AWS Architecture Monthly

July 2020 **Advertising & Marketing**

Notices

Customers are responsible for making their own independent assessment of the information in this document. This document: (a) is for informational purposes only, (b) represents current AWS product offerings and practices, which are subject to change without notice, and (c) does not create any commitments or assurances from AWS and its affiliates, suppliers, or licensors. AWS products or services are provided “as is” without warranties, representations, or conditions of any kind, whether express or implied. The responsibilities and liabilities of AWS to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

Editor's note

With 200+ services and 15 years of proven expertise, AWS has everything you need to build faster, operate cost-effectively, and run virtually any advertising and marketing technology workload at petabyte scale and millisecond latency.

For this month's issue, we spoke to three Advertising & Marketing experts about the general architecture pattern trends in this sector as well as their ideas about the role will cloud play in future development efforts. You'll see a case study from The Trade Desk and how it's transforming its real-time bidding strategy on AWS, and well catch you up on service launches and announcements from re:Invent 2019.

I hope you'll find this edition of Architecture Monthly useful, and my team would like your feedback. Please give us a star rating and your comments on the Amazon Kindle page (https://amzn.to/Kindle-magazine). You can view past issues at https://aws.amazon.com/whitepapers/kindle/ and reach out to aws-architecture-monthly@amazon.com anytime with your questions and comments.

In the July 2020 Advertising & Marketing issue:

- **Ask the AWS Experts in Advertising & Marketing**:
    - Gerry Louw, Worldwide Technical Lead
    - Clark Fredricksen, Head of Worldwide Marketing
    - Dmitri Tchikatilov, Head of Worldwide Business Development
- **Case study**: The Trade Desk Transforms its Real-time Bidding Strategy with AWS
- **Reference Architecture**: The Trade Desk Transforms its Real-time Bidding Strategy with AWS
- **Blog**: Building a Customer Identity Graph with Amazon Neptune
- **Blog**: Serving Billions of Ads in Just 100 ms Using Amazon ElastiCache for Redis
- **Solution**: Real-Time Web Analytics with Kinesis Data Analytics
- **Announcements**: Key announcements for the Advertising and Marketing Industry (AWS re:Invent 2019)

Annik Stahl, Managing Editor

What are the general architecture pattern trends for the Advertising and Marketing industry?

Customers in this industry typically have scaled data processing requirements. Take advertising analytics, where companies need to ingest terabytes or petabytes fine-grained data per month about digital events such as ad impressions or ad clicks. The resulting architecture patterns usually require streaming data with Amazon Kinesis or Amazon Managed Service for Apache Kafka (MSK) into a data lake on Amazon S3. Customers traditionally run batch processes at periodic intervals for other types of data, such as advertising or marketing partners pushing web events, purchase events, or demographic data. We've recently seen rapid adoption of serverless technologies for data processing, which simplifies the logic and makes it more event-driven. Customers use serverless technologies to coordinate data flows and transformations with tools like AWS Glue, and Apache Spark on Amazon EMR. We also see customers using containers with Amazon Elastic Container Registry (ECR) and storing data as time-stamped, partitioned parquet files that are organized to simplify the discovery of information. Once the data is in a secure, well-organized, and partitioned form on S3, it can be easily consumed by a variety of analytical services like Amazon Redshift or other interactive querying services like Amazon Athena.

One of the most unique architectures in the industry comes from Real-Time Bidding (RTB), an auction process in which advertising technology firms representing large marketers participate in auctions to buy or sell advertising space in ultra-low latency and mind-blowing throughput.

Performance at scale and cost are the two main business drivers for architectural patterns behind this workload. Typical throughput for a RTB workload is between 500,000 queries per second to over 12 million queries per second. Ad Tech companies also must respond to partners within tens of milliseconds or lose an opportunity buy an ad. Most ad tech firms have a low-latency NoSQL data store such as Aerospike or Amazon ElastiCache for Redis that retrieves user-profiles, audiences and bidding information. Low-latency read and write performance are critical for these data stores, since ad techs receive millions of bid requests per second and write new consumer data continuously into bidding databases. RTB workloads also include bid logging and user information, which is streamed toward a centralized analytics pipeline. Customers often run RTB VPCs with bare metal instances from I3, M5, M6, and R5 families for optimal performance. Customers also often choose a

stateless architecture pattern on their log pipelines so they can take advantage of Amazon EC2 Spot Instances without worrying about availability, and they can ultimately save upwards of 25% on compute for bidding workloads.

