2022

OCIENT®

Next Generation Campaign Forecasting

The Technical and Business Case for Campaign Forecasting on Raw Bidder Log Data with Ocient®

Introduction

Ocient has partnered with leading Demand Side Platforms (DSPs) to deliver a new suite of campaign forecasting capabilities based on Ocient's next-generation SQL analytics platform. Until now, it was technically infeasible to query full resolution bidder log data in interactive time, so complex pre-aggregation and estimating pipelines were required to produce forecasts quickly. These pre-aggregation solutions add cost and have technical limitations that prevent the full breadth of analytics and ad hoc querying desired in a full-featured forecasting and bid optimization solution. With Ocient's ground-breaking architecture, built from the ground up for massively parallel use of NVMe SSDs, it is now possible to build petabyte scale applications with interactive response times that directly query raw log data with SQL.

In this paper, we will discuss the challenges and tradeoffs with pre-aggregation solutions, how full resolution SQL analytics revolutionizes the data stack for large scale data applications, and why Ocient is the best choice for a log analytics application like Campaign Forecasting.

Intended Audience

This paper is intended for a technical leader at AdTech or MarTech companies creating applications built on many Terabytes to Petabytes of data. Leaders in roles such as CTO, VP of Big Data Analytics, VP of Engineering, Director of Data Warehousing, and Product Managers who are evaluating data warehouses and analytics platforms for Demand Side Platforms, Supply Side Platforms, Advertisers, and Publishers will most benefit from this analysis.

Executive Summary

Campaign Forecasting is a challenging workload for many traditional databases due to the scale and selectivity of queries involved. Pre-aggregation offers a possible solution when query characteristics are known ahead of time, when only simpler classes of forecasts are required, and when sufficient engineering resources are available to design, build, and maintain a multi-stage data system. Ocient's Petabyte scale data warehouse and SQL analytics platform deliver a cost-saving solution that is helping DSPs deliver better forecasting tools with less development and operational effort. Ocient streamlines the loading and transformation of bidder log data, simplifies the development of forecasting features, and delivers more accurate results by operating over bigger scale than other industry leading options. Compared to brittle ETL and legacy data solutions, applications built on Ocient are easily extended for future enhancements. Ocient is a new platform built for the future. With elasticity of both storage and compute, Ocient can expand to support the growth of current workloads as well as entirely new use cases. Ocient is helping DSPs create the systems that make it easier for their advertisers to deliver campaigns most effectively.

Real Time Bidding Overview

Real time bidding facilitates spot auctions for advertisement space for advertisers and publishers whenever someone loads a webpage, mobile app, or views a digital video. Two major technical challenges faced in the RTB ecosystem are quickly processing new bid opportunity requests and analyzing the massive volume of data from these bid opportunities to plan campaigns and improve campaign performance.

Millions of auctions occur every second, and each one must occur in less than approximately 100 to 200 milliseconds to reach consumers. A DSP may evaluate 5 to 10 million bid opportunities per second. While this technical marvel is what makes it possible to hold real time auctions, in this paper, we focus on what happens after the auction has occurred. On the demand side, these raw auction opportunities, bids placed, and impression wins are stored for future analysis to improve the quality of advertising campaigns and advertiser return on investment. On the supply side, similar data is analyzed to optimize auction dynamics and yield for publishers.

Example Campaign Analyses:

- Advertisers use historical bid opportunity log data to forecast user impressions for a set of campaign targeting criteria and pricing strategies. This allows media planners to allocate their marketing budget across advertising media and customer audiences. Inaccurate forecasts can lead to wasted ad spend or lost opportunity to reach audiences.
- DSPs use historical bid opportunity data to train machine learning models that automatically adjust bid prices to increase campaign ROI. Poor models can lead to poor performance for advertisers and a loss of business.
- SSPs use historical bid opportunities to run reports or train machine learning models to adjust auction dynamics. This allows publishers to adjust auction floor prices, craft private marketplace deals effectively, and monetize first party data.

Campaign Forecasting Requirements

Campaign Forecasting assists advertisers in discovering what advertising opportunities exist and in fine tuning the delivery of their marketing budgets to the right people at the best prices. To deliver an industry leading campaign forecasting capability, DSPs need to deliver highly specific targeting capabilities with fresh, accurate results in a highly interactive user interface or API.

Data Latency

It is not critical that forecasts query real-time data, but recency is important. Digital advertising experiences seasonality similar to the broader advertising world with surges in activity around annual holidays and major buying events like back-to-school season. The internet is also a rapidly changing ecosystem, where traffic can shift based on trending patterns. Thus, data freshness of a few minutes to 1 hour is desired with a lookback period of up to 30 days. Historical data is used to project the number of impressions a campaign can win in the next 7-30 days.

Highly Specific Targeting

Advertisers need to process large amounts of raw bidder log data to create statistically accurate forecasts. Depending on campaign strategy, advertisers may cast a wide net to expand awareness or focus narrowly on needles in the haystack to drive conversion. In cases with narrow focus, campaign targeting is highly specific and uses many highly selective criteria to create a segment. Forecasts may query 50 different dimensions, each with hundreds or thousands of values. For example, a campaign may include hundreds of zip codes while combining this with contextual information on targeted sites, blocked domains, audience attributes, device, creative or placement properties such as banner dimensions or video constraints. A single day of data at 5 million queries per second is approximately 15 PB of data, so a month at this scale is almost half an exabyte! In addition, campaigns may leverage specific bidding criteria to pace budgets or cap how frequently a user can see an ad. These settings require full data resolution for accuracy.

```
SELECT
       count(distinct user_id) as distinct_users,
       count(user_id) as impression_count
FROM
       auctions.bid_opportunities
(geo_country_code IN ('US', 'CA'))
AND (
       geo_postal_code IN ('53794','53784','53774','53786','53790')
       OR geo_postal_code IN ('T1X','T1Y','V5M','V5L','V5X','L7P','M6S','R3T')
)
AND traffic_type = 'WEB'
AND device_type IN ('TABLET','PHONE','DESKTOP')
AND allowed_creative_types @> 'BANNER_IMAGE'
AND page_position = 'ABOVE_THE_FOLD'
AND banner_dimensions && array['120x600','160x600','300x50','250x250']
AND (deal_ids && array['X-1072', 'Y-5443', '5496444', 'X-7999'])
AND (
  audience_id @> 99122530
  AND NOT (
    audience_id @> 99249960 OR audience_id @> 99249362
  )
)
AND (
  (NOT (contextual_id @> 'X-196451') AND NOT (contextual_id @> 'Y-189823'))
  OR (contextual_id @> '9197502' AND contextual_id @> 'X-159486')
)
AND (
       third_party_audience_id && array[1001,1002,1003,1004]
)
```

