Technical Disclosure Commons

Defensive Publications Series

February 2023

Visualize and Correlate changes to network device to anomalies in network

Software Patents

Krishna Mahadevan Ramakrishnan

Shiva Prakash S M

Prashant Nandagadi

Venkatesh Ramteke

See next page for additional authors

Follow this and additional works at: https://www.tdcommons.org/dpubs_series

Recommended Citation
Patents, Software; Ramakrishnan, Krishna Mahadevan; S M, Shiva Prakash; Nandagadi, Prashant; Ramteke, Venkatesh; and Nanjundaswamy, Chandramouli, "Visualize and Correlate changes to network device to anomalies in network", Technical Disclosure Commons, (February 07, 2023)
https://www.tdcommons.org/dpubs_series/5662

This work is licensed under a Creative Commons Attribution 4.0 License.
This Article is brought to you for free and open access by Technical Disclosure Commons. It has been accepted for inclusion in Defensive Publications Series by an authorized administrator of Technical Disclosure Commons.

Inventor(s)
Software Patents, Krishna Mahadevan Ramakrishnan, Shiva Prakash S M, Prashant Nandagadi, Venkatesh Ramteke, and Chandramouli Nanjundaswamy

This article is available at Technical Disclosure Commons: https://www.tdcommons.org/dpubs_series/5662

Visualize and correlate anomalies in monitored infrastructure objects

OpenText R&D Lab
Krishna Mahadevan Ramakrishnan, Shiva Prakash S M, Prashant Nandagadi, Chandramouli Nanjundaswamy, Venkatesh Ramteke

Problem description

NOM products enable users to manage their network by enforcing controlled changes to configuration of network devices apart from monitoring performance, fault and compliance of all devices in their network. While monitoring a network, it is always been a challenge to deduce and represent correlation, among anomalies across thousands of devices, in a format that users can consume, validate and act on. This paper proposes a method to deduce and visualize, possibly related anomalies on connected devices in the network based on user's selection of an anomaly on a device of interest. While this paper takes network domain as an example to demonstrate the idea, it could be applied to any of the infrastructure and application management solutions in a datacenter. This paper proposes a unique approach to solve this problem by leveraging ML and network operator's deployment/domain knowledge

Terms Used

NOM	OpenText Network Operation Manager
Performance Troubleshooter (PT)	Performance Troubleshooter (PT) is a data exploration component of NOM that enables you to troubleshoot network issues by exploring the performance metrics on UI dashboards.
Anomaly	Anomalies are classified as deviations observed against normal expected behavior of certain characteristics of monitored objects. In network domain it could be fault, thresholds violation of performance metrics, config changes, status etc... observed of the monitored object.
Topology Range	The term "topology range" defines monitored objects within a pre-configured topology hop. The active events to be considered for correlation are bounded by topology range or topology hop number as configured by the user (else model relies on default topology hop of 2)

Proposed solution

The idea proposed in this paper is a comprehensive solution that helps users to detect and visualize correlation among anomalies detected in the infrastructure they manage.

The solution work by building machine learning models on anomalies detected by NOM. These anomalies are presented to user on an active Performance Troubleshooter (PT) dashboard. The visual representation provides first level of insight to the user. Many of the real-world competitive solutions stop at this initial visual representation.

Our solution allows user to analyze further by providing correlation between combination of objects on which anomaly occurred and the anomalies on other objects within the topology distance. The

analysis of requested correlation is presented to the user on active PT dashboards. Using this visual representation, user gets insight on how various anomalies are related within a configured topology distance. The solution can also be tunned by user to introduce supervised learning via custom correlation allowing to discover correlation between undetected objects/anomalies.

The solution contains following phases:

1. Visualize active anomalies with probability analysis on PT dashboard.
2. Provide correlation with anomalies within the topology distance by selecting one or more combination of active objects/anomalies on PT dashboard
3. Allow custom correlation to discover and visualize undetected active correlations

The details on visualization and analytics algorithm used at each phase are as described below:

Phase 1: Visualize active anomalies with probability analysis on PT dashboard.

The backend model starts off by recording occurrence of every anomaly against the object on which it was reported. The higher the occurrence of anomaly on an object, higher the probability. These probabilities are pushed to user on the visual map representations.

Below figure shows how we propose to visualize anomalies such as change, performance, incident, status, forecasted values, predicted failure etc… as overlays, in a topology map, on individual nodes.

