Monitoring Platform-1

From Reactive Management to On-line Proactive management by On-line Monitoring of the enterprise

Introduction

Organizations today are moving from reactive management to more proactive management. Proactive management is some times understood as trends analysis, however this is the lowest level of proactive management, which is called off-line proactive management.

There is a need for an On-line proactive management methodology. For example we would like to discover problems and anomalies in the enterprise just before they occur or at the time they occur. We must know about them before the customers discover them. In order to do that we need to monitor all the relevant components of our enterprise in a high frequency, and make sure that all is well.

Examples of such monitoring in the network management area can be:

- Availability monitoring of all important end-points
- Line quality of all clients
- Bandwidth utilization exceptions
- Memory utilization exceptions
- CPU utilization exceptions
- and more.

There can be many areas where this type of proactive high frequency monitoring can be relevant:

- The enterprise network
- The enterprise server farm hardware & software
- The end-users desktops, telephone and other related devices
- Business applications and components
- and many others.

Although there are many management platforms that address this type of monitoring, we found out that many of them fail when it comes to scale. Their common scaling method is to put agents in or near the monitored devices. This is not simple to implement and in many cases it is not possible.

When they try to monitor large number of components from a central point they fail to do high frequency monitoring for a large number of parameters on a large numbers of components. They usually suffer from a very slow monitoring cycle.

The approach taken in this paper is to addresses this issue of on-line-proactive management by enabling real time monitoring, of large number of parameters in large organizations using a single central collection station or a relatively small number of agents to collect data from a large number of components in the enterprise.

Monitoring needs

The enterprise of today becomes more and more complex. The number of intelligent components that exist on their networks is growing in an exponential rate, when such a component can be either a hardware device or a software application. Those components get more and more complicated.

Increasing number of monitored parameters

If in the past it was enough to know if a component is up or not, however today we can not suffice with only this information. We expect to know much more on each component. We want to know for example:

- What its performance
- What resources does it use, and how much of each resource does it use
- What is it's response time
- whets it's jitter
- And so on.

Multi-Protocol parameter collection

More intelligent components, do not lead to more standardization in accessing the required parameters. Any monitoring solution should support multiple methods for collecting data.

Examples to such protocols are:

- snmp
- icmp
- wmi
- http
- sql
- and more

Simple installation and operation

As the monitoring operations get more and more complex, the need to simplify its installation and configuration grows. It is not very practical to have a monitoring agent for every component or small group

of components, when the number of monitored components grows rapidly. This makes the operability and manageability almost impracticable. The ideal solution could have been a centralized monitoring product, which is able to perform its required actions from a single point, however it is clear that this approach will fail on scale.

It looks like the optimal solution will be a centralized monitoring platform with as less agents as possible, when every agent monitors as many components as possible. This will simplify the installation and operation of this kind of platform and enable scale.

Complex correlations and conditions on the monitored parameters

The requirements for conditions that are needed to be tested on the monitored components have also changed dramatically in the past few years. If in the past, events were based on status changes and threshold passing, today there is a need to correlate multiple parameters' behavior over time, in order to detect anomalies in the components, and alert or even prevent problems prior to the time they escalate to failures.

Flexibility

As the number of possibly monitored parameters increase exponentially, and the number of the possible correlations that make sense increase exponentially too, it is clear that out of the shelf products can not prepare in advance to answer all the possible needs of all enterprises. As such it is needed that a monitoring solution will be flexible in a way that will enable with ease to add more monitoring parameters and more correlation rules, and conditions.

Scalability

Scale can be measured by this relation:

monitored components x
average parameters count x
average monitoring frequency x
average affective correlation rules

of course, this equation, gives only a ruff estimate to the scale required by a monitoring solution. As can be seen from the previous equation there are 4 major aspects of scale that should be taken into consideration:

Component count scale

As said before, the number of intelligent components in the enterprise increases exponentially both software and hardware. Any monitoring solution must address this increase in a way that will not increase the monitoring cycle time.
There should not be also any effect between the monitored nodes (i.e, if a given node does not respond it should not effect the time the other node is monitored.

Collected parameters count scale

Each component has multiple parameters that can be collected from it. Every monitoring solution should make it possible to collect the increasing number of parameters from the different components.

Monitoring frequency

Once it was good enough to make a certain checks every day or hour, now it is required to know of changes to parameters, on line, as they occur. A monitoring solution must support short monitoring cycles.

Correlation rules

Now days it's not enough just to collect the data, checks must be made on the collected data, and some of the checks can be quite complicated. A monitoring solution must scale well with the number of checks performed.