Example Forecast Query: This query finds the count of distinct users over a few US zip codes, Canadian postal codes, specific banner and device criteria, and a set of audience and contextual values. Attribute lists have been abbreviated for readability. In practice most lists of values in this query would include dozens of values. This query could be modified to include bid price information, to frequency cap by user, or to include breakdowns by specific query dimensions such as device_type.

Interactive, Accurate Planning

Marketing campaigns combine many targeting and bidding criteria to effectively spend a marketing budget for the highest return on ad spend (ROAS). The consequences of inaccurate forecasts include both overspending and underspending. If forecasts are too optimistic, an advertiser could fail to spend budget causing them to miss top line revenue targets. On the other hand, if forecasts are too conservative, advertisers could spend their budgets prematurely or overspend on impressions leading to poor ROAS results.

Furthermore, forecast speed is critical. When planning ad campaigns, the end user may test dozens of different settings in a DSP's User Interface and combine different strategies to deploy their marketing budget. These must calculate quickly so that advertisers can explore inventory and create effective campaign strategies. A user experience rule of thumb is that analytical user interface interactions should complete in under 5 seconds to help users maintain a consistent train of thought in their workflow. Short-term memory maxes out around 10 seconds, and it is generally accepted that responses over 5 seconds require additional progress indicator feedback to demonstrate progress and reduce user frustration.[1]

Big Data Challenges in Forecasting

These requirements create a big challenge for AdTech developers. Each forecast is unique, has many dimensions, many attributes per dimension, and must evaluate a sizeable sample of data to be statistically representative. Delivering on this challenge in 5 seconds or less for the end user requires substantial engineering effort or functional tradeoffs. Statistical approaches may work in certain limited applications but can lack the required accuracy or flexibility on highly specific forecast queries.

First, we will review a few common approaches to solving these challenges along with their benefits and drawbacks. Then we will describe a new approach made possible by Ocient's innovative platform.

Common Campaign Forecasting solutions include:

1. Pre-aggregation and rollups
2. Statistical estimations using sketches
3. Custom engineered solutions

Approach #1: Pre-aggregation and Rollups

One approach that optimizes for speed at query time limits the size of data required for forecasts by pre-aggregating results for the end user in either a regular batch process or via a

[1] Nielsen, Jakob, "Response Time" Excerpt from ch. 5 in *Usability Engineering, pg 135.*
https://www.nngroup.com/articles/response-times-3-important-limits/

streaming algorithm. This approach follows the mantra to "make big data small data" and comes with significant complexity and functionality tradeoffs.

Pre-aggregation systems are designed to explicitly include a set of queryable dimensions and aggregates like an OLAP Cube. Importantly, pre-aggregating sacrifices the ability to run ad hoc queries for forecasting. Analysts are also unable to easily change the way aggregates are computed to support use cases like frequency capping. Independent sets can be unioned together to obtain overall counts, but unique and more complex metrics are not possible. This can deliver approximate results on simpler forecast queries or allow users to see basic inventory breakdowns but is not suited to an enterprise grade forecasting solution.

In practice, the intersection of N dimensions is required to compute accurate user level forecasts. For example, an advertiser may target a set of domain names, exclude a second set of domain names, target a list of hundreds of zip codes, target a few private marketplace (PMP) deal IDs, exclude dozens of contextual values, target devices, and video placement types that allow a specific video attribute. There are millions of possible domain names, tens of thousands of zip codes, tens of thousands of PMP deal IDs, hundreds of contextual values, dozens of device types and video dimensions to consider. This alone could represent on the order of 10^{20} combinations if all combinations were pre-aggregated. Because data cardinality is high in AdTech, representing this number of combinatoric rollups becomes cost prohibitive on standard databases, data warehouses, and in-memory databases. In addition, building applications on top of rollup data can make it more difficult to include new feature requests like time zone-based reporting if rollups are performed on a daily basis.

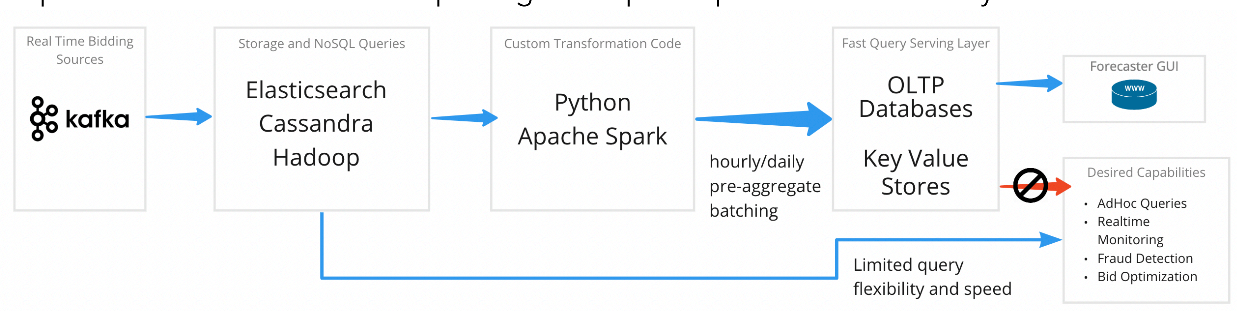


Figure 1 Example System Diagram for Pre-aggregation Based Solutions. Data streams into a NoSQL or Hadoop storage layer, then custom ETL code is written in frameworks like Spark to batch data into a "fast query" layer which may be a standard OLTP database or OLAP system.

