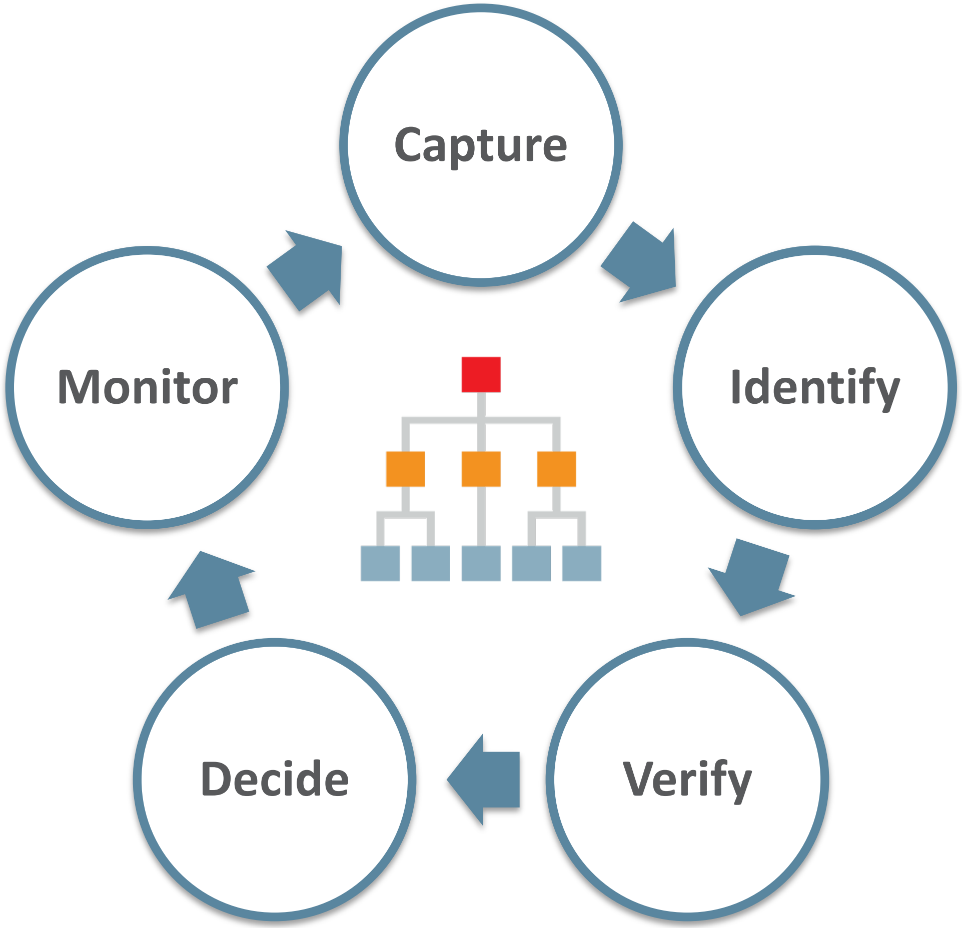


Metrics evaluated by the primary model to determine if there are anomalies

If there is no primary model
(i.e. <7 days of data or <=95% model confidence)
then SME rules are used for anomaly detection

Any anomalies are raised along with recently captured ASH report

Identify the best indexes



- An expert system that implements indexes based on what a performance engineer skilled in index tuning would do
- It identifies candidate indexes and validates them before implementing
- The entire process is full automatic
- Transparency is equally important as sophisticated automation
  - All tuning activities are auditable via reporting

Conclusions

- ML is here to stay and is just getting started
- The last 2.5 years of advances in this field dwarfs the previous 50 years of growth
- We need to identify use cases to make the business better
- Modeling and ML infrastructure will become standard aka AutoML
- Getting the right data to train matters to have a successful outcome
- Models will get better with sparse data
- Most enterprise applications are already using embedded ML

ORACLE


Questions