

Guide to Sub-Second Analytics

Table of Contents

Understanding Sub-Second Analytics	02
Real-World Use Cases of Sub-Second Analytics	04
Common Challenges in Sub-Second Analytics	05
Key Components of Sub-Second Analytics	08
Concurrency and Low Latency for Serving Data	12
Leveraging the Cloud for Sub-Second Analytics	15
Implementing Sub-Second Analytics	17
Optimizing for Sub-Second Analytics	19
Future Trends in Sub-Second Analytics	25

Understanding Sub-Second Analytics

What is Sub-Second Analytics?

Sub-second analytics leverages cutting-edge technologies, optimized data processing techniques, and high-performance infrastructure to enable organizations to respond quickly to changing business conditions, customer demands, and market trends. By reducing the time between data capture, analysis and serving, sub-second analytics empowers businesses to make proactive decisions, uncover hidden patterns, detect anomalies, and capitalize on new opportunities.

Benefits of Sub-Second Analytics

Sub-second analytics offers numerous advantages over traditional analytics approaches. Some of the key benefits include:



Rapid Decision-Making

With sub-second analytics, decision-makers can access up-to-the-minute insights, allowing them to respond swiftly to critical situations, seize opportunities, and mitigate risks in a rapidly evolving business landscape.



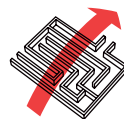
Enhanced Operational Efficiency

Subsecond analytics improves operational efficiency by enabling faster data processing and analysis. It reduces the time and effort required to obtain insights, streamlines workflows, and enhances productivity.



Improved Customer Experience

Sub-second analytics facilitates organizations to personalize offerings, provide targeted recommendations, and deliver a superior customer experience.



Competitive Advantage

The ability to analyze data and derive insights quickly provides a significant competitive edge in today's data-driven landscape. Sub-second analytics enables organizations to respond swiftly to market changes, identify emerging trends, and make data-driven decisions ahead of their competitors.

In summary, sub-second analytics empowers organizations with faster decision-making, improved operational efficiency, and a competitive advantage in the market. By harnessing the power of sub-second analytics, businesses can unlock the full potential of their data and drive innovation across various domains.

Real-World Use Cases of Sub-Second Analytics

Sub-second analytics has a wide range of applications across various industries. Here are some examples.

E-commerce and Retail

Inventory Management

By analyzing sales data, stock availability, and customer demand, E-commerce and Retailers can optimize inventory levels, reduce stockouts, and ensure timely replenishment.

Personalized Recommendations

Understanding customer browsing behavior, purchase history, and demographic data, E-commerce platforms can offer relevant and targeted recommendations, increasing sales and customer satisfaction.

Finance and Banking

Fraud Detection

Financial Institutions can identify suspicious transactions through analytics on transactional data, user behavior patterns, and historical data. This can help them take immediate action to prevent fraud.

Risk Management

By analyzing market data, trading patterns, and economic indicators, Banks can assess risk and adjust investment strategies or initiate risk mitigation measures promptly.

Healthcare

Patient Monitoring

By analyzing vital signs, sensor data, and patient records, healthcare providers can detect anomalies, trigger alerts, and take immediate action to ensure patient safety.

Predictive Analytics for Disease Outbreaks

Early identification of disease outbreaks through social media and geographical data can help identify disease hotspots, so that health care resources may be deployed effectively.

Manufacturing

Predictive Maintenance

Manufacturers can detect patterns, identify potential failures, and schedule proactive maintenance activities, minimizing downtime and optimizing operational efficiency.

Supply Chain Optimization

Manufacturers can make informed decisions, such as adjusting production schedules, optimizing logistics routes, and managing inventory levels, to enhance efficiency and reduce costs.

These are just a few examples of how organizations across different industries can leverage sub-second analytics to drive business outcomes. Many industries are evolving from batch reporting to interactive data apps to serve insights. These customer-facing dashboards rely on sub-second analytics to enhance the user experience.

Common Challenges in Sub-Second Analytics

Implementing sub-second analytics can come with its own set of challenges. This chapter will explore common challenges organizations may encounter when implementing sub-second analytics and discuss potential solutions.

Data Integration

Data integration from disparate sources such as databases, APIs, logs, and SaaS applications can be challenging. Integrating and harmonizing diverse data sources with schemas, data types, and granularities becomes complex. Additionally, data engineers must reconcile the volume and velocity of data from the source with user expectations. For example, if the velocity exceeds the infrastructure capabilities to process the data, this will result in stale data. If the source systems impose rate limits, this could also impose limitations in processing data irrespective of infrastructure sizing.

Solution:

Building a flexible architecture that can connect to disparate data sources quickly and efficiently is critical. Infrastructure elasticity to meet data flow demands can help manage volume and velocity variations. Data integrations require a thorough understanding of the source and target data models, transformation and enrichment requirements, and generation and delivery of insights to the target audience. The technical architecture should align to support these needs regarding speed, scale, availability, supportability, and efficiency.