Benefits	Drawbacks
• Quick response times when pre-aggregated value is available • Reduced storage requirements • Many databases can perform this kind of pre-aggregated lookup on modest hardware	• Poor accuracy for "real" forecast queries • Lacks the ability to query more than a couple of combinations of targeting attributes accurately

	• May require multiple data systems to handle underlying data vs rollup data queries • Less flexible solution, so new parameter combinations or capabilities difficult to introduce • Cannot provide user level data for advanced queries like frequency capping • Still requires massive storage if storing combinations of parameters that are high cardinality

Approach #2: Statistical Estimations using Sketches

A second approach to campaign forecasting is to leverage algorithms that estimate counts for different dimensions as the data is streamed into the system and query those estimates at forecast time. Different classes of sketches are useful for estimating values like count distinct, quantiles, and the most frequently occurring items. Open-source libraries, such as Apache DataSketches, have made the mathematics of this easier for practitioners, but the overall architecture to support them remains complex and tuning is required to tradeoff accuracy, performance, and cost.

Sketching is a statistical approach to estimating aggregates. Also called streaming algorithms, they have a mathematically guaranteed accuracy and perform in near constant time with minimal storage requirements by storing the result in a hashed format in a single pass so that source data is not needed to satisfy a query. If the groups of data that are queried are mutually exclusive, different sketches can be summed together. However, some sketch types like HyperLogLog (HLL) sketches cannot be combined accurately to get aggregate values where two sketches intersect. This creates significant challenges in the highly specific queries needed in Campaign Forecasting. Others sketch algorithms, like Theta sketches may be used in set operations, so they may be more useful for scenarios like Campaign Forecasting.

While many of the benefits of sketching are powerful for simple use cases, engineers should be aware that Sketching suffers similar drawbacks to pre-aggregation for "real" forecast queries. When an end-user combines many different dimensions in a query that needs to intersect query predicates, one must understand the accuracy of intersected theta sketches. Depending on the sampling size used on sketches, estimator accuracy can vary significantly. This is dominated by the sketch with the least accuracy.[2] One option is to sketch various

[2] Source: Theta Sketch Set Operations Accuracy https://datasketches.apache.org/docs/Theta/ThetaSketchSetOpsAccuracy.html

common combinations of predicates, but this quickly spirals into combinatoric problems as the number of relevant combinations that must be considered explodes.

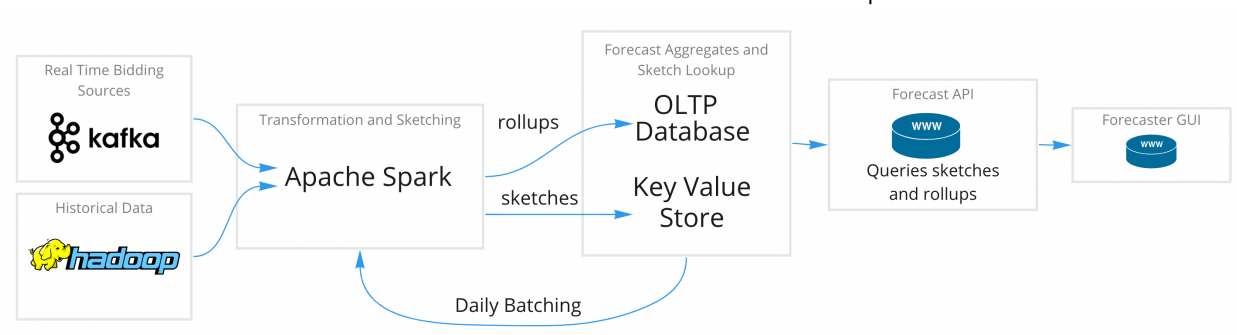


Figure 2 Sketch Based System Diagram

Benefits	Drawbacks
• Quick response times when sketch results are precomputed • Reduced storage requirements • Summability makes late data arrival and combination across multiple dimensions possible. • Query time and sketch update time scales sublinearly with sketched record count • High concurrent query processing • Mathematically known error bounds	• Poor accuracy for "real" forecast queries that require intersection • Specialized knowledge of sketching systems required for development, maintenance, and extension of applications • Lacks the ability to query more than a couple of combinations of targeting attributes accurately • Requires coordination of multiple data processing systems to handle sketch computation, storage, and querying • Less flexible solution, so new parameter combinations or capabilities are difficult to introduce • Cannot provide user level data for advanced queries like frequency capping

Approach #3: Custom Engineered Solutions

A third approach occasionally pursued in forecasting situations is a custom solution. This approach is usually a custom superset of pre-aggregation and statistical estimation. Unlike standard application Sketching, custom solutions may use special data structures like bitmap indexes to allow for rapid set operations. In general, custom solutions are useful when straightforward pre-aggregation yields too many distinct values and ingest, storage, or performance becomes challenging. Custom applications are then built on top of these performance optimizations to deliver a campaign forecast UI that is quick enough for end users to operate.

An alternative custom engineered solution involves using batch ETL processes to combine and downsample the data to a scale that is manageable by a team's existing database stack. A processing framework like Spark or Map-Reduce is used to run the ETL and the downsampled results are stored in a database like MySQL or PostgreSQL for querying. In these cases, lengthy ETL processes are scheduled on a daily or weekly basis to fully replace the working data set or to incrementally update the data. These processes grow in complexity over time and can become brittle. Additionally, the processing time elongates as ingested data scales up leading to cost and data freshness issues. Any feature requests for the forecasting application tier require extensive data engineering and backporting of historical data features before they are production ready. For this reason, the batch ETL approach can lead to lengthy delays in new feature development and eventual product stagnation. The ETL approach can also suffer from an inability to easily perform computations at an individual user or account level as the tie to underlying data is usually lost in transformation.

Like any custom solution, this can lead to technical debt that makes future extension of the framework challenging. Retention of key contributors to the project is critical for long term maintenance and upgrades. Unlike modern databases that are built for streaming data sources, any custom engineered solution would require a further investment in stream processing tooling or to settle for longer latency batch ETL processing.

