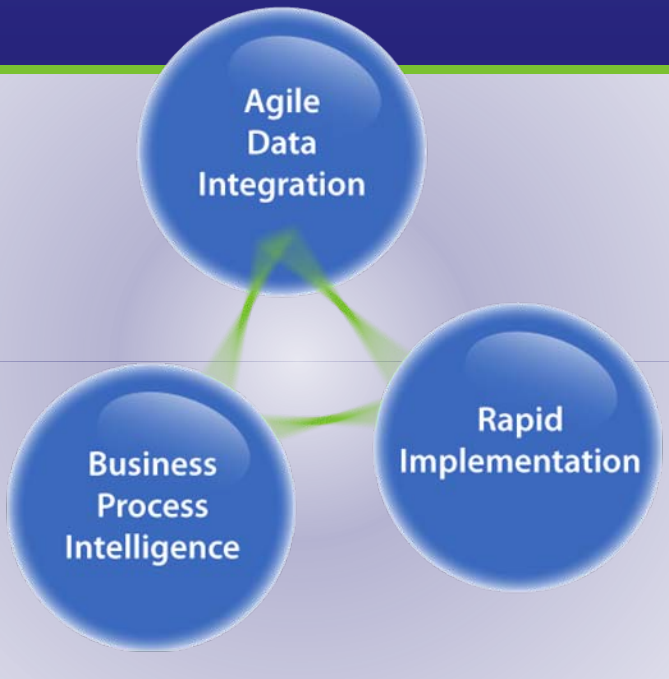
# Bringing Business Intelligence to Business Operations

*An Architectural View*

March, 2009




Altosoft

## Summary

Traditional business intelligence (BI) architercture represents a tried-and-true mechanism for delivering strategic-level insight and decision support. But it is not designed to address the requirements of organizations that are looking to add business intelligence directly into live, running operational processes. In this environment, integrating BI directly into business processes can drive business process improvements that make the difference between organizational success and failure. But effective operational business intelligence (OBI) presents unique challenges for solution architecture. This paper examines how traditional BI-based architectural approaches and business activity monitoring (BAM) architectures fail to adequately address the complex and unique requirements of OBI. The paper then presents a new architectural approach pioneered by Altosoft, which supplements and enhances existing BI investments, enabling organizations to address the challenges of OBI quickly and cost-effectively.

## Introduction

Better business decisions will, in the aggregate, result in superior strategic business results. Traditional business intelligence tools, considered a core part of most organizations' IT infrastructure, are designed to support decision-making at a strategic level where planning and analysis can take weeks or even months (for example, sales forecasting, campaign management, or new product planning).

Of course, many business processes don't take weeks or months to reach fruition. In fact, the cornerstone of success for many businesses hinges on their ability to effectively manage and optimize high volume, short-cycle activities. Competitive pressure has exacerbated the operational challenge. Orders that once took a week to fulfill must now be processed in days. Manufacturing operations which used to maintain weeks worth of part inventory on site now rely on daily replenishment shipments. Call center operations are tasked with reducing customer wait times while simultaneously improving up-sell and cross-sell performance.

As the speed of business has accelerated, demand for a new variant of BI, operational business intelligence (OBI), has emerged. OBI is designed to support decision-making within tactical, operational business processes, like inventory replenishment or customer call center interaction. OBI comes into play

where the window for information gathering, analysis, decision, and response is measured in minutes or hours, not days, weeks, or months.

The following scenarios describe the type of operational processes where OBI is essential to enabling business success:

**Scenario 1:** Inventory Management and Upside Demand Flexibility for Retail Operations

> A retailer selling DVDs places an initial DVD order for a new movie release based on box office receipts. If demand for the product exceeds expectations, the retailer must re-order from the manufacturer; but all retailers replenish from the same manufacturer based on a finite supply of DVDs. The manufacturer processes orders on a first-come, first-served basis, so if demand is high and DVD supplies run out the retailer could lose significant sales opportunities. On the other hand, if demand is lower than forecast and the initial DVD order is too large, the retailer could be stuck with an unsold inventory of DVDs, since the manufacturer will only allow a limited number of returns on a first-come, first-served basis. This is a fairly common order optimization scenario for DVD retailers like Target or Best Buy, which can have a daily sales impact of millions of dollars.

**Scenario 2:** Order Fulfillment for a Supply Company under Service Level Agreement

> An electronics supply company maintains and ships products from multiple warehouses, and is subject to service level agreements with its customers. Typically the SLAs promise same day shipping for orders received before 3:00 pm. If a surge in orders occurs, the company could be in jeopardy of not maintaining its SLAs. The problem can be avoided by always maintaining extra warehouse labor (pickers) on site, but this is an expensive strategy. It is possible to spot increase the labor force on a given day, but enough advance warning of an order surge is required to bring in the extra labor force in to make an impact before the 3:00 pm deadline.

Unfortunately, traditional approaches to BI are not designed to support OBI scenarios. Operational business intelligence presents unique challenges in terms of solution architecture. This paper identifies those unique requirements and examines how traditional BI-based architectural approaches and business activity monitoring (BAM) architectures fail to adequately address the complexities of OBI. The paper then presents a new architectural approach pioneered by Altosoft, which supplements and enhances existing BI investments, enabling organizations to address the challenges of OBI quickly and cost-effectively.

*"The demand for operational business intelligence (BI) is increasing as companies look for ways to infuse context-aware reports and analysis into the workflow of decision-makers…. There is clearly a need out there for BI applications that address specific business needs at the operational level within companies to help deal with issues such as order fulfillment and credit card fraud."*

- **Forrester**
  Keith Gile with Connie Moore and Eric Kim
  *Operational BI Heats Up with an Infusion of Context*, June 5, 2006

## Requirements for Operational Business Intelligence Solutions

Ultimately, the effectiveness of an operational business intelligence solution hinges on the technology's ability to support an organization's analytical, decision-making, and response processes, specifically:

- The time it takes to identify and aggregate relevant data;
- The time it takes to prepare and deliver analytics pertaining to this data (for example, KPIs or other contextual information) and deliver the analytics to a decision-maker;
- The time it takes for a decision-maker to review, analyze, and determine a course of action related to the information; and finally
- The time it takes an organization to respond to implement that course of action.[*]

OBI must support this entire cycle of information-decision-response. In a sense, operational business intelligence is decision support for time-sensitive business processes. In many cases, these are also high volume, high transaction processes. Data latency must be low and transactional throughput must be high. Operational decision-makers require real-time data about business conditions in order to make optimal operational decisions and to respond immediately and effectively to operational problems and exceptions.

The unique requirements of operational business intelligence include:

[*] Adapted from material by business intelligence industry analyst Richard Hackathorn of Bolder Technology, Inc.

