Event Analytics for Adabas: Technical Concepts

Version 1.1

June 2014

This document applies to Event Analytics for Adabas Version 1.1 and to all subsequent releases.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Document ID: EAA-CO-11-20140630

Table of Contents

About this Guide

This guide describes the technical features and architecture of *Event Analytics for Adabas*.

The product *Event Analytics for Adabas* provides extensive functionality for monitoring Adabas Events for Alerting, Security and Auditing.

1 Introduction

Adabas is a high-performance, multithreaded, database management system, and is a key component on large ETS production sites. It supports many mission critical applications holding sensitive data in the databases involved. Some Adabas administration activities become challenging, such as monitoring the operation of these databases while guaranteeing their security, as well as answering the increasing and challenging auditing requirements.

Customers are constantly in search of solutions for improving the performance of their environment in order to offer better services to their clients. Also key for these customers is the ability to answer the auditing and security challenges of the applications based on Adabas.

Event Analytics for Adabas combines the best in performance monitoring with complex event processing to drive down costs, improve performance, prevent fraud and provide an audit trail.

With event analytics, you can use data collected to determine chargebacks, debug programs, resolve error messages, identify programs using excessive resources, track historical resource utilization, and tune the database to achieve optimal performance.

To ensure security and create audit trails, event data is collected and monitored in real-time. These streams of events are evaluated (correlated) against a unique, multi-dimensional filtering mechanism that quickly sifts through multiple event data streams, detects sought-after patterns and identifies appropriate responses - within milliseconds or less.

You can visualize data in real-time with dashboards, and be alerted proactively to unusual behaviors or events. Statistical data is cached to preserve many audit trails.

Features

- **Centralized data collection**

  Collect event data across all Adabas databases and any application (Natural, COBOL, Assembler, etc.) accessing Adabas in a central repository.

- **Operational data analysis**

  Track and analyze Adabas online and batch database operations.

- **Resource utilization**

  Identify application problems by capturing excessive calls, I/Os or volume of held records or other resources that can cause unexpected error messages or delays in processing.

- **Event processing**

Monitor inbound event streams for patterns that match defined conditions. Detect time-based, attribute and location-based relationships with unparalleled responsiveness. Support hundreds of thousands of individual event-processing scenarios simultaneously.

- **Streaming analytics**

  Correlate, aggregate, filter and query large volumes of fast-moving data from multiple sources. Enrich streaming events, detect patterns and derive contexts for improved decision-making.

- **Dynamic event rules**

  Define event rules to meet various security and auditing requirements.

- **Historical analysis**

  Put real-time data in context and analyze current events against historical norms.

- **Test and audit**

  Retain events for later event replay and analysis. Event streams can be captured and written to any standard data store, so that new event-processing scenarios can be tested prior to deployment, and audit trails recreated.

- **Integration**

  A robust integration framework includes adapters and APIs for bi-directional exchange with event sources, dashboards, clients and event correlators. It seamlessly integrates with Adabas.

- **Developer productivity**

  Developers can build, debug, profile and maintain best-practice behaviors or specific analytics that can be provided to business users for them to re-use in a plug-and-play environment.

- **Proactive alerts**

  Get automatic alerts, based upon what has happened or what is about to happen, delivered through a console, email, SMS or integration with other applications.

- **Data visualization**

  Interactive, self-service dashboards provide real-time insights into critical aspects of the business. Easily mash up data and customize visualizations.

- **Non-invasive architecture**

  Preserve your existing investments while simultaneously leveraging the latest in performance monitoring, security and event processing technologies. The event analytics non-invasive architecture has no impact on application availability.

2 Key Characteristics

One of the great challenges of monitoring solutions is the ability to be constantly technically enhanced in synchronism with the evolution of the object that they intend to monitor.

Event Analytics for Adabas guarantees seamless compatibility with new upcoming Adabas versions, thus ensuring the transparent evolution of the components and the preservation of the customer's investment.

Another important requirement is to quantify how invasive the solution is to the customer production environment. The less invasive it is, the simpler and more transparent its implementation will be in the environment.