Benefits	Drawbacks
• Quick response times possible for targeted subsets of queries • Full control over low level engineering details • Possible storage and hardware savings if application is well optimized • Proprietary software retains intellectual property	• Very inflexible – new features or queries require development effort or may not be possible • Challenging to adapt to changing AdTech landscape • Key contributor risk around maintaining system long term and growing technical debt • Up front expense and time to develop can be prohibitive • System accuracy and performance may vary significantly based on engineering decisions • In batch ETL flows, data processing time increases with scale leading to eventual need for replacement when data is not fresh enough or costs grow too high

Summary of Existing Solutions

In summary, existing solutions to campaign forecasting each have their shortcomings. Pre-aggregation or in-memory databases attempt to shrink the amount of data that must be handled by queries, but they fall short on query flexibility and analytical power. Sketch estimation approaches can scale well by moving processing into the streaming layer but involve complex pre-processing and lose accuracy when handling the type of complex query predicates common in forecasting. Finally, custom solutions or batch ETL workflows may solve initial forecasting use cases, but quickly become brittle in the face of frequent feature requests. What works now becomes dated and can quickly hit scale limits as data volumes increase. In the next section, we will describe a radically simpler approach leveraging raw bidder log data, enabled by Ocient's revolutionary new architecture.

A New Approach: Direct SQL Query of Bidder Log Data

Recent breakthrough innovation in NVMe SSD storage has made a new class of database applications feasible and affordable. Ocient's optimized engine is built specifically for NVMe SSDs. This design enables novel, simpler approaches to many big data challenges including Campaign Forecasting.

Rather than pre-aggregate or statistically estimate a predefined set of queries, Ocient users write SQL to directly query high resolution bidder log data. Ocient's Loading and Transformation makes streaming data available for query with a low latency on time to query (in seconds) and with interactive query response times across trillions of data points.

Why haven't AdTech firms queried raw bidder log data before now?

Before Ocient, databases could not process raw log data efficiently enough to deliver fast responses at an affordable price. Queries took minutes to return in Presto or Hive for Petabytes of bidder log data. Other columnar SQL solutions could handle small samples (~ 0.5%-1%) of bidder log data but were designed in an era when 10-100 TB of data was considered large. With these legacy data warehouses, all but the simplest queries will fail to return in interactive time. Legacy data warehouses were designed around spinning disk hard drive technology and lack the architecture and low-level code to optimize for today's NVMe SSD hard drive performance. This leads to common scale problems: higher cost, slower queries, and time-consuming workarounds.

Now that it is technologically possible to build interactive PB scale applications directly on raw bidder log data, this approach will become the new standard in many applications. Engineering managers will face a choice between augmenting legacy data warehouses or adopting a newer more highly optimized solution like Ocient. In the following section, we outline why Ocient is the best choice for full resolution analytics applications and an excellent choice for Campaign Forecasting.

Why Ocient is the Best Choice for Bidder Log Analytics

Built to scale to exabytes of data, Ocient is the only solution that has the power to deliver flexible SQL analysis on the full resolution digital trail required for Campaign Forecasting and to lay a foundation that scales cost effectively into the needs of the future. Ocient radically simplifies the system architecture for campaign forecasting and log analytics. This leads to lower total cost of ownership, quicker development of new products, richer end user experiences, and faster response times.

This best-in-class price to performance and system simplicity is made possible by Ocient's unique design:

1. Custom built data warehouse engine, optimized for commodity NVMe Hard Drives at scale
2. ANSI SQL queries with powerful windowing, aggregation, and machine learning models
3. Complex Data Types
4. Advanced Stream Loading and Transformation

Designed for Massively Affordable Scale

The guiding design principle of Ocient's architecture is to deliver massive scale data exploration at an unbeatable total cost of ownership.

Ocient is a massively parallel distributed system which stores data in optimized, partitioned, indexed columnar data structures optimized for extremely high disk input/output operations (IOPS). Ocient Foundation Nodes run highly optimized C++ code that leverages the incredible throughput of commodity NVMe Hard Drives to achieve random read data rates that rival DRAM at a fraction of the cost ($0.20/GB vs $5/GB). Ocient's ZeroCopy Reliability™ uses parity coding of data across the Foundation Nodes to further reduce the cost of data storage relative to redundant copy strategies. This allows Ocient to operate on 20%-40% of the storage of a copy-based system. This cost advantage is multiplied when disaster recovery systems are added.

The cost of NVMe Solid State Drives (SSD) is expected to continue decreasing as they are applied in broader applications and supplier scale increases. Rob Cooke, SVP and GM Intel NAND Products and Solutions Group illustrated this trend in the Intel Memory and Storage Briefing from December 2020 found at https://www.intel.com/content/www/us/en/events/memory-and-storage.html. Intel projects that the total cost of ownership (TCO) of SSDs will be lower than HDDs in 2022. This will drive Ocient's cost advantage over traditionally spinning disk-based technologies like Hadoop or systems based on object storage as Ocient optimizes database performance on storage adjacent to compute.