***Historical Data Analysis.*** An OBI solution must be able to analyze historical data in the form of KPI metrics, and deliver these metrics through an easy-to-use dashboard interface for visualization and dimensional analysis. In this context, KPIs measure business history. Users can interactively slice and dice this historical data, enabling an historical understanding of business performance. This can then be compared to real-time, or right-time, operational data.

***Real-Time Data Monitoring.*** The solution must be able to monitor and analyze current, intraday business performance from both relational databases and real-time events. Event-based data is by definition real-time. Access to data contained in relational data sources must be extremely low latency (low enoght to meet business requirements for responsiveness) and at the same time impose a minimum query load on source data systems. Operational data must also support access to individual transaction data, facilitating drill-down to identify issues and resolve problems on an instance-by-instance basis. For example, bringing together historical and real-time data, a KPI could show that current inventory availability was normal by historical standards but insufficient to meet current in-process orders; analysis of individual orders might reveal that a data entry error resulted in an order being processed for 10000 units instead of the intended 100 units.

***Understanding of Process Context.*** An effective OBI solution must understand process context. Process analytics are critical to effective operational analysis and incident response. For example, without an understanding of process context, an OBI solution in a supply chain scenario wouldn't be able to deliver fundamental KPIs such as "Percentage of Perfect Orders", which indicates the number of order processes completed without exception or incident. Process context is also critical for identifying issues on at the level of an individual process instance. For example, an effective OBI technology needs to be able to detect when an individual running process has gone awry – for example, when a large trade has failed to receive confirmation or when an order to a high priority customer has been delayed.

***Alerting and Incident Management.*** Operational BI should provide a robust capability to deliver alerts based on user-defined KPIs to a variety of output channels, including email, dashboards, portal, SMS message, and more. Moreover, the system should support flexible, workflow-driven management of business incidents to ensure timely resolution of potential issues and problems.

***Simple and Cost-Effective Implementation.*** Solutions that require significant custom coding, major process documentation programs, or intensive data engineering exercises, plus the ongoing cost of associated maintenance, consume enormous financial and human resources. An OBI solution should be relatively easy to implement, cost-effective to maintain, and deliver rapid return on investment. From a technology standpoint, an OBI solution should integrate easily with other technologies, both in terms of data sources and acquisition, as well as delivery mechanism (via portal, desktop productivity tools,

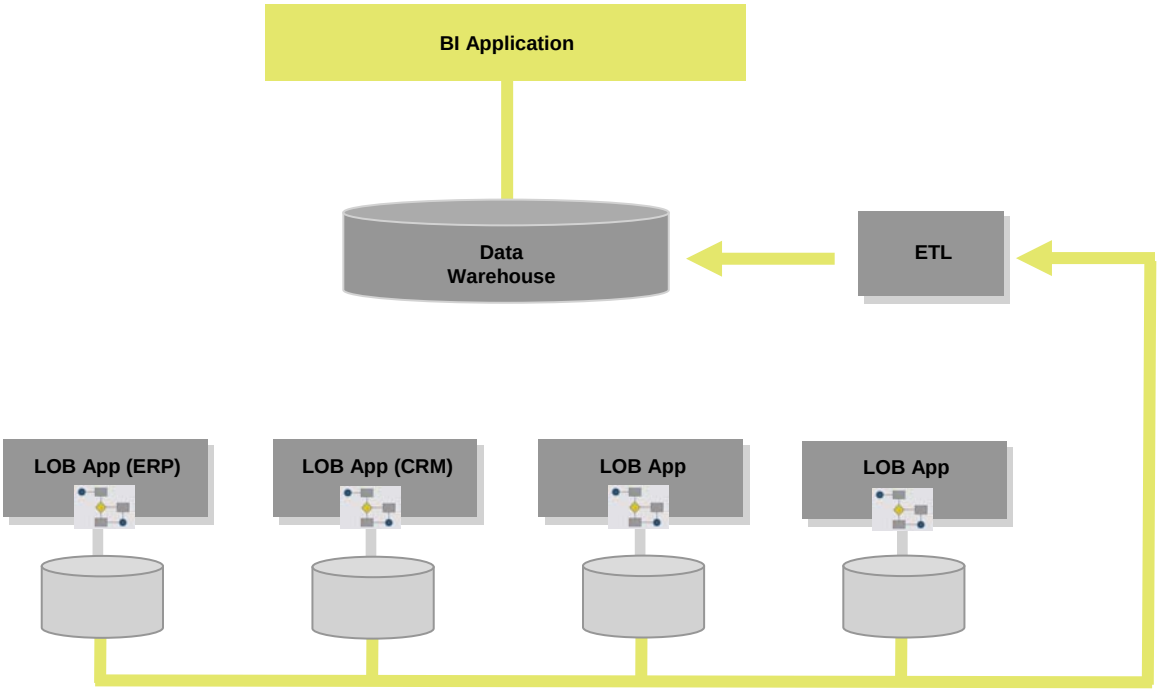
dashboard, or other model), without substantial custom coding or integration work, or reliance on third party integration tools.

The various major architectural approaches to operational business intelligence will now be analyzed in the context of these critical OBI requirements and the performance of each approach will be considered in the context of the three OBI use cases described above.

## Traditional Business Intelligence Architecture

The fundamental basis of traditional business intelligence solutions from vendors like SAP/Business Objects, IBM/Cognos, or Oracle/Hyperion, is the view of the world as a collection of records. The records are collected over a period of time; then the data is analyzed. The techniques used in BI products, are capable of efficiently processing huge amounts of records. However this efficiency comes at a price. BI tools require the data to be “prepared”, usually in the form of the data warehouse or a “cube”. The construction of the cube requires a lot of expensive labor and long project cycle. Any changes in the data sources or reporting requirements result in new implementation project. Even when the cube is implemented, moving large amount of data creates significant latency and, in some case, just plain impossible.

This diagram shows a traditional architectural approach to BI:



Line of business applications such as ERP (financials, human resources, manufacturing, etc.), CRM, and others are managing portions of enterprise business processes. BI requires extraction of data directly from the data stores of these systems. The data is then aggregated, cleansed, transformed, and normalized, often using third party transformation tools. This enables like-to-like comparison and analysis of data which might otherwise differ in how it is recorded in the underlying enterprise applications.

Once the data is aggregated and transformed, it is loaded into a data warehouse, which is optimized for query and response along certain parameters or use cases (i.e. the data is pre-dimensioned into "cubes"). If the use cases (expected queries and analysis) change, in many cases the cubes will have to be re-calculated. Extraction, transformation, and loading are conducted in batch mode, as opposed to as a real-time data stream. Finally, the BI tool acts as the query engine into the data warehouse, supporting everything from pre-defined reports to completely ad-hoc analysis. The latter capability is typically restricted to a limited set of power users.

While proven, this architectural approach faces significant limitations when confronted with the unique requirements of OBI.