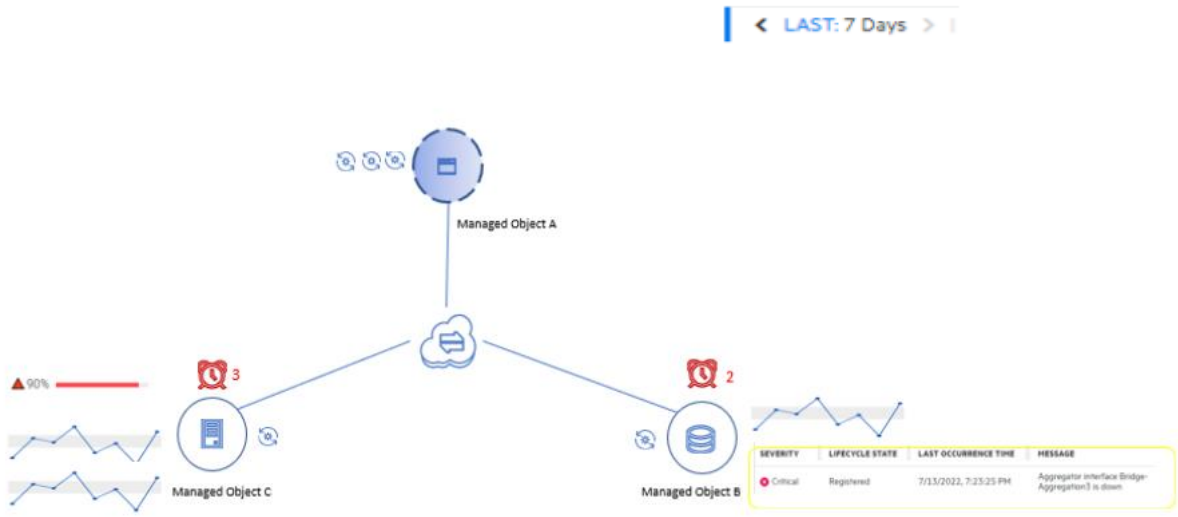

Figure 1

For the last 7 days, Managed object A has three instances of ( ) which could be config change while Managed Object B & C have a single ( ) config change while multiple other anomalies are shown as overlays on the node.

The anomalies with higher probability of occurrence are colour coded. Example, a red icon in above figure indicates a high occurring anomaly on the object.

Phase 2: Provide correlation with anomalies within the topology distance by selecting one or more combination of active anomalies or objects on PT dashboard

We now allow user to select one or more combination of objects/anomalies on PT dashboard. Post this selection, PT visualizes how other active anomalies within the topology distance are correlated with user selection and to what degree of correlation.

Below PT dashboard shows one such representation which visualizes one or more correlated values on related devices, when one of the overlaid anomalies of a node in the topology map is selected by the user.

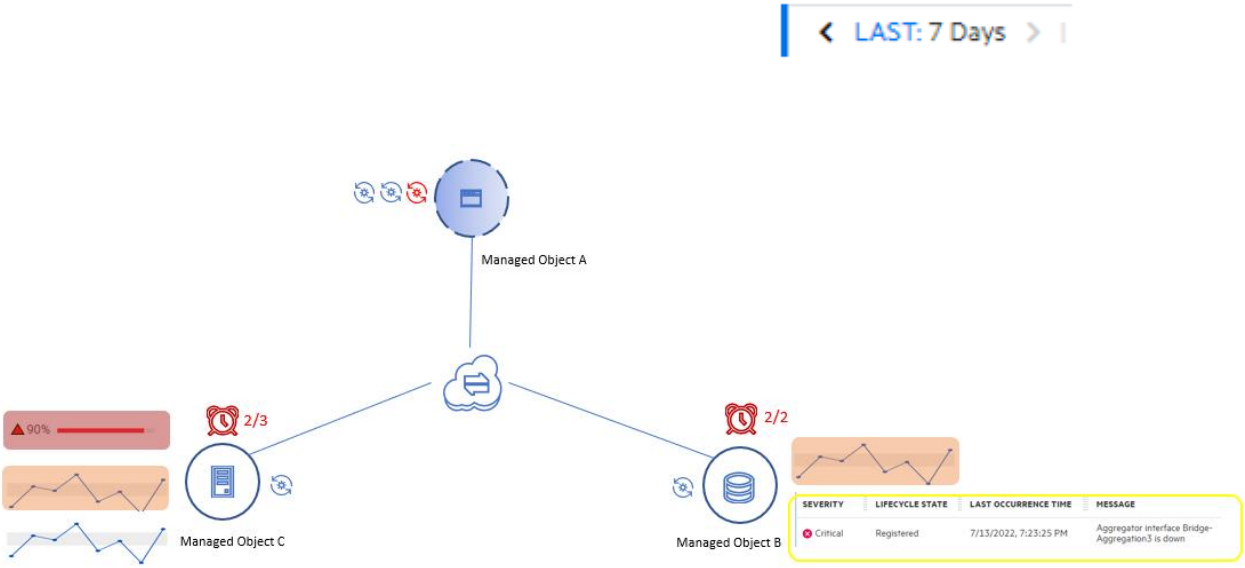

Figure 2

In the above topology, when the user selects an anomaly ( ) in Monitored Object A, the different correlated anomalies in related devices, Monitored Object B and Monitored Object C, are highlighted. ( ) overlay depicts the number of related anomalies in Monitored B and Monitored C. The related instances of anomalies themselves are highlighted with different colors based on the strength of correlation. For example: The anomaly with background color - is highly correlated than the one with background color . While the those with plain background depict unrelated anomalies.