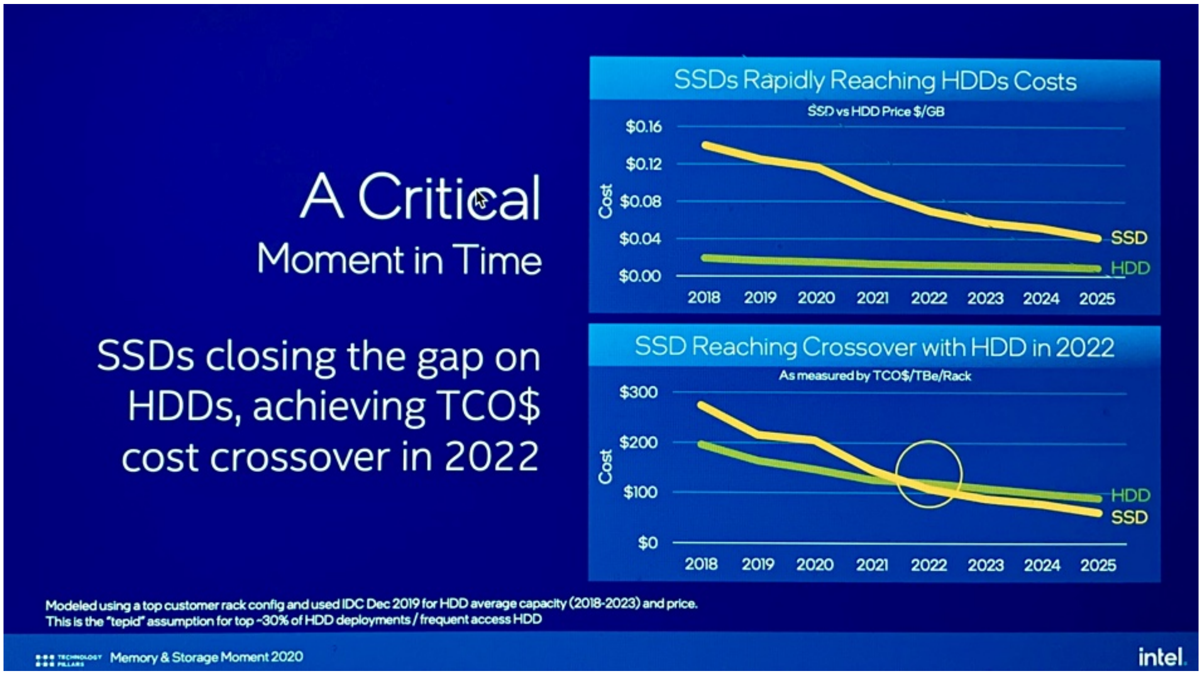


Figure 3 Intel illustrates TCO crossover for SSD[3]

NVMe SSDs are also being adopted as the storage layer in other data stacks as the costs decrease. As detailed in the "Ocient Cost vs Performance Deep Dive" section, while other databases try to "bolt on" NVMe SSDs, Ocient is highly optimized for this breakthrough technology creating high performance that legacy systems cannot touch. Key architectural decisions and low-level engineering create the efficiency at the core of Ocient's ability to scale cost-effectively. At petabyte and exabyte scale, these efficiencies compound into substantial speed gains and cost savings as fewer nodes, fewer cores, less memory, and lower IT management costs are needed to deliver the required performance.

[3] Source: Intel Storage and Memory Moment 2020, Rob Crooke, Senior Vice President and General Manager, NAND Products & Solutions Group Video. https://www.intel.com/content/www/us/en/events/memory-and-storage.html

Ocient Cost vs Performance Deep Dive

NVMe Native

NVMe SSD drives have revolutionized the storage stack. SSD drives were introduced 10 years ago, but the SATA interface constrained the speed to 70K IOPS and 4Gbps because of the lack of parallel access. It was only with the introduction of NVMe in 2015 that SSD drives have been able to achieve much high levels of throughput. NVMe enabled this performance disruption achieving throughput of 25Gbps with 800K IOPS through parallel access and much higher queue depths. Queue depths greater than 256 are required to saturate most NVMe drives. Many servers have up to 8 NVMe drives in them, so the software layer needs over 2,000 requests in flight to saturate an 8 drive NVMe system. All databases designed prior to NVMe drives were not designed for this level of queue depth. By focusing software on the new parallel characteristics of NVMe drives, Ocient has optimized its software to realize the full performance from this storage disruption.

NVMe Direct

Most databases access the hard drive through a filesystem and a kernel mode driver. This layer of abstraction was acceptable when using spinning disks, but as hard drives have increased their performance, it has been increasingly expensive to keep this level of abstraction. As a result, Ocient has removed this abstraction and has bypassed the kernel to read and write directly to NVMe drives through the PCI bus. This direct level of access dramatically lowers the number of context switches and memory copies that need to occur in the data path. This dramatically frees up the memory bandwidth and CPU usage for other tasks.

Parallelism

Until relatively recently, increases in computational or data processing speed were made by increasing the speed of processing elements such the CPU clock rate. In the foreseeable future, increases in total computational throughput are realized less through increasing the speed of single components and more through having multiple components acting in parallel. Achieving peak throughput and performance from NVMe storage, high core count processors, and modern memory systems requires software that has been designed to interact with and consume these components in a highly parallel manner. The Ocient database engine has been designed in just this way, with a significant focus on this parallelism to fully capitalize on the capabilities of the modern hardware on which it runs. This includes processing patterns that view individual CPU cores as parallel data channels, storage structures designed to effect parallel IO requests, and data flow schemes that allow the engine to maintain highly parallel flow across process and network boundaries.

This sort of highly parallelized design must be built into the engine at its lowest levels and cannot simply be "bolted on" after the fact. Most existing database systems were designed in a world where serial spinning disk technology was the only viable bulk storage available and with internal structures that don't align with the parallelism exposed by processing elements. As a result, they are often fundamentally incapable of fully utilizing the resources available. Since future increases in computational throughput will be brought about by increases in parallelism at all levels of the hardware, existing database systems will be even less capable of capitalizing on this increased throughput. In contrast, the Ocient database engine, due to its highly parallelized design, will be able to scale its processing throughput in line with future increases in hardware capabilities.

Optimizer Optimizations

Optimizing database queries is an essential part of a relational database or data warehouse. The number of permutations available for complex queries is endless. Finding an optimal solution has dramatic impact on query speed.

The Ocient optimizer uses multi-dimensional probability density functions, instead of histograms, to calculate selectivity. In general, this allows the optimizer to compute much more accurate selectivity estimates and to understand correlation between columns in intermediate results. This helps the optimizer to cost out alternative plans. Additionally, combining a rich set of runtime operators and novel stochastic optimization techniques allows Ocient to generate execution plans for queries that would be impossible or never found by traditional database systems.

Finally, the Ocient optimizer has a model of the network and hardware infrastructure layer to manage the optimizer and build a model to find the most efficient execution path for the query.

Highly Optimized C++

Ocient is built in C++ with key performance optimizations to maximize processing efficiency and throughput.