***Historical Data Analysis.*** Traditional BI approaches are designed to deliver powerful query and analysis capability for historical data.

***Real-Time Data Monitoring.*** In cases where real-time, low latency data is required, traditional BI architectures fail. With existing BI solutions it can take days or sometimes even weeks to integrate, contextualize, and present operational information to business managers.[*] In fact, a recent survey from the Gartner group showed that 95% of all BI applications have a data latency of one day or more.[**]

***Understanding of Process Context.*** BI solutions have no native understanding of business process context. They are unable to deliver metrics like "Average Time to Complete a Task" or "Percentage of

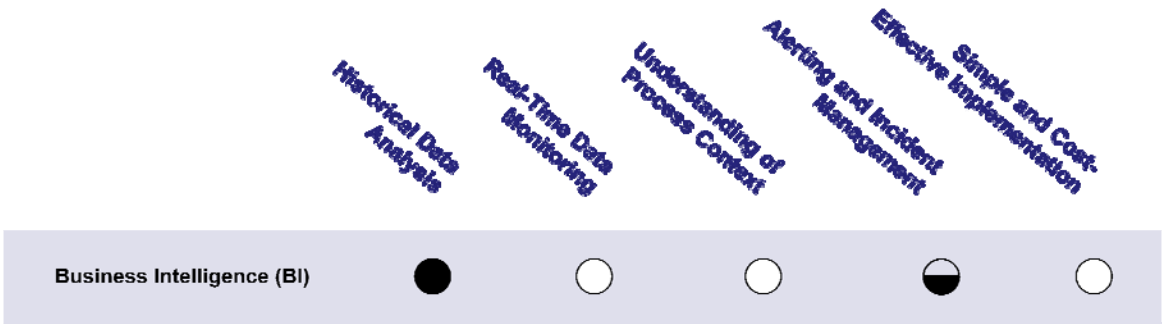
[*] There are additional techniques for reducing latency, each with certain drawbacks. One is to build a virtual cube using integration techniques. Data is pulled from operational systems using federated queries when a report is required. In this context, data isn't moved or copied from source systems, so there is no need for a large data store. The downside of this technique is that is can be only used for small data sets (due to performance limitations of moving data) and data acquired via integration can only be minimally transformed. In addition, this approach doesn't scale well to support large numbers of users because it imposes significant query load on source systems. Another approach is change data capture (CDC). This technique facilitates continuous updates throughout the day by only passing changes or "deltas" from source data to the BI system. The downside to this technique is that it can negatively impact the performance of enterprise systems, which are forced to do double duty delivering data to the BI system and supporting operational activity. Additionally, if source data is complex or multiple data sources are involved, implementation and maintenance of this type of solution can become extremely challenging.

[**] *Survey Shows BI Users Want Fresher Data*, Bill Gassman, Kurt Schlegel, Mark A. Beyer, September 2006.

Perfect Orders", particularly if transactional data is not robustly time stamped in underlying enterprise systems and cubes are not pre-configured to provide analysis based on time stamped data.

***Alerting and Incident Management.*** BI systems typically deliver reports through a variety of different media. Alerts can be delivered as well, although they are not timely due to the high-latency characteristics of BI-based analysis. Most BI approaches do not natively include workflow-driven incident response systems, largely because the intended use case of BI isn't the type of operational decision support that requires this type of incident response capability.

***Simple and Cost-Effective Implementation.*** BI is typically expensive to implement and maintain. It requires significant data integration work, major data engineering efforts to create a data warehouse, the hardware associated with maintaining a high performance environment for that data warehouse, and all the associated costs and complexity of maintaining ongoing batch updates to the system. As a result, implementation is generally both expensive and time consuming. Even more modern BI architectures employing technology such as in-memory analytics encounter these same challenges. If data from multiple sources must be analyzed (which is the norm, rather than the exception), significant, time-consuming up front data engineering is inevitably required.




We now consider how BI stacks up when applied to two OBI scenarios:

*BI versus Scenario 1: Inventory Management and Upside Demand Flexibility for Retail Operations*

Applying a traditional BI architecture to solve this operational retail problem reveals significant shortcomings. First, the data latency issues of BI present a practically insurmountable challenge, particularly if source data resides in multiple systems (logistics, point of sale, inventory), as it almost inevitably will. If data latency is too high (e.g. the twenty four hour or longer latency typically seen with 95% of BI implementations), then the DVD sales data will not be available for analysis until it is too late. The retail operation will essentially be flying blind with respect to its reorder or return strategy. So if the

new DVD release hits the shelf on Tuesday and demand is running exceptionally low, the retailer won't have the data ready for analysis in a cube for at least twenty four hours. Add to that the analysis time and the time it takes to initiate a re-order process and the window of opportunity to beat the competition to apply for a return authorization could be closed. The inability of BI tools to adequately address this sort of dynamic demand-response scenario is why so many unsold DVDs make it to the bargain bins of major retailers nationwide.

*BI versus Scenario 2: Order Fulfillment for a Supply Company under Service Level Agreement*

Similar problems arise when BI is applied to the warehouse management problem in scenario 2. Again, latency is the major issue. By the time the BI system has the day's order data loaded into cubes for analysis, the day is over and the opportunity to add staff to hit promised SLAs is long passed. Even if the latency issues could be addressed, the BI solution lacks the process awareness necessary to address the staff resource problem. In fact, under the guidelines of SCOR (Supply Chain Operations Reference model) Level 3, which provides guidelines for planning and setting goals for supply chain improvements, BI tools are explicitly not used for operations.

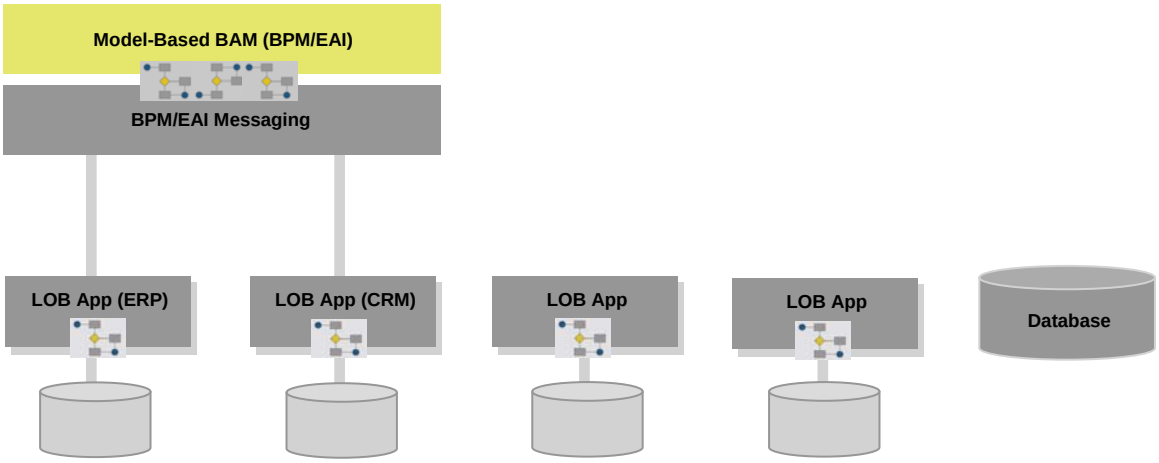
Traditional BI approaches have been on the market for a long time. While this technology delivers good results in specific applications (such as data mining, for example), it fails in the area of operations where the low latency and process analytics are required.