Handling Large Data Volumes

Dealing with large volumes of data can impact the performance and speed of sub-second analytics. Processing and analyzing massive datasets can be resource-intensive and may lead to delays in delivering insights. Analytics processes rely heavily on sorting, aggregating metrics across various dimensions. Sorts and aggregates are expensive operations, requiring data to be shuffled across the network and consume compute cycles. As the data volume grows, so does the complexity in delivering low latency data.

Solution:

Distributed Computing: Implement distributed computing frameworks that allow data to be processed in parallel across multiple nodes. This parallelization enables efficient utilization of compute resources and improves data processing speed.

Data Partitioning: Divide the data into smaller partitions and distribute them across multiple nodes. This approach enables parallel processing and reduces the time required for data retrieval and analysis.

Pre-processing: Sorting and aggregating data ahead of time, either during the ingestion or transformation, can generate insights rapidly. This pre-processing shifts the burden to the left, to the development/data engineering team, as each aggregation or sort is addressed in code. Adopting platforms that optimize pre-processing can help developer productivity by eliminating secondary pipelines and simplifying the development process.

Data Latency and Query Latency

Data latency refers to the delay between data generation and its availability for analysis. Minimizing data latency is crucial for real-time insights and timely decision-making in sub-second analytics. Query latency refers to the time to analyze and serve insights. This attribute is focused on the data-serving needs of the end user. Various factors impact query latency, ranging from the amount of data, lack of resources, poor data model, etc.

Solution:

Real-Time Data Integration: Implement real-time data integration techniques to capture and process data as it is generated. Utilize technologies such as change data capture (CDC) to propagate data changes in real-time, ensuring the availability of up-to-date data for analysis.

Stream Processing: Utilize stream processing frameworks that enable the ingestion and processing of high-velocity data streams. These frameworks reduce data latency by processing data as it flows into the system, enabling real-time analytics.

To address query latency challenges, purpose-built solutions can be leveraged. This ranges from specialized compute instances with varying amounts of memory and NVMe storage or using fast caching layers such as Redis. Continuous optimization is the name of the game for addressing query latency.

Data Accuracy and Consistency

Ensuring data accuracy and consistency is crucial for reliable sub-second analytics. Only accurate or consistent data can lead to correct insights and decision-making.

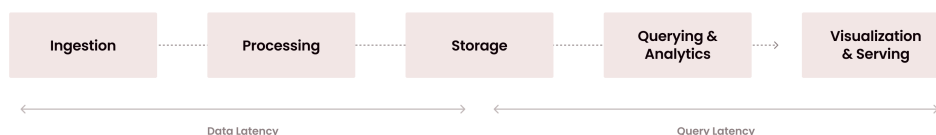
Solution:

Data Quality Assurance: Implement robust data quality assurance processes, including data cleansing, validation, and verification. Utilize data profiling techniques to identify and address data quality issues, ensuring the accuracy and consistency of the data used for sub-second analytics. Additionally, by automating and orchestrating data quality checks early in the data integration process can help eliminate data quality issues.

Data Governance: Establish practices to define data standards, access controls, and data lineage. Implement data governance frameworks to ensure data remains accurate and consistent throughout its lifecycle in the sub-second analytics environment.

Key Components of Sub-Second Analytics

Sub-second analytics relies on several components that enable fast and efficient data processing and analysis. Understanding these components is essential for implementing a successful sub-second analytics solution. This chapter will explore the fundamental elements that constitute sub-second analytics infrastructure. As shown below, the idea of managing sub-second analytics spans data latency and query latency. These two latency factors must be considered while building the architecture for sub-second analytics. A data pipeline comprises various stages, each being addressed by a specific set of technologies. For example, a data warehouse solution can be leveraged for the first four stages, while visualization and serving can differ.



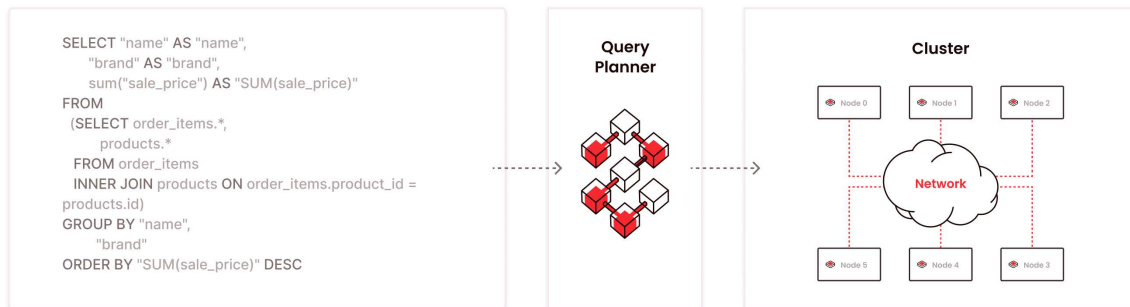
High-Performance Data Processing

High-performance data processing forms the foundation of sub-second analytics. It involves employing advanced data processing techniques and technologies to optimize the speed and efficiency of data ingestion, transformation, and analysis. Key components of high-performance data processing are listed below