- Ocient makes heavy use of C++ template capabilities that speed up the code paths during processing and tend to minimize cache misses on the processor.
- Ocient has also designed its algorithms and structures to further minimize cache misses.
- Ocient uses huge pages in Linux and has written custom memory allocators for different algorithms that run frequently. By avoiding fragmenting memory and using the knowledge of how the algorithms are going to use memory, memory bandwidth and CPU utilization are dramatically improved.

Zero Copy Reliability™
Many data systems rely on replicating data in triplicate to have the ability to recover data in a hardware loss scenario. Ocient has developed proprietary Zero Copy Reliability™ approaches that use parity encoding techniques across the Ocient storage cluster to ensure data reliability. System architects can choose a preferred level of reliability based on the data parity width. At scale, the costs of reliably storing 100's of PB of data are reduced from 200% to only 15% - 25% additional space required to ensure reliability.

Power and Flexibility of ANSI SQL Queries at Scale

Built on the Foundation Nodes is a SQL processing layer that uses probabilistic query planning to optimize queries and execute plans. Ocient SQL looks exactly like familiar SQL dialects like PostgreSQL so the learning curve for new engineers is essentially zero. Ocient SQL supports the full suite of ANSI SQL capabilities through industry standard JDBC or ODBC drivers and works easily with common visualization and data tools. While NoSQL databases, Spark ETL jobs, or Map-Reduce languages are challenging to write well, SQL can easily perform powerful windowing functions, aggregates, joins, set operations, and even machine learning functions within the database. For example, Ocient allows the automated creation and training of a multiple linear regression model and the subsequent querying of data against that model to make predictions.

Campaign Forecasting with SQL brings the flexibility to add new features such as hourly frequency capping with a single query. Ocient's rapid response times allow this query to run on the fly for a user so no pre-aggregation or ETL is required. This streamlines the prior workflow which required the addition of new steps to the ETL flow, complex aggregation steps to handle user-based computations, batching delays, and the possibility that results would be inaccurate in the end. Simplifications in the data stack like this reduce the amount of code and number of systems involved in creating insights that lead to faster engineering team velocity.

Optimized for Complex Types

Complex Data Types are essential to high performing Campaign Forecast queries. Ocient can deliver blazing fast performance at low cost due to its complex types such as Arrays, Matrixes, and Tuples that allow nested data structures to be queried with amazing efficiency. Expensive query time joins are easily avoided due to this support, and rich set operations are also made more efficient for Campaign Forecasting applications. Ocient has also added indexing strategies on these complex types that deliver even faster query performance.

In Campaign Forecasting, deep lists of attributes such as audience segments or contextual data attributes can number in the hundreds or thousands depending on the data partnerships. When querying tens of billions of bidder log events, the total entries in an array column could number in the tens of trillions. Ocient's phenomenal disk throughput on Foundation Nodes and Complex Data Type optimizations make forecast queries possible in interactive time. Competing technologies often require joins or runtime parsing of string representations of Array, Tuple, or JSON data types. These approaches introduce excessive query processing overhead that leads to slower queries and higher compute bills.

Ocient also has other performance optimizations that benefit AdTech solutions:

- Geospatial data types such as points, lines, and polygons to deliver geofencing capabilities
- High performance IP Address functionality
- Varchar column indexing strategies optimized for domain names and URLs
- Join optimizations using Bloom filters and Semi-join

Advanced Stream Loading and Transformation

Ocient's massively parallel stream loading system can stream transform and load hundreds of thousands of records per second per node with no complex stream engineering. This gracefully scales out horizontally to accommodate the largest data volumes in bidder log systems. When a 100% sample is not required, it can scale back and operate in a sampled mode to save costs.

Data is received in Ocient in standard formats such as JSON, AVRO, and TSV. As the data is streamed, Ocient's Loading and Transformation system uses transformations based on familiar SQL syntax to transform incoming data streams into tabular data. Transformed data is delivered to Loader Nodes that partition, order, and index the data for optimal storage on the Ocient Foundation Nodes. Loader Nodes are decoupled from the storage system such that they can scale horizontally to accommodate the ingest rate separately from storage and querying requirements. Ocient seamlessly parallelizes incoming streams onto available Indexer Nodes, maximizing throughput.

Data is made available for query within seconds. This low latency availability of data for queries is critical in many AdTech applications. Ocient's Loading and Transformation system handles streaming and batch jobs alike and leverages industry standard queueing systems like Apache Kafka to provide exactly once processing. Finally, Ocient allows for dynamic schema modifications while streaming to ensure that data migrations are reliable and effortless to deploy to production.

In historical scenarios, large volumes can be batch loaded at high rates to initiate analyses and support new data exploration. In a continuous streaming scenario, an N+M architecture is used for Loader Nodes to provide high availability uptime of loading systems. The addition of new streams of data is as simple as publishing a stream to a Kafka topic and writing a single text transformation so new data streams can be effortlessly onboarded by data engineers or database administrators.

Benefits of Ocient for Bidder Log Analysis

Ocient's simplified approach replaces multiple ETL, storage, and processing applications and hundreds of servers with a streamlined data analytics cluster. This simplified architecture benefits many types of users. Key benefits include:

- Query Performance with minimal tuning
- Simplicity for developers – improving time to market and reducing development costs
- Massive scalability and speed that future proof data infrastructure
- Serving mixed workloads on a single system
- Cost benefits that continue to increase at scale due to storage innovations and optimized processing

Role	Simplification Benefits
Data Engineers	• Transforming and loading data directly in Ocient reduces effort for data engineering projects • Highly scalable loading processes
Database Administrators	• Shared workload environment with advanced workload management settings • Limited tuning required due to optimized storage and compression strategies
Software Engineers	• Build petabyte scale applications directly using SQL dramatically decreases project complexity • Eliminates lengthy data prep and engineering steps to condition data for interactive applications • Ability to handle massive datasets eliminates pre-aggregation and need for batch processes
Data Scientists	• Reduce time to run machine learning tasks • Rapidly explore raw data with SQL instead of less expressive or flexible query languages • Perform powerful analytical windowing and statistical queries in SQL
IT/Developer Operations	• Streamlined hardware requirements with commodity parts for low cost, high availability implementations • Fewer system interconnections and failure points and interconnections • Simplified system monitoring via unified platform
Data Protection and Governance Managers	• Reduced number of external data processing systems streamlines data protection efforts