## Model-Based Business Activity Monitoring (BAM)

Unlike traditional BI approaches which view the world as a collection of records, both model-based and event-based BAM products view the world as a flow of events. Events are captured and processed in real time, eliminating reporting latency and the need to move data to a central warehouse or cube. There are two major variants to the event-based BAM approach, model-based and direct event-based. Model-based BAM vendors such as Tibco or webMethods (typically coming from the integration or business process management (BPM) space) require a user to completely define the business processes and map that process to events occurring inside their environment before realizing any reporting or analytical value from the BAM solution. Basically, the model-based BAM solution must "own" and orchestrate the business process before it can monitor it. Model-based BAM can deliver powerful, real-time event analytics with a complete understanding of process context. However, the drawback of the architecture is that is requires complete up-front definition of a business process and integration with all third party systems involved in that process. This is a significant obstacle for many customers as very few companies have completely mapped their business processes across multiple integration environments

and LOB applications. In rare situations where process mapping has occurred, it is typically fragmented and does not cover a business process holistically.

This diagram shows a typical model-based BAM architecture of the type typically offered by EAI and BPM vendors:




Again, line of business applications such as ERP (financials, human resources, manufacturing, etc.), CRM, and others are managing portions of enterprise business processes. Additional data relevant to a process (e.g. customer info) may be contained in independent databases. In this diagram, a messaging layer has been added to integrate the ERP and CRM applications.

Model-based BAM solutions are typically tied to their messaging layer. So for instance, a Tibco BAM solution must rely on a Tibco messaging and integration layer for connectivity into line-of-business applications. The BAM solution does not draw data or events from enterprise applications where that messaging layer does not extend.

Model-based BAM requires complete, up-front process definition. Once defined, events occurring in LOB applications are mapped directly to that process. Once processes are mapped and integrated with LOB applications, the BAM solution can monitor process activity and deliver analytics and reporting for all data and events associated with that process. In the case of BPM solutions, the process itself is the mechanism controlling the movement of data between the various enterprise applications it is integrated with.

This architecture delivers robust process analytics and context. However, the downside is that process models must be fully and exhaustively defined in advance. Process modeling exercises can be extremely time-consuming and complex. For a moderately complex process that spans multiple functional areas

and back end systems in an organization, they typically involve a cross functional team including IT and staff from various functional areas. Each area brings some fragmented knowledge of the "correct" business process; these must somehow be rationalized into one visual model definition that everyone can agree on. It's no surprise that process definition exercises can take months or even years, and in some cases end up failing completely.

As a result, in spite of its power, this architectural approach also faces significant challenges in OBI applications.

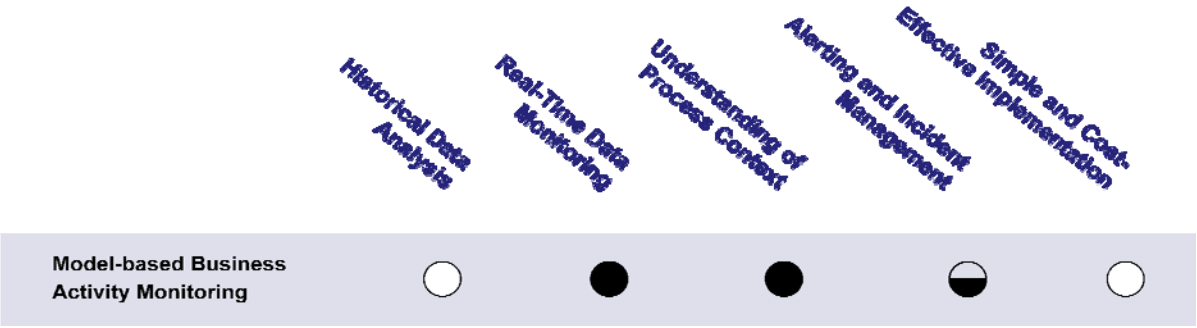
***Historical Data Analysis.*** Model-based BAM is focused on real-time event monitoring. There is no integration to historical record data.

***Real-Time Data Monitoring.*** Model-based BAM is well suited for real-time event monitoring. However, it only monitors events from systems that it has connectivity into, and connectivity is typically limited to the messaging platform that the particular BAM solution leverages. If the platform is only connected to systems which span a portion of the overall enterprise process, only a portion of that process can be analyzed.

***Understanding of Process Context.*** Model-based BAM has an excellent understanding of process context and can deliver complete and robust process analytics for all events and activities that are orchestrated by or attached to the modeled process. The downside to this powerful capability is the requirement to fully and completely model every business process in advance – a proposition that in most cases is expensive and time consuming.

***Alerting and Incident Management.*** Model-based BAM typically monitors real-time events by calculating KPIs which are then measured versus historical or target values. These solutions then deliver alerts via a variety of media if operations fall outside defined ranges. Integrated incident or exception management can be deployed leveraging integrated BPM or workflow capability. This may require some custom development or at least additional exception process definition, however. Unlike BI and direct event-based BAM, model-based BAM can also monitor individual process instances and detect if there is a problem or deviation from the normal flow of business operations.

***Simple and Cost-Effective Implementation.*** Like BI, model-based BAM is typically expensive to implement and maintain. As discussed, process definition is difficult and time consuming. In addition, data mapping between process and underlying systems can be complex and challenging to maintain.




We now consider how model-based BAM stacks up when applied to the two OBI scenarios:

*Model-based BAM versus Scenario 1: Inventory Management and Upside Demand Flexibility for Retail Operations*

A model-based BAM approach would require complete definition of the demand-response process. This process definition would require mapping a process that involved integration with multiple transaction systems including logistics, inventory management, point of sale, and more. Assuming this challenging implementation could be accomplished, DVD sales and inventory levels could be monitored in real time. However, it would not be able to automatically adjust KPIs based on seasonality or other exogenous historical factors. It would not be able to factor in order lead time, inventory buffer, or other variables to optimize timing of the initiation of a reorder or return process. Because KPIs cannot be calibrated to historical demand factors, a KPI-based trigger to reorder or return DVDs effectively becomes useless. In a situation where there is abnormally high demand, a rule could trigger a reorder or return if inventory fell below a certain predefined level; but that level could not easily be made to vary intelligently depending on how variable outside factors such as how long it took to receive shipment of new DVDs, whether the holiday season (and increased demand) was approaching, etc. Effectively, the solution would require a human operator to monitor the system continuously and make a decision when to reorder or return – a scenario that is not reasonable across hundreds or thousands of individual SKUs. As a result, returns or reorders could perhaps be initiated in a timelier manner than with a BI solution, but the result would still be far from optimal.