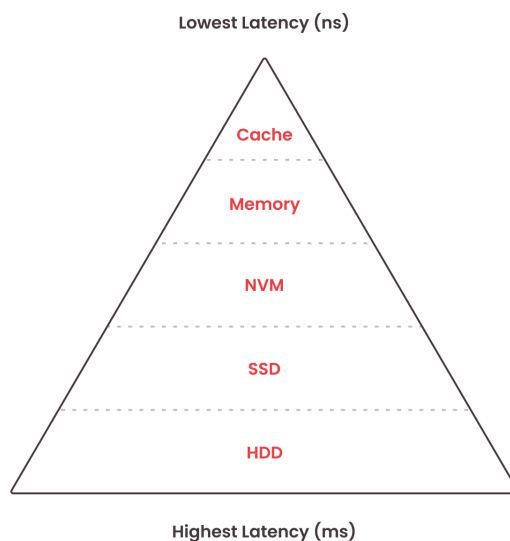
Distributed Computing:

Distributed computing frameworks enable data to be processed in parallel across multiple nodes. This parallelization significantly speeds up data processing and analysis, allowing for sub-second response times.

Distributed query engines parallelize query execution across multiple nodes. This parallelization enhances performance and accelerates data processing and analysis. MapReduce frameworks divide data processing tasks into smaller units that are executed in parallel across a cluster of machines. This parallelization improves efficiency and enables sub-second analytics.



In-Memory Computing: Storing data in memory rather than on disk allows for faster data access and processing. In-memory databases and caching techniques ensure that frequently accessed data is readily available for analysis.



Columnar Storage and Data Partitioning: Columnar storage stores data in a column rather than a row-wise format. This technique improves query performance and reduces data retrieval time, making it well-suited for sub-second analytics. Storing column-level data together results in data storage efficiencies, including better compression. With analytics queries accessing specific columns, there is no need to retrieve entire rows with unneeded columns, as in the case of row format. Partitioning data across multiple nodes allows for parallel processing and efficient utilization of compute resources, enabling faster analysis of large datasets.

Row Format

All fields for a single record are stored sequentially on disk

ID	Name	Address	City	State	Date of Birth
4344	John Doe	1203 Oak Ct	Palo Alto	CA	02/06/1998
21210	Jane Smith	Apt 401, 4301 Orange Blossom Trail	Houston	TX	04/12/2001
31456	Jake Smith	35 Main St	Chicago	IL	11/05/2003

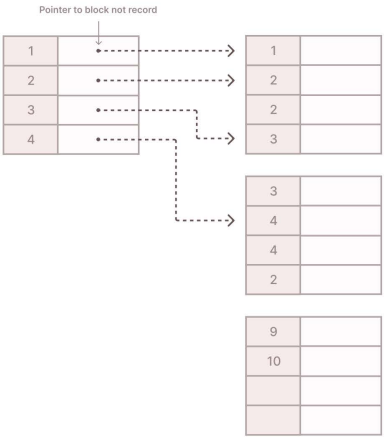
Columnar Format

All values for a single column are stored sequentially on disk

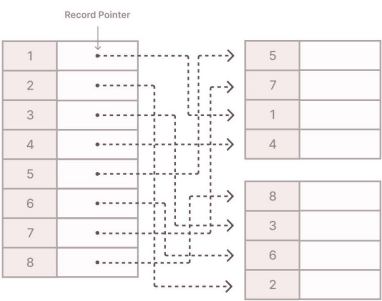
ID	Name	Address	City	State	Date of Birth
4344	John Doe	1203 Oak Ct	Palo Alto	CA	02/06/1998
21210	Jane Smith	Apt 401, 4301 Orange Blossom Trail	Houston	TX	04/12/2001
31456	Jake Smith	35 Main St	Chicago	IL	11/05/2003

Indexing: In the realm of analytics, indexing allows for lightning-fast data exploration and real-time insights. When dealing with large datasets, indexing provides a way to create optimized data structures that enable rapid querying and analysis. By indexing specific fields or columns within a dataset, analysts can access subsets of data in subsecond timeframes, facilitating quick decision-making and enhancing productivity. Sorting and grouping are expensive operations in the analytics world. The use of indexes makes these operations efficient.