	• Built in transformation simplifies data lineage questions • Strong user access controls
Engineering Leadership	• Simplified data stack requires less specialized talent allowing more team members to participate in active data development and easing recruiting challenges. • SQL reduces training time to bring team members up to speed vs Spark/Map-Reduce/NoSQL queries

Designed for Diverse Enterprise Workloads

On Ocient, an analyst running an exploratory query on all historical data that will take minutes to execute and a customer facing GUI running a query that must return in seconds can operate simultaneously. These mixed workloads are made possible on a single system through sophisticated workload management and resource limit capabilities. Database administrators can limit the number of rows returned, max elapsed time for a query, max concurrent queries in a service class, and throttle access to system resources like disk spill space. Administrators can also assign priorities to different service classes to ensure that higher priority workloads are prioritized by the query processing engine.

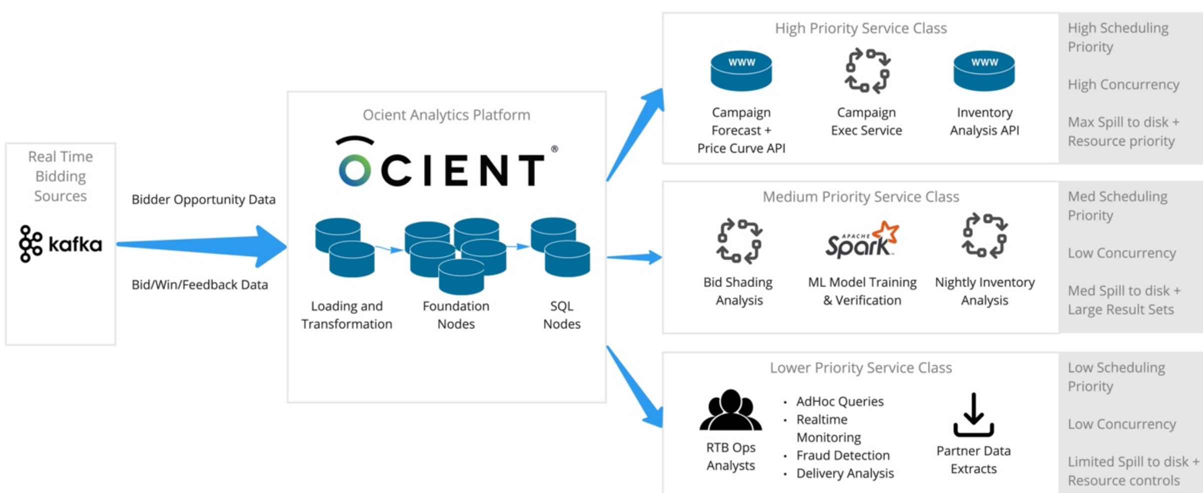


Figure 4 Example Mixed Workload Implementation in AdTech. High priority workloads are allowed to run more concurrent queries and are scheduled with higher priority so that they perform at the highest level. Background processes and lower priority analyst users receive fewer resources and limited concurrent queries which limits their impact on critical performance workloads.

Designed for Enterprise Security and Privacy

Ocient also provides fine grained access control to allow data controllers to assign rights to access and manage data. These rights are integrated with Single Sign On (SSO) providers to allow enterprise IT teams to easily manage which users, roles, and groups can interact with which data. Ocient's monitoring solutions allow complete auditing of user access and activity providing essential tools to data security organizations.

AdTech Beyond Campaign Forecasting

Ocient's streamlined big data SQL architecture is well suited for applications like Campaign Forecasting where interactive response times are required at PB scale. However, Ocient goes beyond these use cases to serve a broader array of more traditional OLAP style workloads while enabling new possibilities. Within AdTech, Ocient is well suited to deliver outstanding performance per dollar:

- Campaign and Private Marketplace Deal Troubleshooting
- Supply Side Yield Optimization
- Bidding Optimization
- Impression Analysis and Attribution
- Campaign Reporting
- User Profiles and Identity Resolution
- Machine Learning and Data Science Rapid Exploration
- Large Scale reporting over billions to trillions of records

Ocient Campaign Forecaster Case Study

In this example, Ocient was installed to replace ElasticSearch as a campaign forecasting solution for a DSP. ElasticSearch had performed adequately at a small scale (0.5% of bid opportunities) but was too expensive to operate at the scale the DSP desired. A 50 node Elastic Search cluster was required to serve forecast queries on demand, to support nightly aggregation queries and to allow ad hoc explorations by the operations team and analysts.

The Challenge

ElasticSearch was configured to consume a sample of the bid opportunity stream of JSON messages. Indexes were applied across most columns to improve query performance. Queries needed to return in less than 5 seconds on average to provide an interactive experience to end users. Data freshness of one hour or less was preferred by product management with a 7-day lookback period. Daily aggregation jobs were created to digest data from ElasticSearch and store it in MySQL, a relational database tier. This allowed pre-aggregates to be frequently queried, but some pre-aggregate sets were still large enough that developers avoided querying the pre-aggregates unless strictly required. The system setup worked well at smaller scale, but had the following drawbacks:

1. High cost of operation limiting forecast accuracy: The DSP was managing a 50 ElasticSearch node cluster and wanted to increase the amount of data in the system