*Model-based BAM versus Scenario 2: Order Fulfillment for a Supply Company under Service Level Agreement*

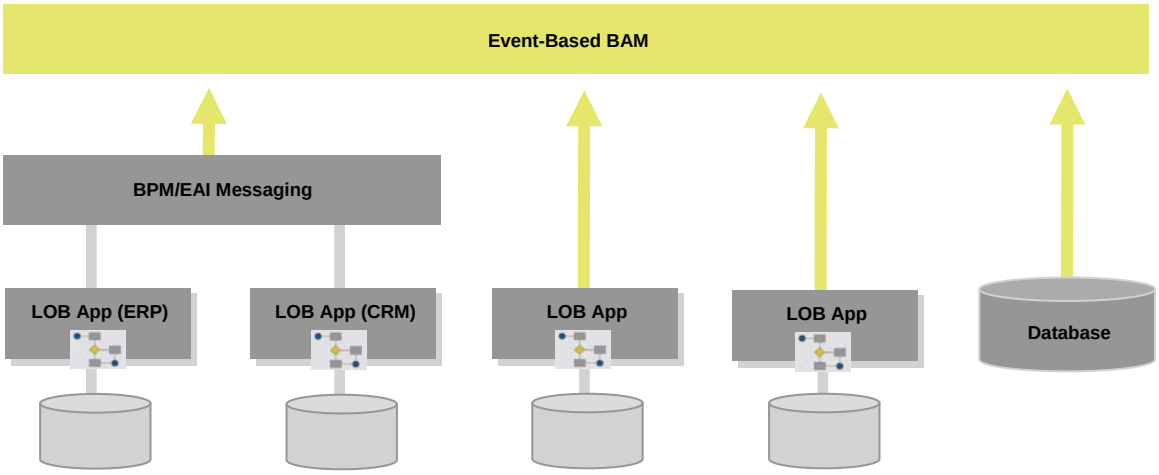
Again, similar limitations arise when model-based BAM is applied to the warehouse management problem in scenario 2. First, the order fulfillment process must be modeled, and integrated with external systems, so the process is “owned” by the model-based BAM solution. Assuming this can be

accomplished, the BAM solution can identify an increased flow of orders and warn that a problem with shipping all those orders may exist if orders suddenly rise; but the warning would come without the historical perspective that would enable management to determine how much additional staffing might be required to fulfill the predicted surge in orders by close of business.

## Direct Event-Based Business Activity Monitoring (BAM)

Like model-based BAM, direct event-based BAM solutions (such as those offered by vendors like CognosNOW!, Systar, or Aleri/Coral8) view the world as a flow of events rather than a collection of records. However, direct event-based BAM solutions completely ignore business processes. They capture and process events from other enterprise systems in real time, and calculate KPIs directly (as opposed to mapping events to processes, which then become the basis for KPI calculations). This approach completely eliminates the need for exhaustive process modeling and, in theory, significantly reduces implementation time as a result. However, attractive as it might seem, the lack of the business process context that results causes major weaknesses when direct event-based BAM is measured against OBI requirements.

This diagram shows a typical direct event-based BAM architecture:




Event-based BAM captures events directly from underlying systems or through integration tools such as enterprise service buses if applicable. The event flows are completely disassociated from any process context or information. As noted above, KPIs are calculated directly from events; real-time events and KPIs are delivered directly into a dashboard for monitoring and limited analysis. Event-based BAM solutions can scale to support very high transaction volumes in underlying systems.

As a result of its relative simplicity, this architectural approach suffers from significant limitations.

***Historical Data Analysis.*** There is no reporting or analysis of any historical data.

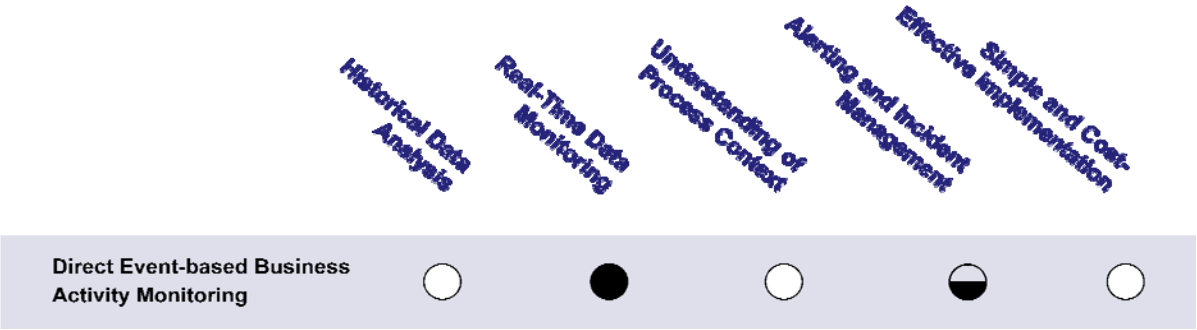
***Real-Time Data Monitoring.*** Real-time events are supported. Direct event-based BAM can calculate KPIs based off of real-time data from underlying systems, but sophisticated KPI definition can require significant customization. Specifically, KPIs are defined as a function of captured events. Since the captured events are specific for underlying platforms, LOB applications, and business practices, KPI definition is different for every customer and every business process. As a result, KPIs must be manually re-coded for each and every direct event-based BAM implementation.

***Understanding of Process Context.*** Direct event-based BAM solutions have no understanding of process context. By creating KPI definitions which leverage time stamps on events or by using other workarounds, metrics for process performance can indirectly created. However, custom coding and custom calculations for process-related KPIs are even more complex to develop than normal KPIs, since they rely on an understanding of the sequence of events in the underlying business process (for example, *Order Processing Time*).

***Alerting and Incident Management.*** Direct event-based BAM typically monitors real-time events by calculating KPIs which are then measured versus historical or target values. These solutions then deliver alerts via a variety of media if operations fall outside defined ranges. However, since it has no process awareness, direct event-based BAM cannot monitor individual process instances and detect if there is a problem or deviation from the normal flow of business operations. As a result, if an individual process is stalled, skips a critical step, or otherwise encounters a problem, the BAM solution cannot detect it and the error may fall through the cracks as it is averaged into a reported KPI figure that could include event data from hundreds or thousands of individual process instances. Finally, most direct event-based BAM solutions have only primitive incident management capability, or nothing at all.