Another scenario is shown in Figure 3, were user selects multiple anomalies for correlation.

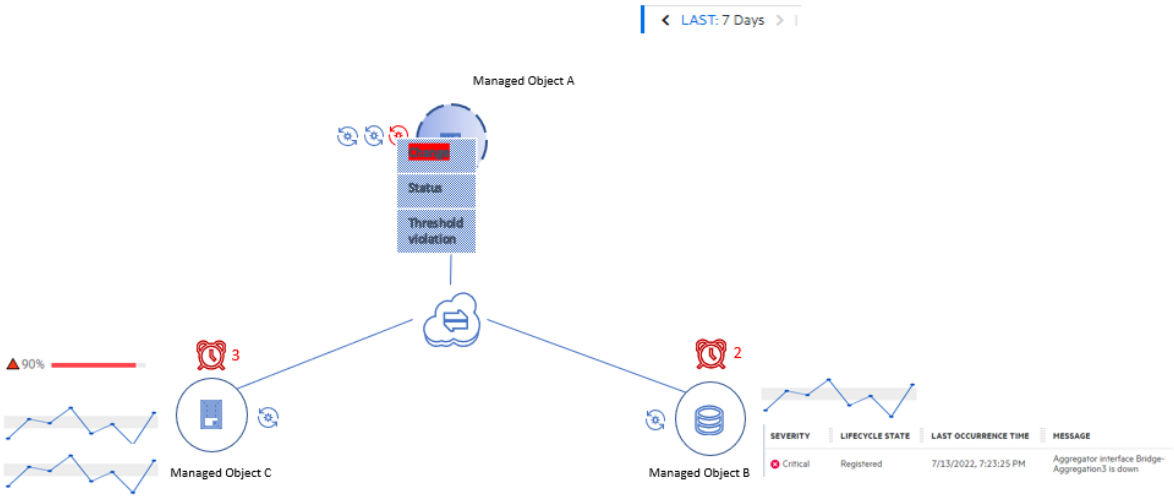

Figure 3

In the above visualization, when the user right-clicks on an anomaly ( ) on Monitored object A, based on the corelation detected by the algorithm, the user is presented with list of possible anomalies (change, status, threshold violation etc..) that he will like the algorithm to use to calculate the correlated anomalies on connected devices. The algorithm will be executed with the input from the user to recalculate the correlated values and highlight them in the topology map as was done in Figure-2. The difference between and Figure 2, is that in Figure 2, the corelation is deduced by the algorithm by operating on the entire data pattern while in Figure 3, the user drives the algorithm to calculate the corelation among the anomalies of his choice and view the result in the topology map.

To provide information as depicted in Figure 2 and Figure 3, we start by analysing anomaly data in timeseries format and run it through multi-variant regression analysis.

Multi variant regression is a machine learning technique that can be used to analyse the relationship between a single dependent variable and several independent variables. The objective of multiple regression analysis is to use the independent variables whose values are known to predict the value of the single dependent value. Each predictor value is weighed, the weights denoting their relative contribution to the overall prediction.

The user selected anomalies form the independent variables in our regression model. All the other active anomalies in the topology range, form the dependent variable in regression model. To improve performance on analysis of each of the dependent variable, we order anomalies within topology distance by occurrence probability. Anomalies with highest probability are selected early for analysis We can optimize further by restricting number of anomalies analysed as dependent variable via a count max on the probability order.

Phase 3: Allow custom correlation to discover and visualize undetected active correlations

As part of final phase, we introduce supervised learning to our Machine leaning model by allowing user to introduce/inject custom correlation.

Consider a scenario arising from our Phase-2 analysis, where user has selected two anomalies viz. CPU utilization violation on node-1 and config change event on node-1. On selection, anomaly on chassis temperature violation is not considered for analysis due to lower probability factor. User can override the probability for temperature violation event and force algorithm to always consider it while evaluating CPU utilization and config-change event correlation. This in turn bring in the associated correlation obtained via temperature violation anomaly, thus increasing the range of correlated anomalies to be visualized. We have now introduced supervised learning to our algorithm which will always consider temperature violation event on all occasions when user selects CPU utilization or config change event on node 1.

Conclusion

Using Visual correlation and Machine learning analysis, we can detect correlation among anomalies on monitored object and present it to users so that they could visualize and act on the anomalies appropriately across monitored objects. There have been several attempts made to apply Analytics/AIOps to solve this problem, but they are rarely accurate as they purely rely on ML and don't consider the domain aspects. As said earlier, though network domain is taken as an example here, the idea proposed in this paper can be applied to any infrastructure domain that requires constant monitoring for fault, performance, config change and compliance.