Example of Clustered Index

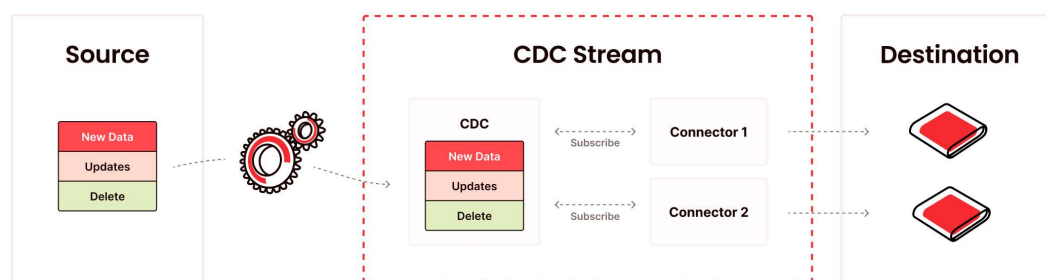


Example of Non-Clustered Index

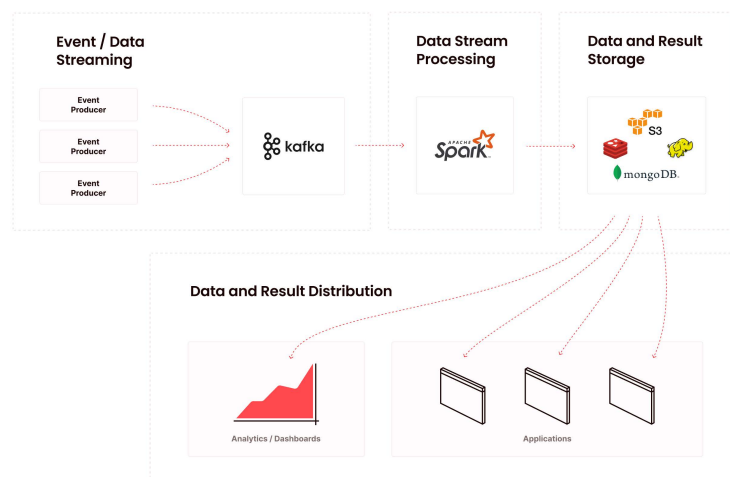


Real-time data integration is another component of sub-second analytics. This specific component may not apply to all use cases. It involves capturing, integrating, and processing data as it is generated or updated in real-time. Real-time requirements should be clearly evaluated as these requirements can result in a complex solution with a high total cost of ownership. Key components of real-time data integration include:

Change Data Capture (CDC): CDC techniques capture and propagate data changes from source systems to target systems, ensuring that the most current data is available for analysis, enabling real-time insights. CDC eliminates the need to reload an entire dataset, and changes are incrementally incorporated into the final dataset.



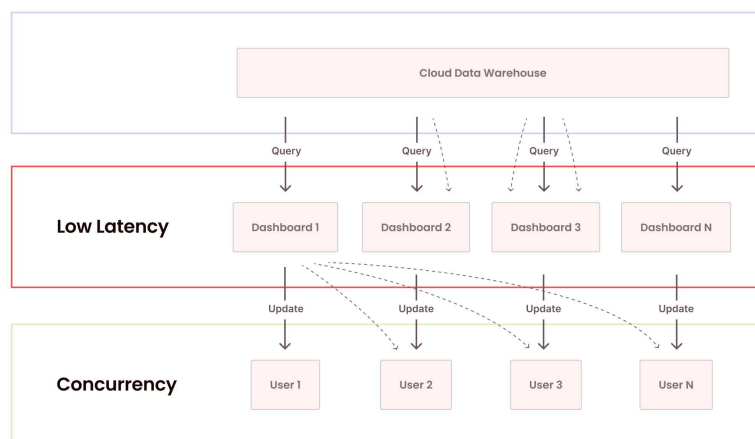
Streaming Data Ingestion: Stream processing frameworks and technologies enable the ingestion and processing of high-velocity data streams in real-time. They facilitate real-time data integration and analysis for sub-second analytics.



Concurrency and Low Latency for Serving Data and Insights

Concurrency and Low Latency

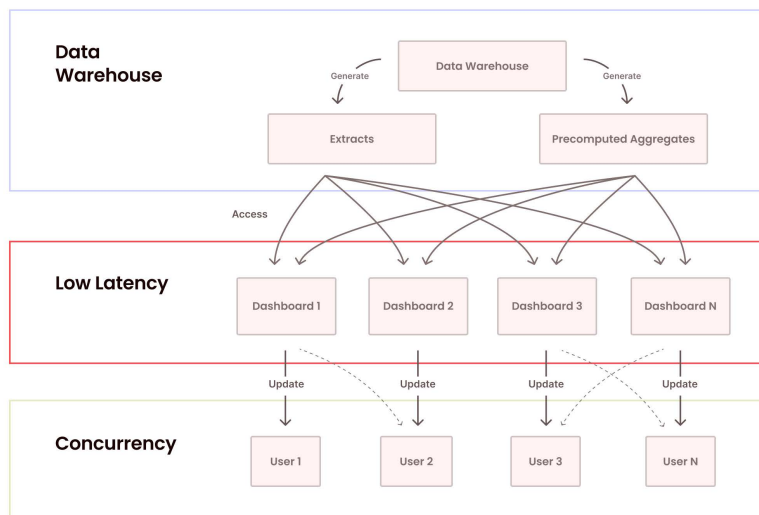
In the ever-evolving landscape of sub-second analytics, the ability to serve data and insights with high concurrency and low latency is crucial. The number of concurrent users and queries, especially with the advent of data apps, is on the rise. A simple dashboard could synthesize results from multiple tables and views, launching not one but many queries against the data warehouse or other data stores. Additionally, a single query can be broken down into multiple tasks to be scheduled across a pool of processors available in the data infrastructure. The cost of delivering low latency analytics under high concurrency varies by technology and architectural decisions.