***Simple and Cost-Effective Implementation.*** While architecturally direct event-based BAM Is probably the simplest of the various approaches considered here, it still requires significant implementation time and expense. Very simple event-based BAM can be delivered relatively quickly; but as soon as custom KPIs are required project costs rise considerably. Some of this has to do with the fact that event –based data sources, unlike standard SQL-based relational databases, lack a standard interface. This means custom adapter development. Also, since custom KPIs are necessary for every enterprise that has unique business events and unique process flows (i.e. every enterprise), it is almost inevitable that direct event-based BAM projects, which start off looking simple in proof-of-concept phase, rapidly become an

extensive and costly coding exercise. In fact, most direct event-based BAM implementations end up being comparable to BI implementations since so much custom coding and integration work is required.




We now consider how direct event-based BAM stacks up when applied to the two OBI scenarios:

*Direct event-based BAM versus Scenario 1: Inventory Management and Upside Demand Flexibility for Retail Operations*

A direct event-based BAM approach would enable KPIs for DVD sales and inventory levels to be calculated and monitored in real time. However, like model-based BAM, the solution would not effectively integrate any visibility into historical inventory data. As a result, it would suffer from the same practical deficiencies as model-based BAM (see above). In fact, the solution would be even more limited than the model-based BAM solution, since it would be completely unable to deliver any process-based KPIs which might be required (e.g. average time to initiate and receive confirmation of a return authorization from a particular supplier) or identify deviations from norms in individual process instances (for example, if an individual RMA request has been issued but for some unknown reason the supplier has not sent a confirm within the normal time it usually takes for that supplier to respond).

*Direct event-based BAM versus Scenario 2: Order Fulfillment for a Supply Company under Service Level Agreement*

A direct event-based BAM solution, like a model-based approach, can identify an increased flow of orders and warn that a problem with shipping all those orders may exist if orders suddenly rise; but without the perspective provided by historical analysis, users of the system would be unable to determine how much additional staffing might be required to fulfill the predicted surge in orders by close of business.

## Hybrid Solutions

Given the failure of any single architecture to effectively support the requirements of operational business intelligence, it is not surprising that many companies are turning to hybrid architectures out of necessity. These hybrids typically combine a traditional BI environment for historical data analysis with a BAM environment for real-time data and event monitoring. A hybrid approach certainly can be more effective than a pure BI- or BAM-based approach, but it still cannot effectively address all the requirements of OBI.

***Historical Data Analysis and Real-time Monitoring.*** A hybrid approach can deliver historical reporting via a BI component and real-time reporting via a BAM component. However, this information cannot be synthesized or delivered via a single dashboard presentation without significant custom coding since a hybrid by definition requires integration of multiple solutions.

***Understanding of Process Context.*** A hybrid can deliver process analytics and context if it leverages a model-based BAM technology.

***Alerting and Incident Management.*** A hybrid can conceivably leverage the workflow environment of a BAM vendor to deliver exception management with some additional process definition required. Additional integration work would be necessary to trigger exception or incident management from historical data analysis.

***Simple and Cost-Effective Implementation.*** Neither BI nor BAM solutions can be cost effectively implemented as standalones. Implementing the two together as a hybrid more than doubles the cost and complexity, since each must be individually implemented and then integration must occur. The cost and resources involved here are prohibitive except for the most mission-critical applications.

In short, no existing approach or combination of approaches effectively addresses the challenges of operational BI. A new approach is required.

## A Better Approach: Altosoft InsightBI and MetricsMart

Altosoft has developed a very different approach to operational business intelligence. This approach leverages a number of patent-pending technologies and methods to support the full range of OBI requirements.

The implementation process for an Altosoft-based OBI solutions begins with MetricsMart. Altosoft's MetricsMart solution fundamentally changes the way any company approaches the problem of data

aggregation. Instead of writing volumes of ETL scripting and SQL to first centralize all data into a data warehouse or mart, and then following this up with more coding, scripting, and/or SQL to derive KPIs from this raw data, MetricsMart delivers the final KPI metrics in a single step – with ZERO coding. The result is that all KPIs are processed and stored in a third party repository (generally SQL Server or Oracle) as fully dimensioned tables that can be easily accessed by Insight's web services interface or directly via any other off the shelf SQL reporting or analysis tools. Thanks to MetricsMart's highly-graphical, intuitive UI, Altosoft can integrate data from multiple data sources and calculate and store multiple KPIs often in as little as a few hours.

MetricsMart makes data integration flexible and response to the needs of business users. It enables *dynamic latency selection*, so that metrics are delivered to business users when they need them, rather than at the whim of some batch refresh cycle. It can handle *real-time* environments. And it's easy to configure and maintain. Rather than requiring custom code or scripting, it's 100% codeless.

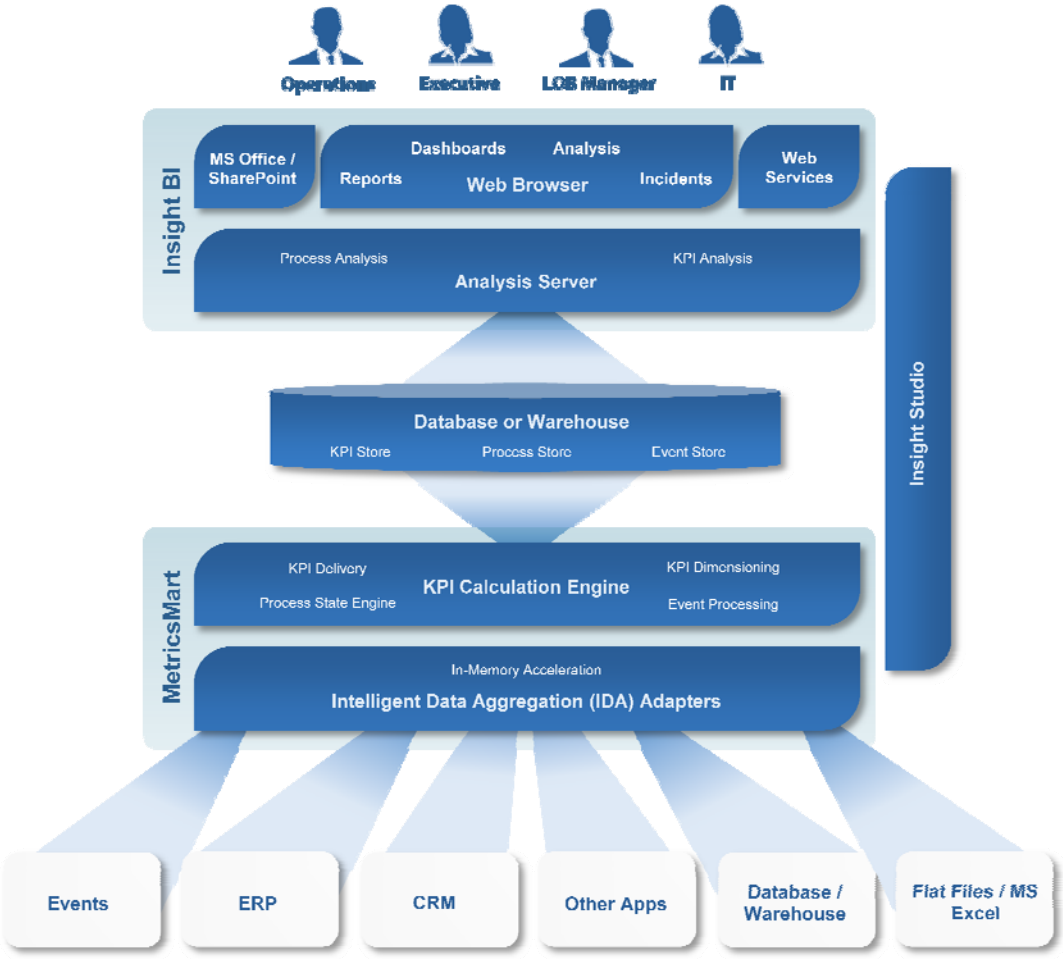
MetricsMart is fundamentally different from competitive BI- and BAM-centric approaches because:

- Rather than moving mass amounts of data in batch to a data warehouse, MetricsMart intelligently moves the data needed to deliver the KPIs your business needs. This approach, called Intelligent Data Aggregation (IDA), makes MetricsMart extremely efficient.
- MetricsMart generates pre-calculated, on-demand KPIs. That means that you tell it when you need a particular KPI, and it will deliver the calculated values of that KPI into a database or warehouse for you automatically.
- MetricsMart is business process aware. MetricsMart's optional process state engine, ProcessMart, tracks every business process and enables you to view process performance and process quality.
- MetricsMart uses high performance in-memory processing to perform the most complex, time-consuming calculation, such as outer joins. That means low latency data and minimal query load on source systems.
- MetricsMart is easy to use. It's configured using the Insight Studio tool. No code or script is required.
- MetricsMart will work with any data source. RDBMSs, warehouses, flat files, Excel, events, proprietary formats.

Once MetricsMart has been configured, and optimized KPI metrics are available for users in an open database format, InsightBI takes over. With InsightBI, users interact with Altosoft via intuitive dashboards to monitor real-time and historical enterprise data, as well as accurate prediction of short-term operational activities. Personalized dashboards for business monitoring can quickly and codelessly be configured by

business users. Altosoft analytics can also be delivered via MS Office or an enterprise portal. The solution also supports end-user driven reporting capabilities.

This diagram shows Altosoft's architecture, including MetricsMart and InsightBI:




Altosoft translates all data into business events, regardless of their original source (event-based or otherwise); as a result, for database connectivity Altosoft deploys a database adapter that determines what activity within the database constitutes a new business event. Only event data that is relevant to calculating KPIs is captured.

With Altosoft's optional ProcessMart module, Altosoft delivers full capability for *business process intelligence* -- the science of monitoring and analyzing business performance through the lens of business processes. With ProcessMart, captured event data is correlated to processes managed by a unique process state engine, Altosoft's Process Intelligence Engine. Process definition and event correlation is easy (orders of magnitude easier than designing processes for a BPM system) because only the key

milestones that business users need to understand are modeled. Unlike BPM-centric approaches, there is no exhaustive elaboration of business processes, or need to address the complexities inherent in building a complete transactional system.

The Process Intelligence Engine manages the state and the events associated with every in-flight business process. This enables Altosoft Insight to calculate KPIs for event data with full awareness of business process context, delivering process analytics for processes that span multiple enterprise applications. KPIs are calculated at defined intervals and stored in an efficient KPI storage database. Because only historical KPIs relevant to operational processes are being calculated and stored, performance and throughput of the system is extremely high. The system is designed to deliver powerful analytics and predictive capability in the most complex business environments, where scalability and ability to support high transaction volumes are prerequisites.

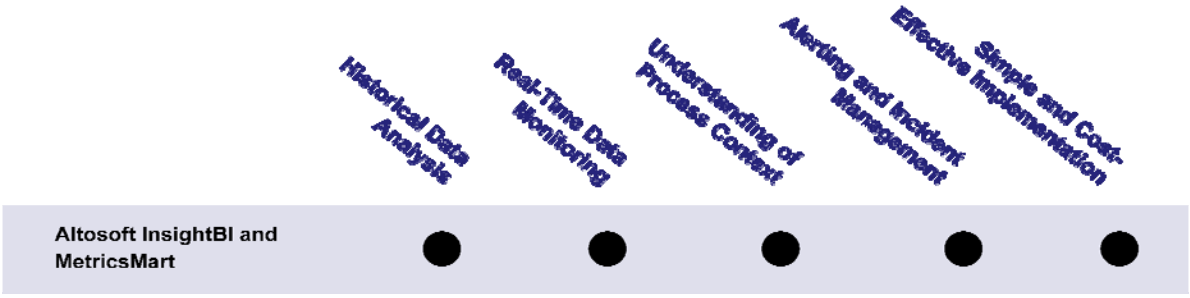
This architecture enables full support for OBI requirements.

***Historical and Real-Time Data Analysis.*** As discussed above, Altosoft delivers monitoring of both real-time events and analysis of historical KPIs.

***Understanding of Process Context.*** Altosoft's ProcessMart module delivers the power of process analytics without the headaches and cost associated with extensive process modeling. As a result, Altosoft can deliver any operationally relevant process KPI, even if processes extend across multiple source applications or databases. Also, unlike BI and direct event-based BAM solutions, Altosoft delivers awareness of individual process instances for monitoring purposes. If there is an irregularity in a specific process instance (e.g. a shipment for a priority customer that has been sitting on the loading dock for an inordinately long time), Altosoft can identify that problem and trigger a corrective workflow.

***Alerting and Incident Management.*** Altosoft's alerting capability enables users to codelessly define and monitor KPIs for operational processes. Altosoft continuously calculates, displays, and monitors KPIs based on real-time data from operational systems. If KPIs (or KPI valued that have been predicted by Predictive Process Analytics) fall out of user-defined acceptable ranges, Altosoft automatically alerts users alerts via dashboard, email, portal, wireless SMS, and more. Once an alert has been generated, Altosoft delivers fully integrated incident management service to manage response and resolution. Incidents are managed processes that include alert aggregation, escalation, reassignment, response, and verification. Altosoft's incident management capability enables immediate response and real-time resolution of current business issues. From monitoring to incident management and response, Altosoft enables closed-loop optimization of operational business processes.

