PYRAMID ANALYTICS

The End of Ingestion?

The Beginning of In-Place Analytics

WHITE PAPER

Contents

Introduction

Ever since the formative stage of commercial computing, much emphasis has been placed on improving and accelerating the performance of database queries. As the volume of data and complexity of queries has increased, this has become an even larger imperative.

From the early days of indexed file systems and network databases, through relational systems and multidimensional OLAP engines, to the present day reality of in-memory analytic models, this is a lineage of technologies designed to speed up the responsiveness of database systems for complex analytic queries.

We can see then that ingesting data into in-memory analytics is simply the latest and last in a long line of different approaches to solve the same problem, an approach that has been mainstream for the past ten to fifteen years, but which perhaps has now had its day and can be relegated to its own specific niche.

With the advent of cloud-based computing, there is effectively a limitless resource that can be applied to the processing of analytic queries. It becomes, then, a different issue: how can the same level of analytic power be brought to bear against wildly different underlying storage options, completely transparent to most business users?

By abstracting the analytic capabilities from the underlying storage systems, Pyramid effectively solves this problem and provides a scalable, enterprise-class analytic environment that can be tailored to the functionality required by individuals or groups of users and which can operate without compromising the analytic power across a wide range of underlying data storage engines, providing the ultimate in In-Place Analytics.

> “With an ever-faster migration to cloud-based computing, processing and storage resources have become effectively limitless.

In-Place Analytics

Organizations have wrestled with analytic query performance for more than 50 years. In each decade, RAM, disk space, and CPU processors have become progressively cheaper and faster.

Compared to the processing capabilities of today, in the 1970s computing capabilities were rudimentary. Disk space was very expensive and CPUs were relatively slow. In the 1980s, with RAM still expensive, disk space cheaper, and CPUs faster, the rise of relational databases provided much greater flexibility. In the 1990s RAM was still expensive, but disk space was becoming cheaper—the decade saw the rise of pre-aggregated OLAP engines. The 2000s witnessed the emergence of in-memory analytic databases and desktop discovery tools, driven by cheaper RAM, bigger hard drives, and faster CPUs.

Today, with an ever-faster migration to cloud-based computing, processing and storage resources have become effectively limitless. Surely it would be better to analyze the data in place, utilizing the existing data storage engines with cloud-scale resources?

In-Place Analytics can be defined as the ability to conduct whatever analytic functionality is required, from simple arithmetic calculations, through complex multidimensional analyses, to sophisticated Machine Learning algorithms, without the need to ingest the data into a proprietary analytic database, and the ability to perform those calculations via a direct query model into the underlying data storage engine.

Effectively, it is the ability to abstract the analytic capability from the underlying data storage technology, in much the same way that a web browser-based UI is abstracted from the operating system upon which it is running. For a web browser user, all they need to know is how to operate the browser—they do not need to understand, or even know, if the underlying operating system is Windows or Linux.

Similarly, for a user of an analytic platform that supports In-Place Analytics, they do not need to know whether the underlying data they are analyzing is stored in a relational database, a multidimensional database like SQL Server Analysis Services or SAP BW, an in-memory engine (Pyramid's own or another like SQL Server Analysis Services Tabular mode or Exasol), or a cloud-based system like Snowflake. The user interface and analytic and visualization capabilities remain the same.

Some In-Place Analytics platforms, like Pyramid, can also overlay additional metadata and security functions on top of the underlying data storage engine. Data field names, descriptions of attributes and measures, and explanations of calculated items—as well attribute, measure, and row-level security—can be added to the existing schema in order for the business user to better understand the data they are using.

> In-Place Analytics lets users perform analytic calculations—from basic to advanced—via a direct query into the underlying data storage engine without ingesting the data into a proprietary analytic database.

History of Analytic Performance

Organizations have wrestled with analytic query performance for more than 50 years. In each decade, RAM, disk space, and processing capabilities have become progressively cheaper and faster. This progression has enabled an environment which supports In-Place Analytics.

1970s

In the 1970s RAM was ludicrously expensive and disk space very expensive, and it was coupled with relatively slow CPUs. Very detailed design work was required using indexed file systems of network and hierarchical databases to retrieve data from large datasets in an acceptable time period.

1980s

In the 1980s, with RAM still expensive, disk space cheaper, and CPUs faster, the rise of relational databases provided much greater flexibility. However, the vendors of such systems were very much focused on providing higher performance for OLTP systems to displace the previous generation of database systems. High performance querying again required detailed indexing schemes.

1990s

Vendors in the 1990s sought to address the poor query performance of relational systems. With RAM still expensive, disk space becoming cheap, and 64-bit CPUs arriving on the market, highly indexed multidimensional, pre-aggregated OLAP engines provided an easy way for business users to use techniques that delivered consistently fast response times to analytic queries.