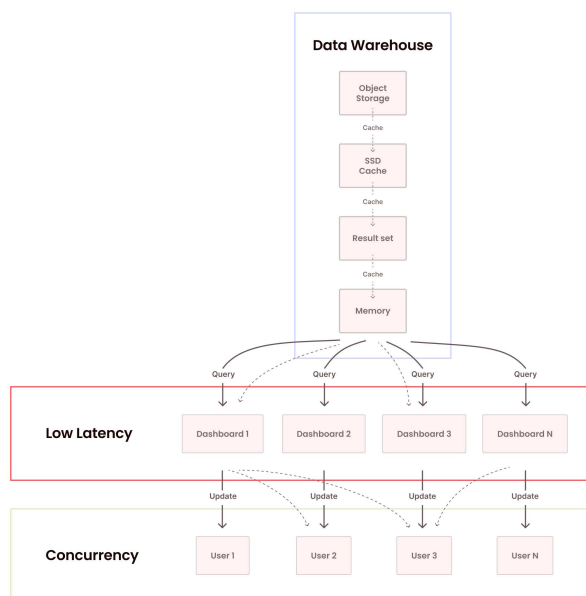
Strategies for Achieving Concurrency and Low Latency

Data Replication and Distribution: Implement data replication and distribution techniques to ensure that data is available in multiple locations, closer to the users. This reduces network latency and enables faster data access and analysis.

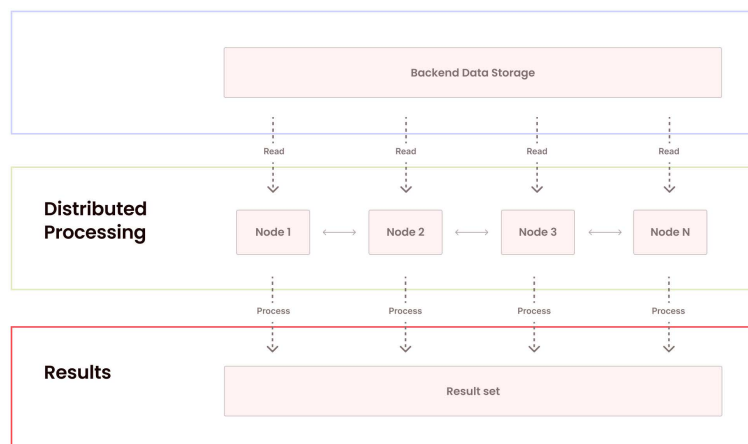
Leveraging Extracts and Pre-computed aggregates: Business intelligence tools typically leverage extracts from source systems to improve performance. This provides a local copy that eliminates the need to go back to a back-end data warehouse or data lake for every query. They provide a snapshot of the data at a specific time and are helpful for use cases where real-time data is not required. Stale data and failed refreshes are constant challenges with this approach. Another approach is to avoid repeated aggregations of data that can be an expensive proposition for back-end data stores. This is done through pre-computed aggregations or materialized views to help reduce back-end processing.



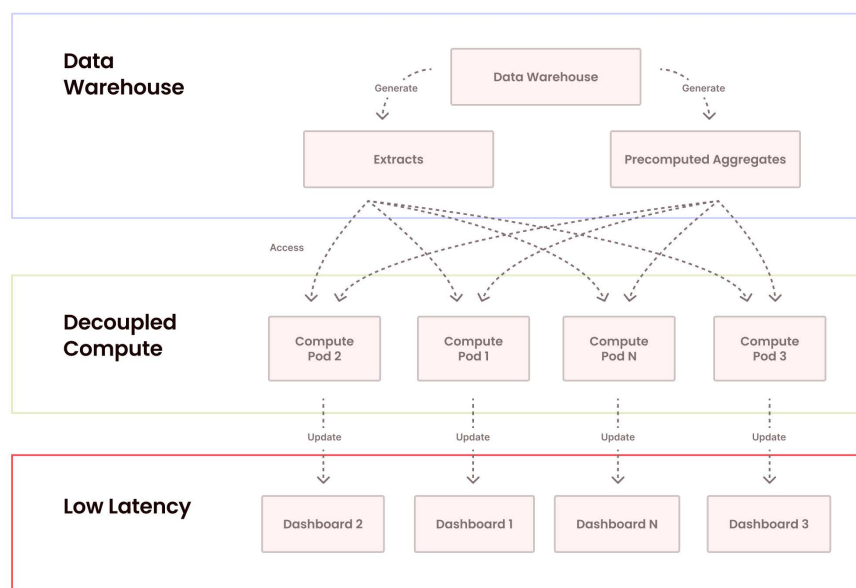
Caching: Utilize caching mechanisms and in-memory computing to store frequently accessed data in memory, reducing the need for disk I/O and accelerating data retrieval. In-memory processing enables faster calculations and analysis, enhancing concurrency and low latency.



Distributed Query Processing: Implement distributed query processing frameworks that enable parallel execution of queries across multiple nodes. This parallelization distributes the workload and improves query response times, enhancing concurrency and low latency.



Decoupled compute & storage architecture: Separating processing from data provides the ability to scale the number of compute engines to increase concurrency.



Concurrency and low latency are vital in serving data and insights in sub-second analytics. The ability to handle high user concurrency and deliver data with minimal delay ensures real-time decision-making, interactive analysis, and timely insights. By understanding the differences between extracts and live data and adopting strategies to achieve concurrency and low latency, organizations can optimize their sub-second analytics infrastructure and deliver superior user experiences.