Monitoring methods

There are many methods for monitoring the components that compose the enterprise. For example:

1. An agent located on the component, collecting the requested information and passing accumulated data to a central point.
2. A remote agent collecting data from multiple components accumulating the collected parameters and passing them to a central point
3. A central collector which collects data from all components, and processes them.

For different components there are different optimal data collection methods. A monitoring platform should support all methods

Central monitoring data repository

No matter what monitoring method is taken, it is recommended that there will be a central repository holding all the monitoring data. This repository can be used for:

1. History data collection
2. Trends analysis, and pattern discovery
3. A central location for data used for events correlations and compound conditions

and many more uses.

Monitoring Data format

The monitored parameters data can be kept in many formats however there are 2 recommended methods to hold the data:

1. Summary data should be held in a relational database, so it will be easily queried and reported.
2. History data should be held in an RRD format (Round Robin Database). This method is a recommended because it puts bounds to the amount of storage used for keeping the data.

Collecting monitoring parameters and keeping them in a raw data format, allows the maximum flexibility, but it also becomes a liability, specially in large organizations when the size of the data collected can increase to huge amounts, and since there is hardly any critical need to know what was the exact response time for a given transaction a few months ago.

Monitoring platform-1

Monitoring Platform-1 from Jilroy Technologies addresses the issue of on-line proactive management by enabling real time monitoring, of large number of parameters in large organizations using a single central collection station or a relatively small number of agents to collect data from a large number of components in the enterprise. The product addresses all the existing monitoring needs of enterprises today, in a way that focuses on scale.

Addressing the Monitoring needs

Increasing number of monitored parameters

Monitoring Platform-1 enables the user to determine which parameters will be monitored, and in which frequency, on which components. The selection is very robust, and very simple. The user can define which parameters will be collected from which components, using masks on discovery information on the components.

Multi-Protocol parameter collection

Monitoring Platform-1 is built to support multiple collection protocols. It currently supports the following protocols:

- ICMP
- SNMP
- SQL
- HTTP
- CSV

In the coming versions it will support the following protocols:

- WMI
- TCP Ports monitoring
- and others.

Its supports of the monitoring protocols integrates with all its other capabilities, like support in increasing number of monitored parameters, ease of installation and scale.

Simple installation and operation

Monitoring Platform-1, is built as a centralized monitoring product. It is centrally managed and controlled. It can operate on a single machine monitoring large numbers of parameters on large number of components in high frequency. When scale requires, it supports distribution of its components to additional machines. It's installation and use out of the box are very simple, however it enables the advanced user to tailor it to its exact needs.

Complex correlations and conditions on the monitored parameters

Monitoring Platform-1 has a built in mechanism for defining correlation rules and conditions on the monitored data, including the historical data, so that events can be scheduled and event handlers launched, based on these correlation rules and conditions.

Flexibility

Monitoring Platform-1 was designed for flexibility. It has several levels where it becomes obvious:

Monitoring Parameters selection

With Monitoring Platform-1 it is easy to define what parameters will be monitored on which nodes, in a simple and robust way that uses masks and information discovered on the monitored components. It is

not needed to specify for each node specifically what parameters will be collected. It is possible, and recommended to use generic rules to define those values.

Graphical User Interface flexibility

The product was designed in a way that enables the user, to tailor it's User interface to its exact needs. The user can change the product's menu, and add or remove query reports. The product has a permission system that can control what each user will see.

Product's components location

In order to support scale the product was designed to allow flexibility in the location of the product's components. These components can be moved from the server's central point to near by or remote computers.

Scalability

Monitoring Platform-1 was designed for scalability, in all the aspects mentioned above.

Component count scale

The product, out of the box, was built that the monitoring components can collect data from a large number of components concurrently from a single process, in parallel. However the product supports having multiple collection processes, located either on the same machine or distributed in the enterprise. This capability enables scale, as the monitoring of different components can be distributed over the different collection processes.

The product is built in a way that collecting data from one component does not affect the collection of data from other components.

Collected parameters count scale

Each collection process can collect multiple parameters from any given number of components. As described before, the scale in the number of monitored parameters is handled by adding more collection processes.

Monitoring frequency

As the product was designed in a way that monitoring of a given node, can be done in parallel with monitoring of others, there is no theoretical limit to the monitoring frequency, and the only actual limits are CPU, memory, and bandwidth, which are addressed by the capability of adding more collection processes, and distributing the monitored components between them.

Correlation rules

The correlation rules analysis is performed in a special process. This process can also be multiplied and distributed if scale is required.