PRODUCT DATA SHEET

OBSERVE

Active Observability

INTRODUCTION

Active Observability is the ability to monitor and understand the behavior of an entire system, including the frontend, backend, and infrastructure. It is a key aspect of modern system design because it enables organizations to identify and troubleshoot issues in real-time, as well as optimize the performance and reliability of their systems.

Apica is an Active Observability data fabric. Our platform uses AIML capabilities to analyze log data in real time, allowing organizations to identify issues and optimize the performance of your systems.

CAPABILITIES

Apica's Observe capabilities include:

Detect anomalies in log data: This is important because it allows organizations to identify issues as they occur, rather than waiting for them to become more severe. By detecting anomalies early, organizations can prevent problems from escalating and minimize their impact on the system.

Get context for log data: This is achieved through the use of metadata, which is additional information that is associated with log data. By providing context for log data, organizations can more easily understand the causes of issues and take appropriate action to resolve them.

Analyze your API flows, troubleshoot issues, and understand API usage: Essentially, get API Observability into your infrastructure.

To analyze API flows, Apica may collect data on API requests and responses, including information about the endpoints being called, the parameters being passed, and the payloads being sent and received. It may also track metrics such as response times and error rates. This data can be used to identify trends and patterns in API usage, as well as to detect issues or bottlenecks in the API flow.

To troubleshoot issues, Apica uses this data to pinpoint the root cause of problems and suggest solutions. It may also provide tools for debugging API calls, such as the ability to replay requests or view request and response payloads.

HIGHLIGHTS

•Apica is an active observability data fabric that uses AIML capabilities to analyze log data in real-time, detect anomalies, provide context for logs and API flows, as well as troubleshoot issues and understand usage.

• Log aggregation allows organizations to gain a comprehensive view of their systems and applications by collecting log data from various sources into one central location.

• Application metrics are key indicators of the performance and health of an organization's systems which include response time, error rate, throughput, etc., while infrastructure metrics measure the performance & health of underlying infrastructures like server hardware or network traffic.

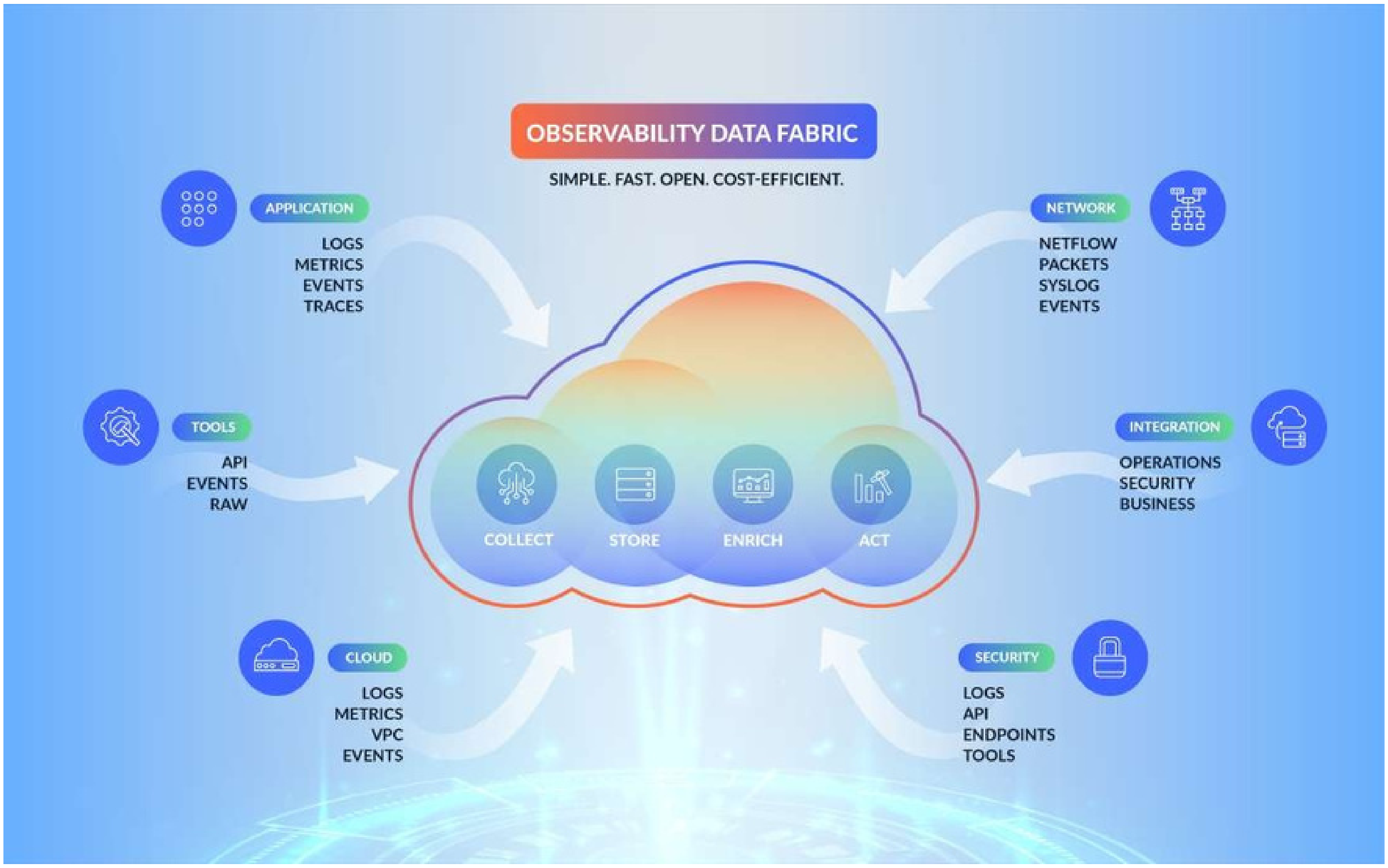
• Tracing refers to tracking requests/transactions through distributed services in order to identify bottlenecks or debugging issues; Apica offers tracing capabilities with its platform providing visibility into the entire lifecycle request flow including latency & dependencies between different services.