2000s

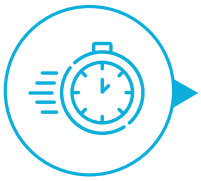
In the mid-2000s, with RAM now cheap and multicore 64-bit CPUs faster than ever, in-memory analytic databases became feasible and highly performant visual data discovery desktop tools exploded onto the market.

2020s

In the 2020s, with an ever-faster migration to cloud-based computing, resources have effectively become limitless. However, the volumes of data requiring analysis has increased exponentially. It is starting to become troublesome, time-consuming, and expensive to ingest data into proprietary in-memory analytic tools, especially as most of those tools were designed for individual desktop use and are awkward to deploy and govern in an enterprise environment.

Advantages of In-Place Analytics

The abstraction of analytic functionality from underlying data engines provided by In-Place Analytics platforms can lead to several advantages.



Speed to analytics and leveraging existing analytic models

The Direct Query mechanism of In-Place Analytics can significantly reduce the time it takes to embark on analysis of the data from the time the analytic platform is in place. For some analytics engines, like SQL Server Analysis Services (both modes) or SAP BW and SAP HANA, it is a matter of "snap-on and go." The analytic platform can consume the existing semantic models in terms of dimensions, hierarchies, attributes, measures, and user-level security, and use that to drive the analytic UI and direct query generation. This means there is no need to recreate data and security models in a proprietary engine, and thus saves considerable time and energy that can be better applied to the analysis of the data itself.

Flexible storage options per project requirements

In-Place Analytics gives much more flexibility to choose the data storage options for different analytic projects. When it's necessary to analyze very large data sets spanning billions of rows of data in an Enterprise Data Warehouse, there is simply not the time or resources to ingest that amount of data into a proprietary in-memory engine.

Similarly, when low latency between capturing and analyzing data is crucial, it's not enough to copy manage the data from the collection database and load it into an in-memory engine. Indeed, many in-memory engines make it difficult to create incremental data loads, and often require the entire model to be reloaded or rebuilt, which only exacerbates the problem.

Centralized analytic data lakes

An increasing number of companies are moving towards a centralized analytic "data lake-type" resource. In this scenario, curated data sets are provided for business use, but any analytic models built using those data sets must also be written back to the centralized storage engine for more efficient governance of the data—and increasingly the analytic content as well. Snowflake, Redshift, and other cloud-based data warehouses featuring high data volume repositories are all technologies that exemplify this approach.

> In-Place Analytic platforms can free an organization from reliance on siloed proprietary in-memory engines, providing a consistent experience (regardless of underlying data services) without compromising analytic power.

Pyramid In-Place Analytics Scenarios

Semantic Models

All analytic models contain a semantic model that describes the data and its relationships. In a relational database, this is often a relatively simple schema that describes the tables and their defined joins through keys and foreign key relationships. Multidimensional engines add defined hierarchies and aggregation rules, plus calculation definitions and additional user access rules. While proprietary in-memory engines also have a similar semantic model, it is often inaccessible to third-party products for consumption and use.

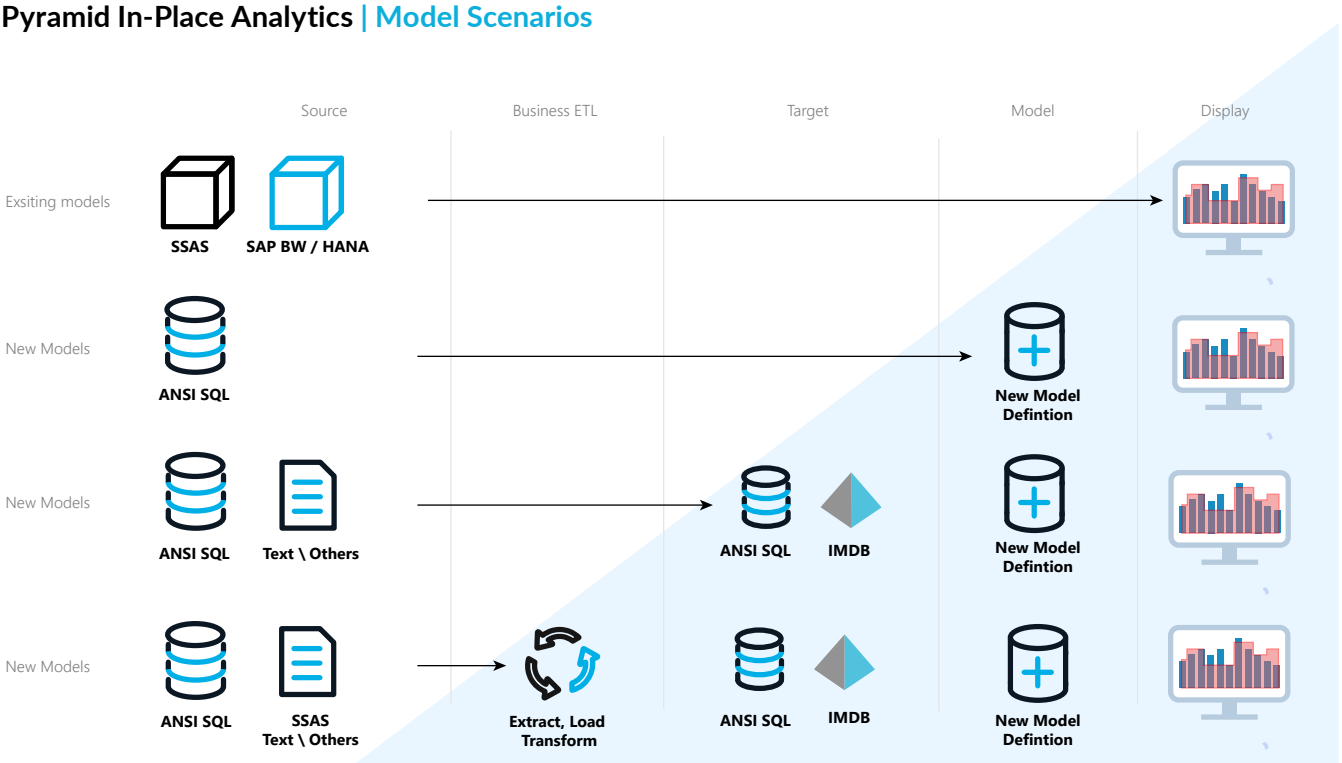
Pyrana – Pyramid's Query and Calculation Engine

