Vitrage root cause analysis

Vitrage root cause analysis enables service providers to gain insights about problems in their cloud in order to address issues before they occur, for improved service assurance.

Vitrage is an OpenStack analytics service, initiated and led by Nokia CloudBand, for organizing, analyzing and expanding OpenStack alarms and events, generating insights on the root cause of problems and deducing the existence of problems before they are directly detected.

What is root cause analysis?

To understand root cause analysis, we need to first define “root cause”: “A factor is considered a root cause if removal thereof from the problem-fault-sequence prevents the final undesirable event from recurring”.

We are trying to imagine what the world would have looked like if the fault had not occurred. Had this event not occurred, then the problem would not have developed. Therefore, determining the root cause of anything can be difficult.

Root cause analysis (RCA) is the method of identifying root causes among system events, usually in failures. We generally look for root causes of “bad things”.

Why is this important? Because determining root causes through RCA opens a lot of possibilities (see Figure 1):

- **Insights**: Gain insights and understand your system and how it works.
- **Fast reaction**: By understanding what caused the problem, you have already gained the benefit of knowing how to fix it in a way that is effective and doesn’t rely on possible faulty elements.
- **Accountability**: It’s very important to be able to say why things happened and how you are going to fix them. For the telecommunications industry, and in particular when transforming to a cloud infrastructure, reliability (5 nines), performance and regulatory compliance are essential.
- **Operations**: After you know what happened and why it happened, you can move on to the practical task of fixing the problem.
- **Prediction**: RCA is actually the reverse process of prediction; once you understand how to determine the root cause, you can reverse the process and proactively address problems before they occur.

1 Wikipedia
Three fundamental approaches to RCA

**Expert judgment**
Even before computers existed, experts were used to assess problems and try to figure out how to avoid having problems recur. This approach depends on a subjective point of view or on an expert’s personal bias.

**Statistical**
A statistical approach is based on collecting large amounts of data and correlating events. When a correlation is found, we understand that there is a link between the events. At this point it’s crucial to understand which event occurred first, causing the other event to happen, based on a constructed timeline. However, this approach does not always work, especially in highly complex systems with multiple events occurring simultaneously. The purpose of a statistical analysis is to detect whether the presence or absence of certain factors impacts the probability of a fault to occur.

**Analytical**
An analytical approach relies on counterfactual reasoning and employs formal approaches for analyzing causal dependencies. This approach asks the question “If X would not have occurred, would Y have occurred?” The difficulty with this approach is that it requires greater amounts of data to reach conclusions.

**The Vitrage approach**
The Vitrage service collects data from many unrelated data sources, both internal (such as Nova, Cinder, Neutron and Heat) and external (such as Zabbix and Nagios). Each source contains a wealth of data regarding both physical and virtual resources but lacks the knowledge to see and understand the full picture—and the full picture is essential to ensure that error alarms get to where they need to go, on time.

OpenStack ingests the data, then Vitrage codifies the data sources into templates based on YAML (Yet Another Markup Language) files, which express the rules by which the system works. When Vitrage is integrated with CloudBand Infrastructure Software, it comes with ready-made templates that can then be modified for specific usage.
Figure 2 shows the high-level architecture of Vitrage. The information collected from all the various internal and external sources is expressed in Vitrage by an entity graph that displays the data sources and the relationships between them.

The benefits of automated expert judgment are:

- A holistic view of many data sources together
- Fast and immediate
- RCA insights are propagated throughout the system, so that deduced alarms and states can expose faults to all relevant users
- Allows user configuration based on specific needs

Vitrage required for complex networks

Network functions virtualization (NFV) has made networks more complex, with both physical and virtual elements in the same network. This creates gaps in service providers’ ability to efficiently monitor all the services in their networks and to easily understand all the relationships between the various elements.

Vitrage addresses these issues by providing a clear and holistic view of the entire network. This network view shows how all the different entities—both physical and virtual—relate to one another. Vitrage allows providers to expose the faults that occur, to understand them, and to clearly view changes that occur to the various resources in the network as a result of failures.
Because Vitrage is an open-source service based on OpenStack, it is very flexible. Vitrage can be configured to accept data from a wide variety of sources and is easily extensible.