• Data convergence enables organizations to bring together multiple sources' data for analysis within a single place; with integrations across cloud services & databases, Apica makes it easy to collect & analyze structured machine data from relational databases such as PostgreSQL, etc.

To understand API usage, we provide tools for visualizing API data and metrics, such as graphs and charts that show API usage over time or breakdowns of API calls by endpoint or client.
It may also provide reports and other forms of analysis to help users understand how their APIs are being used and how they can optimize their API design and implementation.

Apica allows to converge and analyze any data source, including logs, metrics, events, and traces.

With the use of relational databases and "machine data," Apica wants to make it simple for businesses to combine operational intelligence.



Additionally, Apica provides Kubernetes Monitoring capabilities as well. To provide end-to-end observability for Kubernetes, Apica offers a range of features such as real-time monitoring, alerting and notifications, and visualization tools to help users understand and troubleshoot issues in their applications and clusters.

Apica also provide hassle-free integrations with other tools and platforms, such as Prometheus and Grafana, to provide a complete view of an application's performance and behavior.

Logs

Log aggregation, management & analytics

Utilize real-time alerting and monitoring that scales with your systems to organize and analyze your logs without the need for indexing and to fully exploit the insights contained in your data.

- Live data transformation, enrichment, and parsing
- Extract metrics from unindexed logs
- Clump noisy logs into templates

The observability platform from Apica includes log aggregation, management, and analytics capabilities, allowing organizations to gain a comprehensive view of their systems and applications. This helps you improve reliability, performance, and efficiency, and to make informed decisions about how to optimize operations.

Log aggregation, management, and analytics are imperative components of any observability platform.

Log aggregation refers to the process of collecting log data from various sources into a central location. This can be useful for tracking events, identifying patterns, and troubleshooting issues across an organization's systems and applications.

Log management involves organizing and storing log data in a way that makes it easy to access and analyze. This can involve categorizing logs by type, setting up retention policies, and establishing processes for data backup and recovery.

BENEFITS

The key benefits of Observe include:

Comprehensive visibility: An observability data fabric provides a complete view of your system, allowing you to understand how all of the different components of your system are interacting and performing.

Scalability: An observability data fabric is designed to scale with the size and complexity of your system, allowing you to collect, process, and analyze large amounts of data without any performance issues.

Flexibility: Collect and analyze data from a wide range of sources, including applications, infrastructure, and services, making it easy to adapt to changing requirements.

Root cause analysis: Get the ability to easily trace and correlate data across different components of your system, making it easier to identify and resolve issues.

Proactive monitoring: An observability data fabric enables you to proactively monitor your system and identify potential issues before they become critical.

Cost-effective: An observability data fabric can be cost-effective as it allows you to leverage existing data collection and analysis tools, rather than having to invest in new ones.

Collaboration: An observability data fabric allows for collaboration between different teams, such as development, operations, and security.