In this context, *Event Analytics for Adabas* has unique features offering:

- Centralized information collection for multiple DBIDs;
- Parallel support and Adabas Cluster Services;
- Scalability;
- Simplified configuration;
- Reduced overhead through the use of Review "Hub" mode;
- Customization of "Command Logging";
- Collection of historical data;
- Collection of "reporting" information from clients;
- Collection of data from any application (Natural, Cobol, Assembler ...) accessing Adabas;
- Collection of data from online or batch sources;
- A pre-defined set of reports;
- An interactive interface;
- A wide range of data types;
- Definition of "user-fields";
- Online viewing of summary reports;
- Monitoring of event streams to detect and analyze complex patterns in real time;
- Additional sources of information as input;
- Immediate response to events and auditing trails through alerts;
- Customized monitoring through panels;

- Continuous analysis of data to optimize operations, mitigate risk and seize opportunities in real-time
- Maximization of database performance with minimum resource usage
- Identification of programs using excessive resources
- Detection of security breaches on sensitive data
- Proactive fraud identification
- Audit trails relating events from multiple sources
- Resolution of hot spots in complex environments to keep your business running
- Transparency into geographically dispersed database operations
- Connection to historical and current data for intelligent actions based on what has happened, what is happening and what is about to happen

3 Modes of Operation

Event Analytics for Adabas is very comprehensive and encompasses the approaches described in the areas of:

- Adabas Operational Monitoring and
- Adabas Monitoring for Security and Auditing

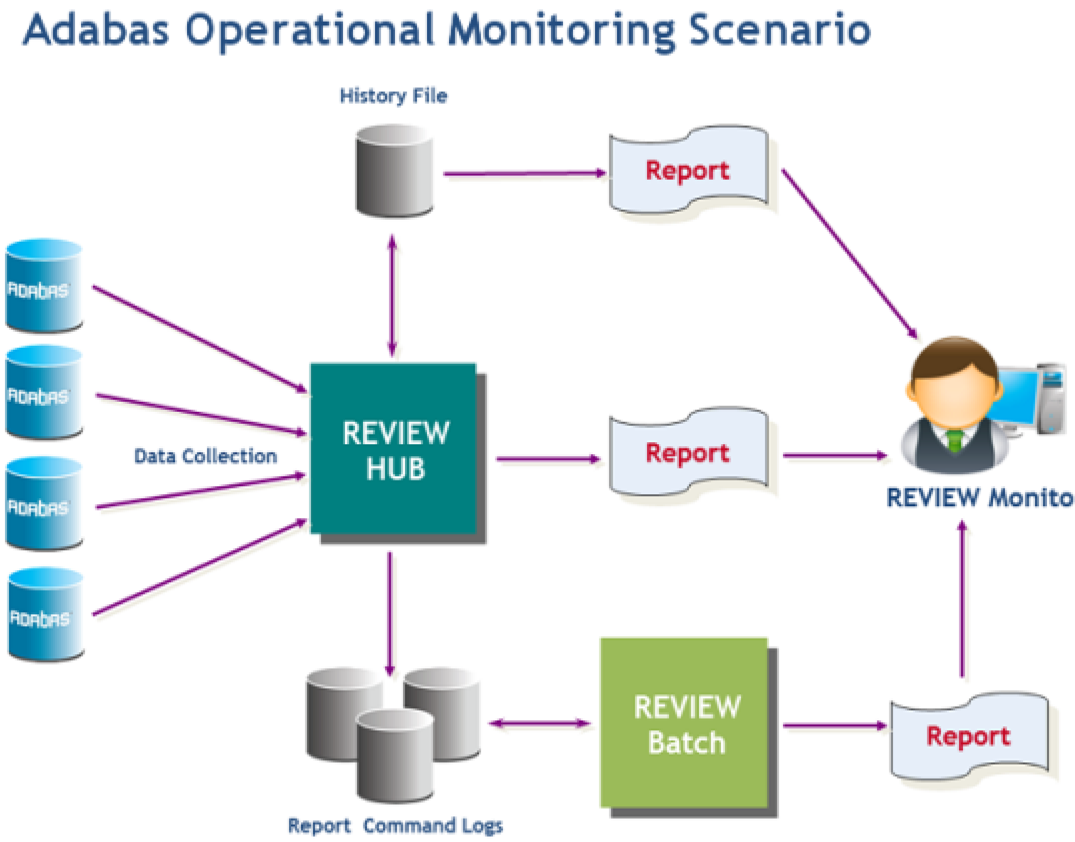
These are described in the following sections.

Adabas Operational Monitoring

Event Analytics for Adabas allows you to monitor Adabas performance in relation to the applications executed against Adabas in the environment. In addition, it is possible to automate the method used to find possible problems with the databases, and indicate the aspects to be improved. The information provided by the solution can be used to improve application programs in order to maximize their performance with minimum resource usage, enabling the optimization of the database for better performance.