Vitrage layers of information

Vitrage provides several layers of information to network users:

- **Root cause analysis:** Understand what causes faults to occur
  - Causality between alarms coming from different data sources
  - Enabling fast and efficient troubleshooting and recovery
  - May trigger proactive actions to provide service assurance
  - Predictive RCA and preemptive actions: Raising alarms about future events before they happen and taking action using Vitrage's ability to link between associated resources
  - Fine-grained RCA: Whether and how each individual virtual machine (VM) is impacted by problems with the physical network interface cards (NICs)

- **Deduced alarms and states:** Raising alarms and modifying states based on system insights
  - Expose more faults and changes to resources than any other OpenStack service
  - Instantaneous alarms—unlike other OpenStack services, which can take a long time to receive alarms, thereby causing extreme problems
  - Instantaneous alarms enable system administrators to solve problems and reroute the network easily and efficiently

- **Complete view of the system**
  - Understand how different entities relate to one another: physical network functions (PNFs), virtual network functions (VNFs), different application layers
  - Clear visualization to reflect real status of the system and to improve troubleshooting (see Figure 3)

**Figure 3. Vitrage user interface examples**
Vitrage and Nokia CloudBand

Vitrage is an integral part of the Nokia CloudBand portfolio, allowing each product to provide root causes and also enabling an end-to-end service assurance flow from the hardware to the network service (deduced alarms) and from the network service to the hardware (RCA).

Vitrage use case examples

Vitrage use cases include use cases for triggered actions and for end-to-end service assurance.

**Triggered actions: Deduced alarms and states**

Problems on a switch can sometimes have a negative impact on the VMs running on hosts attached to the switch. As shown in Figure 4, Vitrage raises an alarm on all VMs associated with that switch. In addition, the state of all these VMs may to be changed to “ERROR” even if it is not possible to directly monitor their exact state. VMs might not be monitored for all aspects of performance or perhaps the problem in the switch makes monitoring them difficult or even impossible. Instead, Vitrage deduces that a problem exists on the VMs based on the state of the switch, then raises alarms and changes states accordingly.

**Triggered actions: Root cause indicators**

Vitrage enables cause-and-effect tracking, i.e. that the problem in the switch caused a problem in the VMs. It is important to note that not all deduced alarms are caused by a trigger, which might only be an indication of correlation, not causation. After the local “causes” links are detected and registered, Vitrage can then follow them one after another to track the full causal chain of a sequence of events.

Figure 4. Deduced alarms and states
**End-to-end service assurance**

In an end-to-end service assurance use case, the process is as follows:

1. A NIC failure triggers alarms on the impacted hosts and VMs. Alarms are sent to the VNF Manager (VNFM).

2. The VNF Manager identifies the impact on the VNFs and sends notifications to the Network Functions Virtualization Orchestrator (NFVO).

3. The NFVO correlates the impact of the VNFs’ failure to network services and sends notification to the operations support system (OSS).

4. The service layer correlates the failure impact with the relevant end-to-end service.

**Notes and acronyms**

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CBAM</td>
<td>CloudBand Application Manager (the Nokia CloudBand VNFM)</td>
</tr>
<tr>
<td>CBIS</td>
<td>CloudBand Infrastructure Software (the Nokia CloudBand NFVI)</td>
</tr>
<tr>
<td>CBND</td>
<td>CloudBand Network Director (the Nokia CloudBand NFVO)</td>
</tr>
<tr>
<td>EMS</td>
<td>element management system</td>
</tr>
<tr>
<td>NFVI</td>
<td>network functions virtualization infrastructure</td>
</tr>
<tr>
<td>NS</td>
<td>network service</td>
</tr>
</tbody>
</table>

Figure 5. End-to-end service assurance
Additional Vitrage features

- Integration with the Red Hat® OpenStack Platform
- Integration with the collectd data source. collectd is a fast system statistics collection daemon with plugins, such as the DPDK plugin, that collect different metrics; these metrics can trigger alarms such as interface failure or noisy neighbors
- Integration with the OPNFV Doctor data source, which handles notifications sent from the Doctor monitor to create deduced alarms on relevant VMs and applications
- Vitrage enables a static data source to recover cloud resources that cannot be retrieved dynamically.

Conclusion

Vitrage root cause analysis automates the process of expert judgment to provide a holistic view of an entire network. With Vitrage, providers can expose faults that occur, understand them and clearly view changes that occur to the various resources in the network as a result of failures. Use cases include triggered actions—deduced alarms and states plus root case indicators—plus end-to-end service assurance.

For more information about how Vitrage can help you to efficiently monitor all the services in your network, please visit the Vitrage wiki.