Leveraging the Cloud for Sub-Second Analytics

Cloud-based technologies provide several advantages for sub-second analytics:



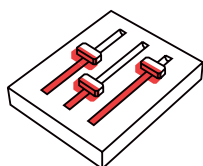
Scalability

Cloud platforms, such as Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform (GCP), offer virtually unlimited scalability. Organizations can quickly scale their infrastructure resources, such as compute power, storage, and networking, to handle the high data volumes and processing demands of sub-second analytics.



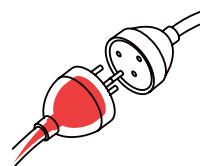
Elasticity

Cloud platforms enable automatic scaling based on demand. With auto-scaling capabilities, organizations can configure their analytics infrastructure to adjust resources dynamically based on consumption. This elasticity ensures that resources are allocated efficiently, optimizing cost while maintaining sub-second performance during peak usage.



Managed Services

Cloud providers offer a wide range of managed services specifically designed for analytics, such as Redshift, Firebolt, Snowflake, Google BigQuery, and Azure Synapse Analytics. These services eliminate the need for organizations to manage the underlying infrastructure and enable them to focus on data analysis and insights generation. Managed services provide high-performance analytics capabilities, scalable storage, and optimized query processing, all crucial for sub-second analytics.



Data Integration and Processing

Cloud platforms offer a variety of tools and services for data integration and processing. For instance, AWS Lambda, Azure Functions, and Google Cloud Functions enable organizations to process data streams and trigger sub-second analytics workflows. Cloud-based data integration services like AWS Glue, Azure Data Factory, and GCP Dataflow simplify ingesting, transforming, and integrating data from various sources in enabling sub-second analytics workflows.



High-Speed Networking

Cloud providers offer high-speed networking infrastructure, allowing for efficient, reliable data transfer and communication between different components of the analytics ecosystem.



Global Availability

Cloud platforms have a vast global footprint, with data centers worldwide. Deploying analytics infrastructure closer to their users or data sources reduces data transfer latency and improves overall performance. Cloud providers offer the flexibility to choose the optimal region for data processing, ensuring low-latency access to data.



Cost Optimization

Cloud platforms provide cost optimization features, such as pay-as-you-go pricing, which allows organizations to pay only for the resources they consume. Additionally, cloud providers offer pricing models and cost management tools that enable organizations to optimize resource allocation and achieve cost-efficiency without compromising sub-second performance.

By leveraging cloud-based technologies, organizations can leverage the scalability, elasticity, managed services, data integration capabilities, high-speed networking, global availability, and cost optimization features offered by cloud platforms. These capabilities empower organizations to implement and scale sub-second analytics solutions effectively, delivering insights and driving data-driven decision-making.

Implementing Sub-Second Analytics

Implementing sub-second analytics requires careful planning, infrastructure, and data management. In this chapter, we will explore the key steps involved in implementing sub-second analytics in your organization.

Define Business Objectives

Begin by defining clear business objectives for implementing sub-second analytics. Identify the specific use cases and areas where rapid insights can add value and drive business outcomes. Understand the key questions you want to answer and the decisions you need to make in real time. Identify the target audience and how they will leverage the insights towards meeting business objectives.

Assess Data Requirements

Assess your data requirements to ensure you have the necessary data sources and infrastructure. Identify the critical data elements for analysis and determine the data capture frequency and granularity required for sub-second analytics. Understand data retention requirements and period for analytics. These have a direct bearing on performance and implementation costs. Additionally, review the need for real-time analytics. In most cases, real-time analytics means updated dashboards every 10 minutes. The technologies required to implement real-time streaming analytics vs. 10-minute updates can differ vastly.

Design Data Architecture

Design a data architecture that utilizes optimal data storage and processing technologies for data ingestion, transformation, and analysis. Consider distributed computing frameworks, in-memory databases, and streaming data processing technologies to handle high-velocity data streams efficiently. Data modeling is a critical component.

Establish Data Governance and Security

Establish robust data governance practices to ensure data quality, privacy, and security. Define data standards, access controls, and lineage to maintain data accuracy and consistency. Implement encryption, authentication, and authorization mechanisms to protect sensitive data and comply with relevant regulations.

Build Data Models and Analytical Capabilities

Design data models that are optimized for sub-second analytics. Consider denormalization, star schema design, and columnar storage techniques to enhance query performance and enable faster data retrieval. Build analytical capabilities, such as pre-aggregation, indexing, parallel processing, and advanced analytics algorithms, to derive insights from the data. Understanding the end-user interaction and serving needs will be critical to developing the appropriate model. Data model implementations vary based on the backend technologies selected. For example, use of specialized indexes can simplify the approach.

Monitor, Optimize, and Iterate

Continuously monitor the performance of your sub-second analytics infrastructure and processes. Monitor query performance, data processing times, and resource utilization to identify bottlenecks and optimize system performance. Iterate your data models, analytics algorithms, and visualization techniques based on feedback and changing business requirements.