Log analytics refers to the process of using specialized tools and techniques to extract insights from log data. This can include performing queries, generating reports, and visualizing data in various ways. By analyzing log data, organizations can better understand the performance and health of their systems and applications, and identify potential issues before they become major problems.

Metrics

Application & infrastructure metrics

Envision your business-level logic using all available metadata, including infrastructure, network, security, and application metrics.

•Connect metrics to all observable data
•Advanced alerting using PromQL syntax
•Create personalized data maps to track system health

Application and infrastructure metrics are key indicators of the performance and health of an organization's systems and applications. Application metrics are specific to an organization's applications and can include things like request latency, error rates, and user engagement.

Application metrics are data points that measure the performance, health, and usage of an application. They help developers understand how their application is functioning and identify areas for improvement. Some examples of application metrics include:

•**Response time:** the amount of time it takes for the
•application to process a request and return a response to the user.
•**Error rate:** the percentage of requests that result in an error
•**Throughput:** the number of requests that the application can handle in a given time period
•**Memory usage:** the amount of memory being used by the application
•**CPU usage:** the amount of CPU resources being used by the application

Infrastructure metrics, on the other hand, are data points that measure the performance and health of the underlying infrastructure on which an application is running. This can include things like the server hardware, the operating system, and the network. Some examples of infrastructure metrics include:

•**Disk usage:** the amount of storage being used on the server
•**Network traffic:** the amount of data being transmitted over the network
•**CPU utilization:** the amount of CPU resources being used on the server
•**Memory utilization:** the amount of memory being used on the server
•**Server uptime:** the amount of time that the server has been running without experiencing an outage

Infrastructure metrics include CPU usage, network bandwidth, and disk space utilization. By monitoring infrastructure metrics, organizations can gain a better understanding of how their infrastructure is performing and identify any potential bottlenecks or capacity issues.

Apica's observability platform includes both application and infrastructure metrics, allowing organizations to gain a comprehensive view of their systems and applications. This helps to improve reliability, performance, and efficiency, and to make informed decisions about how to optimize operations.

Traces

Trace transactions between distributed services

Tracing refers to the process of tracking the flow of a request or transaction as it moves through an organization's systems and applications. This involves tracking things like request latency, error rates, and the dependencies between different services..

Tracing transactions between distributed services can help developers understand the flow of data between different parts of their systems and identify bottlenecks or issues that may be impacting performance.

Apica utilizes distributed tracing to provide a comprehensive view of the entire lifecycle of a request. This helps users to identify and fix problems more efficiently, as they can see exactly where the request is being slowed down or failing.

Tracing transactions between distributed services can be challenging, as it requires tracking the flow of a request or transaction through multiple systems and components. This can involve collecting and analyzing data from various sources, such as log files, network packets, and application metrics.

Apica offers to trace capabilities that allow

organizations
to track the flow of transactions between distributed services.

Apica's observability platform provides for a deeper understandingof your systems and making informed decisions about how to optimize their operations.

Convergence

Converge and analyze any data source

Data convergence is important in observability because it allows organizations to bring together data from various sources and analyze it in a single place. This can be especially useful when it comes to understanding the performance and health of your organization's systems and applications, as it allows for a more comprehensive view of the entire system.

Customers can store enormous volumes of data affordably because of this, which also hinders lock-in.

A few key useful features of Data convergence via OBSERVE include:

Overcoming the problem of running multiple applications in silos
Simplified scalability – with a single location for data storage, scaling becomes faster and easier
Increased security – with fewer routes for unethical sources to access the system, the data becomes more secure and less vulnerable to attacks
Improved disaster recovery – data can be backed up across multiple geographical locations, and the overhead of backing up multiple data stores is eliminated as all data is located in one place.

These capabilities help organizations to identify and troubleshoot issues in real-time, as well as optimize the performance and reliability of their systems.

SUMMARY

Apica's OBSERVE is a platform that provides Active Observability for modern systems, allowing organizations to monitor and understand the behavior of their entire system in real time.

It uses AIML capabilities to analyze log data, detect anomalies, and provide context for log data. T

he platform also includes Kubernetes monitoring, API observability, and hassle-free integrations with other tools such as Prometheus and Grafana.

OBSERVE allows organizations to improve reliability, performance and efficiency, and make informed decisions about how to optimize operations by aggregating, managing and analyzing log data, extracting insights, and visualizing data.

Additionally, it enables organizations to envision their business-level logic and track system health by using all available metadata, real-time alerting, monitoring, and in-stream tracing analysis.