The Pyramid analytic platform also relies on semantic models of the data to drive the user interface and query generation process. However, because of its unique Direct Query approach, it makes much better use of—and exploits—the model further than many other tools.

Pyramid's Pyrana Query and Calculation Engine uses the semantic model to drive its MDX or SQL Direct Query Engine, and provides additional analytic power where the underlying engine does provide it directly (for example, defined hierarchies against a relational database).

The graphic below represents the different semantic model scenarios that Pyramid supports.

Pyramid In-Place Analytics | Model Scenarios



Pyramid Semantic Model Scenarios

Multidimensional

For multidimensional engines, Pyramid consumes and directly uses the semantic model already defined within that engine. Thus, for analytic engines such as SQL Server Analysis Services and SAP BW and SAP HANA, there is no need to develop any semantic model in Pyramid. Pyramid reads, consumes, and uses the existing model—and that model drives the UI and query generator. It also uses and respects security rules contained in those models to restrict users to data they are authorized to consume.

In these instances, Pyramid will dynamically generate MDX language queries and submit them to the underlying analytic engine for execution. This applies to calculations defined in Pyramid for derived measures and members, as well as dynamic lists.

Thus all queries and calculations are evaluated and resolved by the underlying analytic engine, making full use of its analytic power as well as ensuring that all data (and not just that which has been extracted and ingested) can potentially take part in those calculations. Further, the calculation results are 100% consistent.

Pure Relational

Where data needs to be directly queried in a relational database, Pyramid can construct a "virtual model" where the tables and data fields can be selected, their relationships defined, hierarchies and aggregation rules created, and security defined. This model is then used when dynamically generating SQL queries to that database.

Blended Data

Where there is a need to blend data from multiple sources (different databases, local files, cloud-based data, etc.), Pyramid can define and orchestrate the data retrieval and flow, then write the resulting blended dataset to a specified target database system. This may be one of any number of relational systems, or even third-party analytic engines such as SAP HANA or SQL Server Analysis Services Tabular mode databases.

Transformations and Machine Learning

If necessary, the orchestrated data flow may also include transformations on the data at the row level as it passes through the process. This may include calculations between data field values, adding richer content (for example decomposing date/time fields into time periods such as years, months, weeks, etc.) or even applying predefined or user-specified Machine Learning algorithms in Python or R. Once any transformations are complete, the data contained in the flow may be written to different targets as above.

Pyramid In-Memory Database

While Pyramid does supply an In-Memory Engine (IMDB) in its platform, it is important to realize that this is treated in the same way as any other SQL-based data source. IMDB is simply an in-memory column store relational database. Pyramid reads and writes to it in the same way as any other SQL data source. Thus, it can be utilized, or not, depending on the requirements of the analytic project concerned or the mandates of the data stewards within the organization.

Conclusion

We are moving at an accelerated rate into an era of almost limitless computing resources. In this reality, it seems to be simply an additional overhead to have to ingest data into a limited in-memory proprietary engine to achieve the analytics required.

In-Place Analytic platforms, like Pyramid, can free an organization from reliance on siloed proprietary in-memory engines, providing a consistent look and feel for all analytic needs, regardless of underlying data services—all without compromising on the analytic power available to the users.

In this case, it's worth evaluating an In-Place Analytics strategy that will insulate your organization from any change to the data storage technologies for years to come. Importantly, it will provide an analytic environment that is consistent, fast, and complete.

PYRAMID
ANALYTICS

CONTACT US
+ 1 800 385 6704
www.pyramidanalytics.com
sales@pyramidanalytics.com