Foster a Data-Driven Culture

Promote a data-driven culture within your organization. Encourage data literacy, provide training, and foster collaboration between business stakeholders and data professionals. Empower employees to use real-time insights for decision-making and support a culture of continuous learning and improvement.

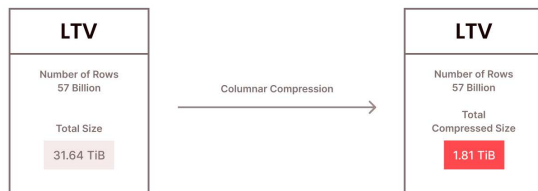
Optimizing for Sub-Second Analytics (Example)

Step 1: Columnar Storage optimization

Let's look at a typical AdTech data set leveraging Firebolt Cloud Data Warehouse. "LTV" table is used to measure ad performance across various apps and devices in the table below. The amount of data in the LTV table exceeds 50 billion records. Data in this table consumes approximately 32TB of storage.

```
CREATE TABLE IF NOT EXISTS "ltv" (  
  "ltv_hour_tz" text NOT NULL,  
  "app_id" text NOT NULL,  
  "campaign" text NOT NULL,  
  "ltv_country" text NOT NULL,  
  "currency" text NOT NULL,  
  "ltv_currency" text NOT NULL,  
  "ad" text NOT NULL,  
  "ad_id" text NOT NULL,  
  "adset_name" text NOT NULL,  
  "adset_id" text NOT NULL,  
  "campaign_id" text NOT NULL,  
  "unmasked_media_source" text NOT NULL,  
  "media_source" text NOT NULL,  
  "partner" text NOT NULL,  
  "site_id" text NOT NULL,  
  "channel" text NOT NULL,  
  "event_name" text NOT NULL,  
  "ltv_device_rank" text NOT NULL,  
  .  
  .  
  .  
  "dashboard_device_rank" text NOT NULL,  
  "source_file_name" text NOT NULL,  
  "source_file_timestamp" timestamp NOT NULL,  
  "ltv_timestamp_date" date NOT NULL DEFAULT  
)
```

Columnar storage is an effective way to optimize storage for analytics. Firebolt uses columnar compression to store data on object storage.



Key Benefits of this approach is the 18 X compression that optimizes data storage. This capacity reduction also reduces data transfer over the network. Column level access eliminates the need to retrieve an entire row of data and reduces disk I/O. Firebolt automatically leverages columnar compression when data is loaded into the database.

Step 2: Query Optimization

Consider the sample aggregation query below.

```
WITH
  Itv AS (
    select
      Itv.*,
      acc.name as acc_name,
      acc.shipping_country,
      acc.region,
      app.name as app_name,
      app.platform,
      app.app_slug,
      app.owner_account
    FROM
      Itv -- (42B rows)
      LEFT JOIN owned_apps AS app ON Itv.app_id = app.app_slug
      LEFT JOIN account AS acc ON acc.id = app.owner_account
  )
SELECT
  Itv.app_name AS "Itv.app_name",
```

```

ltv.media_source AS "ltv.media_source",
ltv.acc_name AS "ltv.acc_name",
ltv.shipping_country AS "ltv.shipping_country",
ltv.region AS "ltv.region",
ltv.platform AS "ltv.platform",
COALESCE(
  SUM(
    CASE
      WHEN (ltv.attribution_type = 'install') THEN ltv.clicks_count
      ELSE NULL
    END
  ),
  0
) AS "ltv.total_clicks_count",
COALESCE(
  SUM(
    CASE
      WHEN (ltv.attribution_type = 'install') THEN ltv.impressions_count
      ELSE NULL
    END
  ),
  0
) AS "ltv.total_impressions_count",
COALESCE(SUM(ltv.inappevents_count), 0) AS "ltv.total_inappevents_count",
COALESCE(SUM(ltv.launches_count), 0) AS "ltv.total_launches_count",
COALESCE(
  SUM(
    CASE
      WHEN (ltv.attribution_type = 'install') THEN ltv.installs_count
      ELSE NULL
    END
  ),
  0
) AS "ltv.total_noi_count"
FROM
  ltv
WHERE
  (
    (
      (ltv.ltv_timestamp_date) >= (TIMESTAMP '2023-03-01')
      AND (ltv.ltv_timestamp_date) < (TIMESTAMP '2023-03-07')
    )
  )
GROUP BY 1,2,3,4,5,6
ORDER BY 7 DESC

```

Use Primary Index (clustered indexes)

With a primary index, the order of the index corresponds to physical ordering of data on storage. For example, if you have a date data type, when we are filtering for ranges, a primary index will deliver a sequential access pattern and data pruning on that range. With a primary index, the data will be retrieved using minimal compute resources.

With the LTV table, Firebolt's Primary index uses the following columns: ltv_timestamp_date, media_source, app_id, sorts and physically orders the data on disk. Additionally, these sparse indexes track ranges of data and help effective data pruning to reduce the amount of data accessed based on the query. With the introduction of the Primary index, the query above does not need to scan 1.81TiB; instead, it scans a mere 48.44 GB in 3.03s.



Optimizing aggregations

Aggregations can be implemented as secondary pipelines or as materialized views. Pre-aggregating data that is accessed repeatedly helps reduce resource consumption and optimizes performance.