This approach enables you to proactively monitor the behavior of Adabas databases, with the following benefits:

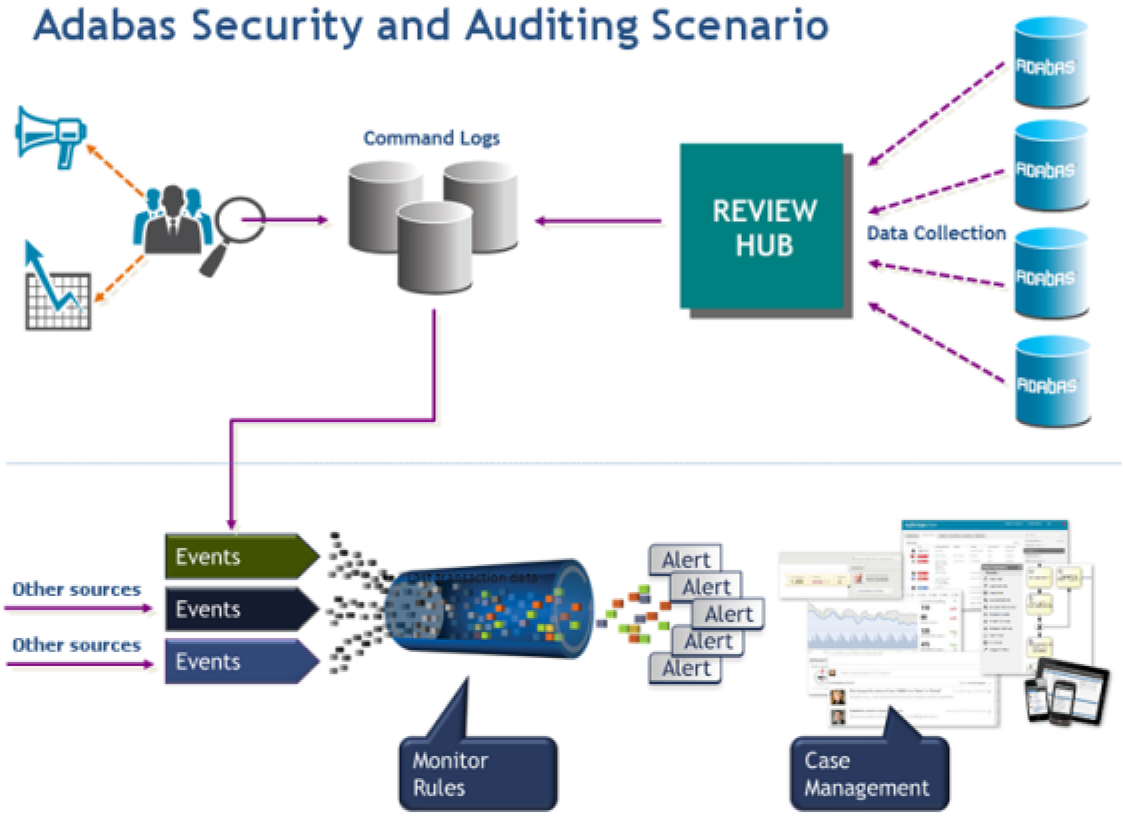
- Quick identification of problems;
- Identification of "offending" applications;
- CPU savings;
- Input for improving applications;
- Reduction in access times to databases;
- Batch window reduction;
- Operational monitoring of Adabas environment.



Adabas Monitoring for Security and Auditing

This approach allows monitoring of events for security and auditing. The solution allows collecting the CLOG records with the purpose of monitoring the defined security tracks, event correlation based on time windows, location, many aggregations such as accumulation, sum, average, minimum, maximum and also the flexibility to combine a mix of different models. For the events captured in the auditing tracks, the solution offers alerting capabilities like monitoring through a console, sending mails, integration with other applications, sending SMS messages, etc. As a summary of the benefits we can mention:

- Monitoring of database events through the prism of security;
- Monitoring of "read-only" (non-update) events in the database;
- Fast fraud identification;
- Proactive detection of problems;
- Greater safety for the environment and for sensitive data;
- Creation of special audit trails relating events from multiple sources;
- Customized and automated sending of alerts.



For "auditing trails" examples we could mention:

- Application "dirty-read" access identification;
- Data "deletes" trigger by users in monitored "locations";
- Online data "updates" out of defined time windows;
- Data "updates" triggered by users not in their "native job location";
- Adabas statistical accumulators, etc.