***Simple and Cost-Effective Implementation.*** Altosoft's 100% code-free, intelligenctly orchestrated implementation process ensure rapid, cost-effective implementation. Altosoft's highly efficient KPI calculation and storage capability eliminate the costs and challenges of maintaining huge operational data stores or warehouses. And Altosoft delivers strong capability for integrating with third party systems via multiple techniques out of the box. As a result, implementation of Altosoft is extremely cost effective and typically takes just days or weeks instead of the months required by ordinary BI and BAM approaches.




Altosoft's approach fully supports the two OBI use cases presented above:

*Altosoft versus Scenario 1: Inventory Management and Upside Demand Flexibility for Retail Operations*

Like model-based and direct event-based BAM solutions, Altosoft Insight will calculate and monitor KPIs relating to DVD sales and inventory levels in real time. However, unlike direct direct-event-based BAM, Altosoft maintains the precise state of all shipments and placed orders. Any deviation from KPI ranges for aggregate sales data or inventory, as well as problems with individual process instances (e.g. a delayed shipment) would automatically trigger alerts to management. The net result is a solution which not only delivers real-time monitoring and requisite process analytics, but most importantly delivers the maximum effective lead time for management to make and execute a decision to return or reorder DVDs based on inventory levels and demand.

*Altosoft versus Scenario 2: Order Fulfillment for a Supply Company under Service Level Agreement*

In this scenario, Altosoft Insight monitors incoming orders and detects any deviation from expected levels. Altosoft's ability to understand historical order patterns and detect a deviation from that pattern can be used to trigger an automated alert to management. Altosoft's ability to monitor business process performance also gives management the ability to identify emerging process bottlenecks and efficiently identify where in the process additional human resources (inventory pickers, packers, shippers, etc.) could best be applied. For example, the system could determine that the main KPI, current orders versus average daily orders, is too high. Based on this, Altosoft would automatically generate an alert and notify

the warehouse manager that the daily fill rate SLAs are in danger of not being met unless action is taken. The warehouse manager can use this advanced warning to call in additional resources (or keep an existing shift overtime). Altosoft's process analytics could deliver additional guidance for what resources are required, and provide confirmation that order fill rates are back on track to meet contracted SLAs.

## Conclusion

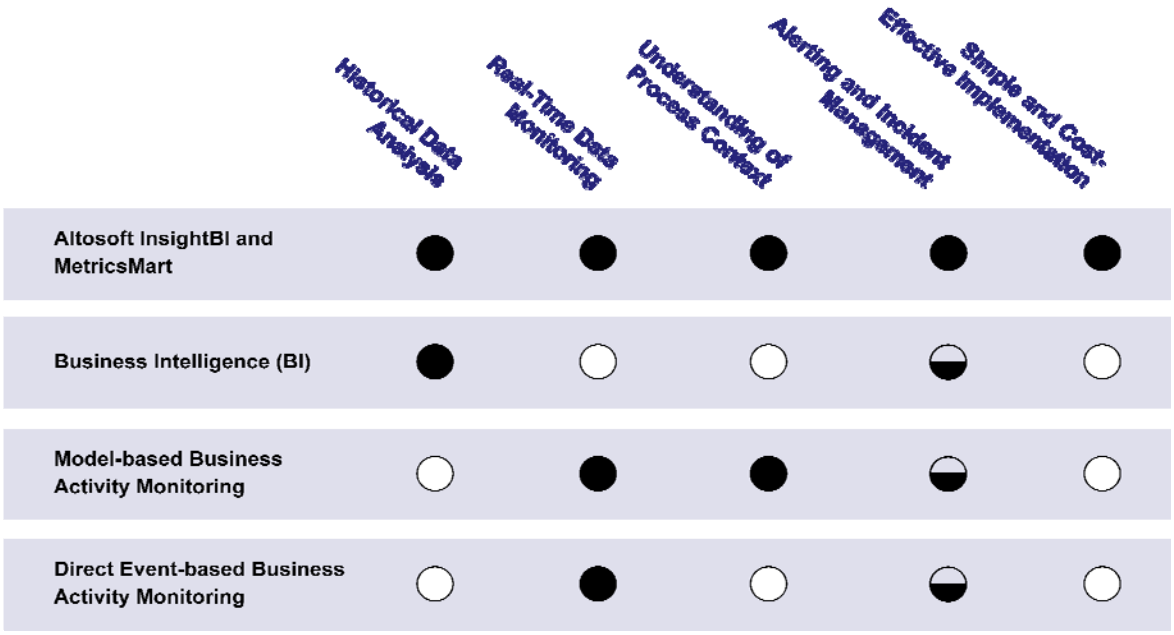
Existing architectural approaches to operational business intelligence have failed. No single approach is capable of supporting the complex requirements of OBI including:

- Historical data analysis.
- Real-time data monitoring.
- Understanding of process context
- Alerting and incident management; and
- Simple and cost-effective implementation.

A combination or hybrid of traditional BI and BAM architectures also falls short, since no BI or BAM solution delivers effective short term, operational forecasting and the project cost of implementing a hybrid solution is significantly higher than implementing a standalone solution.

A new approach is required.

Altosoft's InsightBI, MetricsMart, and ProcessMart solutions deliver a newer, superior architecture ideally matched to the requirements of operational business intelligence. Altosoft delivers both real-time event monitoring and scalable reporting and analysis of historical data from process KPIs. Altosoft eliminates the costly and time-consuming problem of process definition inherent in model-based BAM approaches. Altosoft also mitigates extensive requirements for up-front data warehouse creation, and/or data mart and cube development normally associated with ordinary BI solutions. The result is a powerful, flexible approach that meets the low-latency data requirements of operational BI, and supports dashboard based business monitoring with powerful features for historical data analysis and process optimization.



## About Altosoft

At Altosoft, we make BI work.

Altosoft's process-aware solutions eliminate the cost and complexity of conventional BI while delivering advanced functionality for operational performance improvement. Altosoft delivers the three critical enablers needed to transform existing business intelligence into a powerful, flexible engine of competitive advantage. First, Altosoft's business process intelligence capability adds essential process monitoring and analysis capability. Second, Altosoft provides agile data integration that makes the hardest part of BI easy by gathering data from various sources and converting dynamically it to KPI metrics. Finally, Altosoft facilitates rapid, reliable BI solution development with guided, code-free configuration and data governance features.

Altosoft's revolutionary, 100% codeless approach features ultra low-latency data monitoring and analysis across operational databases, warehouses, and other data sources; integrates real-time event monitoring and business process optimization; and enables dashboard development in minutes with a unique browser-based, drag-and-drop interface.

Headquartered outside of Philadelphia, Pennsylvania, Altosoft delivers solutions for commercial and government organizations worldwide.


1400 N Providence Road, Suite 6010
Media, PA 19063
USA
**www.altosoft.com**

Emmastraat 53
1213 AK
Hilversum
The Netherlands