With Firebolt, you can create an aggregating index using the following columns for group by: ("ltv_timestamp_date", "app_id", "media_source", "ltv_country", "attribution_type") as shown below. An aggregating index on Firebolt is updated at ingest and functions as a single index at various granular levels providing a single mechanism for aggregating data. For example aggregating data at daily, weekly, monthly, and yearly granularity does not require multiple aggregations or secondary pipelines as with other technologies.

```

CREATE AGGREGATING INDEX ltv_agg_idx ON LTV(
  "ltv_timestamp_date",
  "app_id",
  "media_source",
  "ltv_country",
  "attribution_type",
  DATE_FORMAT ("ltv_timestamp_date", '%Y-%m-%d'),
  SUM("clicks_count"),
  SUM("impressions_count"),
  SUM("inapevents_count"),
  SUM("launches_count"),
  SUM("installs_count"),
  APPROX_COUNT_DISTINCT ("app_id"),
  SUM(
    CASE
      WHEN ("attribution_type" = 'install') THEN "clicks_count"
      ELSE NULL
    END
  ),
  SUM(
    CASE
      WHEN ("attribution_type" = 'install') THEN "impressions_count"
      ELSE NULL
    END
  ),
  SUM(
    CASE
      WHEN ("attribution_type" = 'install') THEN "installs_count"
      ELSE NULL
    END
  )
)
]

```

Now, if we re-run query with the aggregating index, the same query scans 2.96GB data and returns the results in 0.52s.

```
3 WITH
4   ltv AS (
5     select
6       ltv.*,
7       acc.name as acc_name,
8       acc.shipping_country,
9       acc.region,
10      app.name as app_name,
11      app.platform,
12      app.app_slug,
13      app.owner_account
14   FROM
15     ltv -- (42B rows)
16   LEFT JOIN owned_apps AS app ON ltv.app_id = app.app_slug
17   LEFT JOIN account AS acc ON acc.id = app.owner_account
18 )
19 SELECT
20   ltv.app_name AS "ltv.app_name",
21   ltv.media_source AS "ltv.media_source",
22   ltv.acc_name AS "ltv.acc_name",
23   ltv.shipping_country AS "ltv.shipping_country",
24   ltv.region AS "ltv.region",
25   ltv.platform AS "ltv.platform",
26   COALESCE(
27     SUM(
28     CASE
```

Results → 2

No	Statement	Status	Statistics
2	WITH ...	Success	10,000* 0.52 s 2.96 GB 30.51 M 100%

This example primarily shows how sub-second analytics can be achieved through proper modeling, and leveraging appropriate technologies.

Future Trends in Sub-Second Analytics

Sub-second analytics continues to evolve rapidly, driven by technological advancements and increased data democratization through data apps. This chapter will explore the future trends and developments shaping the landscape of sub-second analytics.

Edge Analytics

Edge analytics refers to performing data analysis and deriving insights at the network's edge, closer to where data is generated. This trend is driven by the increasing volume of data generated by Internet of Things (IoT) devices and the need for real-time decision-making. Edge analytics allows organizations to process data locally, reducing latency and enabling sub-second analytics in environments requiring immediate action.

Integration with Artificial Intelligence and Machine Learning

Integrating sub-second analytics with artificial intelligence (AI) and machine learning (ML) techniques is a significant trend. By combining analytics with AI/ML models, organizations can gain deeper insights and make more accurate predictions. AI/ML algorithms can analyze data, identify patterns, and make predictions or recommendations instantly. This integration enhances the speed and accuracy of sub-second analytics and opens up new possibilities for automated decision-making.

Enhanced Data Visualization Techniques

Data visualization plays a crucial role in sub-second analytics by presenting insights visually and intuitively. Future trends in data visualization will focus on enhancing real-time visualizations that can dynamically update as data changes. Interactive dashboards, augmented reality (AR), and virtual reality (VR) visualizations will enable users to explore data and gain insights, facilitating faster decision-making and improving user experiences.

Advanced Predictive and Prescriptive Analytics

The future of sub-second analytics lies in advanced predictive and prescriptive analytics capabilities. Organizations will be able to leverage real-time insights to predict future outcomes, anticipate trends, and optimize decision-making. Predictive analytics models will continuously analyze data, enabling organizations to make proactive adjustments, identify emerging opportunities, and mitigate risks before they impact the business. Prescriptive analytics will go beyond predicting outcomes and provide actionable recommendations for optimal decision-making.

Focus on Data Privacy and Security

As sub-second analytics becomes more prevalent, the importance of data privacy and security will continue to grow. Organizations must prioritize data governance, comply with privacy regulations, and implement robust security measures. Techniques such as data anonymization, encryption, access controls, and secure data sharing protocols will be essential to protect sensitive data while ensuring the benefits of sub-second analytics.

In conclusion, the future of sub-second analytics is marked by the convergence of edge analytics, AI/ML integration, enhanced data visualization, advanced predictive and prescriptive analytics, and a heightened focus on data privacy and security. By embracing these trends, organizations can unlock the full potential of data, enabling them to make faster, smarter decisions and stay ahead in an increasingly competitive world.