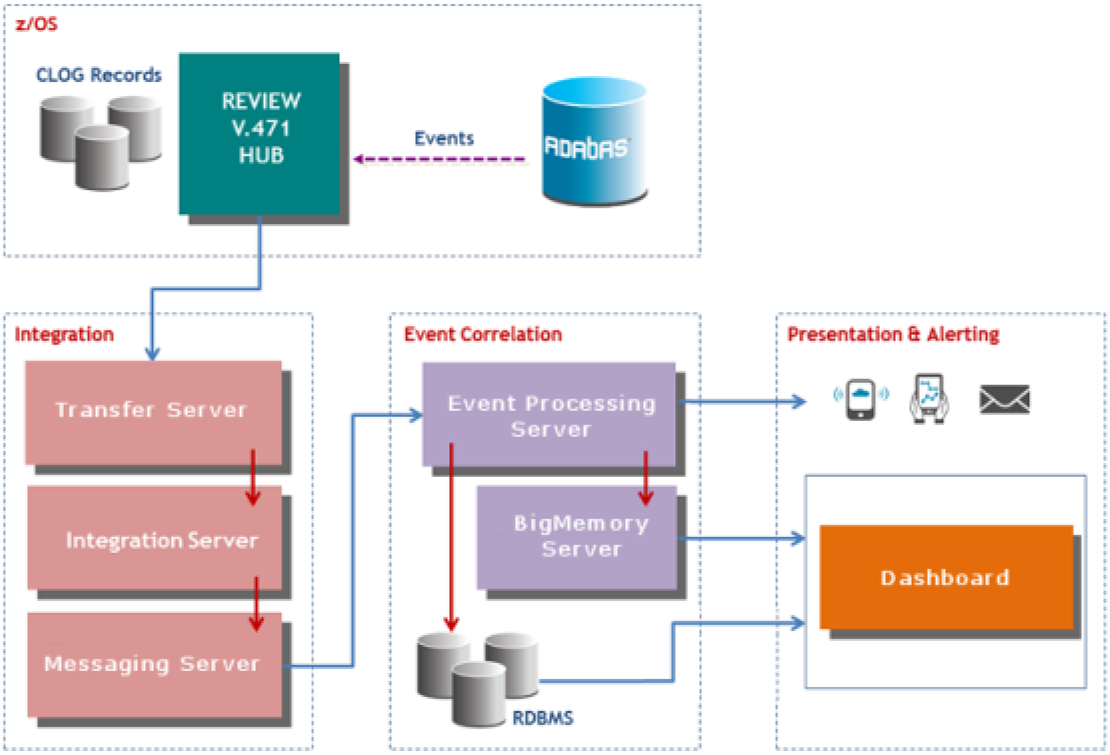
4 Logical and Physical Architecture

Architecture Overview

The following sections describe the logical and physical architecture of the product.

Logical Architecture

Event Monitoring for Adabas: Logical Architecture



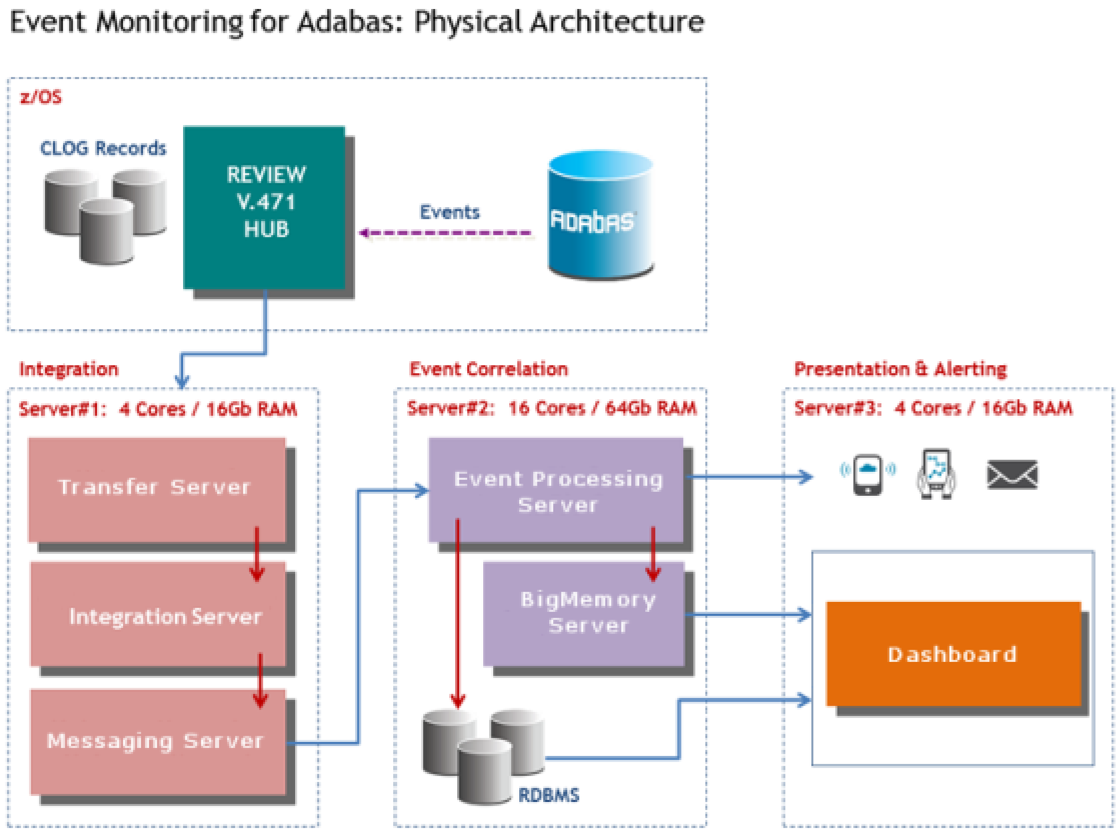
The figure describes the overall processing flow within the solution:

Adabas Review collects Adabas events on z/OS. The events are then accumulated and forwarded via authenticated FTP to the Transfer Server, which triggers a service on the Integration Server that sends each line to the Event Processing Server as an event, through the Messaging Server (JMS).

Inside the Event Processing Server, after retrieving them from the Messaging Server queue, the events are correlated and can trigger alerting functions like SMS and email messages. The Event Processing Server feeds the BigMemory Server caches that hold statistical data from the many audit trails, Adabas usage counters and other information for later dashboarding.

Data visualization is provided by dashboard mashups and features, depending on the way the dashboards will be deployed and the flexibility requirements involved.

Physical Architecture



The figure above describes the servers involved in the solution:

- Mainframe: platform where Adabas and Adabas Review execute.
- Server#1 (Integration): implements the integration functionality, basically receiving Adabas events and forwarding them to the Event Processing Server platform on Server#2.
- Server#2 (Event Correlation): collects the events on Messaging Server JMS queues, performs the correlation in terms of the current auditing trails and triggers accumulation of statistics on RDBMS or alerting via SMS or mail.
- Server#3 (Presentation/Alerting): the platform for holding dashboard servers, where end users will connect for monitoring the dashboards provided.

Some premises:

- Security considerations:

    - File transfer from mainframe to Linux/ UNIX/ Windows will connect to and will be authenticated by the Transfer Server with its own methods. Although this is not a requirement, no data encryption is recommended between mainframe and Linux/ UNIX/ Windows since this would introduce quite a lot of overhead in the process.
    - Dashboard login authentication: this can be implemented with local authentication, LDAP and SSO, depending of the customer requirements and product capabilities.
- Other input sources to the Event Processing Server: the solution can manipulate other sources of information (e.g. log files from other applications) if that makes sense in terms of correlation for the monitored audit tracks. These sources will be grabbed using the existing standard Event Processing Server adapters.
- Alerting: Generation of SMS and email alerts depends on the basic Event Processing Server adapters and external services not listed specifically on this architecture (eg, WebServices for SMS sending).

In terms of platform sizing we can consider:

- Expected load: in the worst case scenario it is expected that Adabas Review would collect ALL CLOG records from production databases from where at least one auditing track is active. This can be many millions of records per day. But in reality not all CLOG records need to be processed, since not all Adabas files are in fact involved in the audit trails. Some filtering techniques can be applied in mainframe side to reduce the amount of records sent to the Event Processing Server.
- Typical sizing:
    - Server#1 (Integration): 4 Cores/16Gb RAM.
    - Server#2 (Event Correlation): 16 Cores/64 Gb RAM.
    - Server#3 (Presentation/Alerting): 4 Cores/16 Gb RAM.

5 Technical Components

For additional information about the technical components in the solution architecture, please refer to the respective product documentation as per the table below:

Capability / technical component	Implemented by product
Transfer Server	webMethods Active Transfer
Integration Server	webMethods Integration Server
Messaging Server	Universal Messaging
Event Processing Server	Apama
BigMemory Server	BigMemory
Dashboard	MashZone, Presto