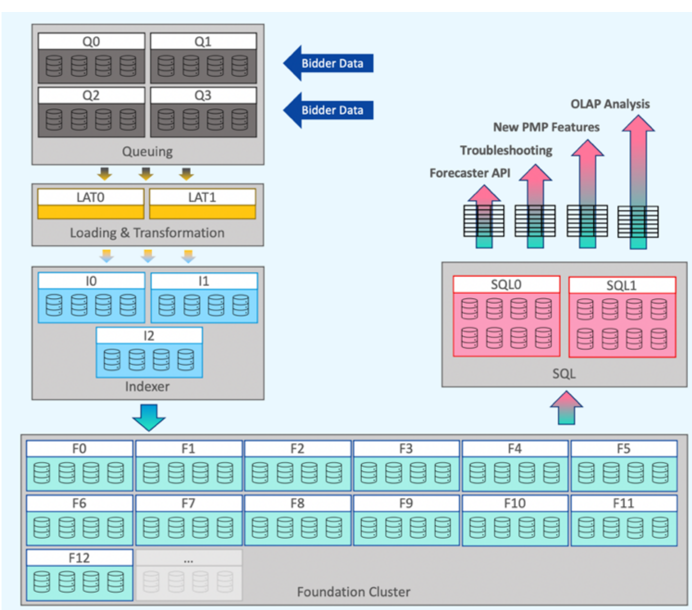
by 2x-3x. The cost to handle this scale was too high to justify a ROI to the business, so they had to settle for less accurate analyses.

2. Query Syntax: Difficult to write EQL queries against highly nested document structure with arrays, arrays of objects, etc.
3. Analysis Limitations: Searching for record counts was simple, but more complex grouping, aggregation or sophisticated analyses that needed to run across joined records or sets not possible.
4. Operations: Regular reindexing was a slow operation on production applications and complicated production updates.
5. Crosstalk: No easy way to limit the impact a resource intensive query would have on the overall system. Troubleshooting queries from analysts could negatively impact other customer facing applications.
6. Time to Queryability: for some queries that merely retrieved records, time to queryability was adequate (< 1 hour), however for queries relying on pre-aggregation of results, fresh results were not available. In some cases, aggregated data was 24-36 hours old before being made available to query due to batching.

The Solution

Ocient engaged the DSP in a pilot project to demonstrate Ocient's extreme performance on the customer's live application at larger scale.

The pilot Ocient system delivered response times with a 4.6 second average response time across tens of thousands of query variations while loading 2.5x more data into the database cluster. The pilot system, illustrated below, used 13 Foundation Nodes with 8x 4TB drives each for a total raw storage of 416TB. Due to compression performance of the data, this system can store approximately 1.2PB of raw data.



Queuing	Receive, protect, queue.
Transformation	Parse and transform Bidder Data in preparation for indexing.
Indexer	Accepting multiple, simultaneous streams of Bidder data to protect, deduplicate and index - made available to query within seconds
Foundation	Redundant storage and processing of data.
SQL	Provides access for forecasting, troubleshooting and analysis.
Network	10 + 100Gbps Networking

Node	Cores	Memory	Storage	Qty
Queuing	32	192GB	12 x 10TB HDD	2
Transform	52	192GB	-	2
Indexer	52	1.5TB	8 x 2TB NVMe	3
Foundation	52	1.5TB	8 x 4TB NVMe	13
SQL	52	768GB	4 x 2TB NVMe	2
Total Core Count			636	
Raw Storage			416TB	
Protection			8 Data + 2 Parity + 3	

Figure 5 Example Ocient Pilot System Design

Data Engineering

Data was streamed into Kafka on four topics at an average rate of 42,000-52,000 documents per second. Average document size on the topics varied from approximately 25kB to 60kB in size. These topics were transformed and loaded into tables using the Ocient Loading and Transformation system, delivering data latency of < 5 min. Ocient's transformation capabilities were able to transform one JSON document into many different tables and to explode rows in the JSON document such that many rows are generated per document in some tables. Beyond this, scalar transformation functions were used to clean data and derive columns in the final schema.

Query Workload

Campaign Forecasting was the focus of the pilot, including over 20,000 different variations of campaign forecast queries. Simpler queries counted bid opportunities for a small set of targeting criteria. More complex queries included up to tens of thousands of filter criteria. Many of these were unable to execute on the original ElasticSearch system but performed well on Ocient. Beyond counts, forecast queries also analyzed the bid and win pricing information to generate pricing recommendations. Forecast queries typically analyzed 7 days of data examining 20-30B auctions at a time. Finally, a set of other analyses that joined more auction data to perform inventory analysis, bid shading analysis, and bid quality assessments were run in nightly batches.

Data Validation

Ocient collaborated with the customer quality assurance team to validate that the Campaign Forecasting query results on Ocient matched the results from the original application. Once the solution was validated, the customer switched their Campaign Forecasting solution to begin using Ocient.

Performance Validation

During the design process, Ocient's performance team evaluated different schema alternatives and identified the best option for fast query response times on the most critical Campaign Forecasting queries. This involved changes to the clustering keys and application of secondary indexes over a few of the largest array columns that store audience and contextual identifiers in the schema to deliver an average query response time under 5 seconds.

Enabling Forecast Improvements

During this process, the flexibility and speed of Ocient enabled additional extensions to the forecasting application including:

- Forecast breakdowns showing the distribution of opportunities across key dimensions
- Advanced bid shading model support
- Rapid inventory analysis on more complex dimensions defined in SQL

Seamless Company Adoption

Ocient pilot projects are designed to lead immediately to production ready deployments. As soon as the pilot period completed, Ocient and the customer moved the pilot cluster directly into production. In addition, because SQL was well known by other engineers, analysts, and operations teams at the company, adoption for other uses proceeded immediately upon moving from the pilot to the production deployment and continuous operation for the DSP's customers.

Conclusion

Recent breakthroughs in storage technologies have opened new possibilities for analytical applications like Campaign Forecasting. Ocient is pioneering a simpler approach to analyzing massive, high-dimensional, high-cardinality data sets. This approach reduces engineering effort, speeds up delivery of new products, increases query accuracy and flexibility, and saves costs. Software efficiencies and a unique architecture, built for NVMe SSDs, allow Ocient to deliver outstanding query performance at an industry leading total cost of ownership. With elasticity of both storage and compute, Ocient can expand to support the growth of current workloads as well as entirely new use cases. In the rapidly changing world of AdTech, Ocient is a clear contender for a better way of doing business.

For more information on Ocient's Campaign Forecasting Solution, please contact our Sales Team at sales@ocient.com