What are some questions organizations need to ask themselves before considering AWS?

We recommend doing a detailed cost analysis of compute, storage, database, and networking requirements for advertising and marketing workloads on-premises. However, comparing cloud and on-premises architecture usually isn't apples to apples, especially for scaled workloads like marketing automation, advertising analytics, or RTB. The main cost savings in the cloud comes from lower cost of compute due to elasticity, serverless technologies, and purpose-built databases. We also suggest working with AWS Cloud Economics to model those costs on AWS. There are many good examples of customers who migrated advertising and marketing technology workloads to the cloud and saw cost savings. In this industry, customers like DataXu migrated their analytics to AWS and reduced costs by 70% compared to colocation, and Quantcast reduced costs on RTB by 25%.

Beyond cost, the most important question to ask is "How fast could I be launching new products, adding new features that drive revenue, or expanding into new markets?" Moving to the cloud can enable industry customers to onboard new clients or open a new region more quickly compared to waiting to set up a new data center. A good example is The Trade Desk, which opened four new cloud-based RTB sites in 2019 on AWS. It took them 2-3 weeks to set up those sites compared to up to 6 months for managed physical sites. Another is AdRoll, whose CTO Valentino Volonghi said, they "have the flexibility to migrate to different regions or availability zones in less than one hour in case of emergency."

When putting together an AWS architecture to solve business problems specifically for an A&M company, do you have to think at all differently?

Architectural decisions in this industry are driven by which services can handle scale at the lowest possible cost. What's different about AWS is that you have much more choice across services for compute, networking, databases, analytics, and machine learning—and that breadth of choice enables great cost-efficiency compared to other cloud providers with fewer options. For example, companies in marketing technology build identity graphs to store consumer profile data for billions of users. Depending on their requirements, they can choose between a graph database like Amazon Neptune or a NoSQL store like Amazon DynamoDB. You could choose to build a streaming pipeline with Kafka (Amazon MSK) or Amazon Kinesis; or you might only need S3 and Amazon Batch Processing. The advantage of AWS is you can pick the right tool to match your requirements—so you only pay for the capacity and functionality you need.

Do you see different trends within the industry in cloud versus on-premises?

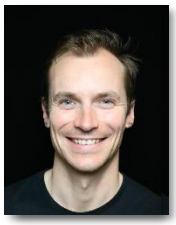
Workloads like RTB or marketing analytics pipelines require high availability, 24/7. Customers with on-premises workloads often overprovision to account for spikes in traffic and load; whereas the cloud has elastic capabilities and tools like auto scaling that can significantly reduce provisioned compute costs. On-premises data centers running RTB tend to see more stateful, static workloads, with servers running on bare metal because there's less opportunity for elastic auto-provisioning and no spot market for extra capacity. In the cloud, customers often will adopt a microservice approach with stateless bidding architecture that enables them to scale independent functions as needed and use EC2 Spot Instances to save on compute costs without worrying about availability. The result is usually that customers running industry workloads in the cloud end up seeing 20%-70% reduction in compute costs alone.

What's your outlook for AWS in the A&M industry, and what role will cloud play in future development efforts?

Overall, the pace of innovation in the industry has accelerated in the last few years, especially as companies use the cloud for rapid experimentation or deployment of new initiatives. Good examples are The Trade Desk, which set up net-new RTB sites in just 2-3 weeks in Singapore and Frankfurt and Nielsen, which built a cloud-native National TV Ratings systems with a 30PB data lake tracking 30 million households "at a much faster pace, at a greater velocity than ever before" with Amazon S3, Amazon Redshift, and AWS Lambda. Connected TV (CTV), OTT, and video advertising are other areas. Firms are adding new CTV inventory with workloads like server side ad insertion using AWS Elemental MediaTailor and using Amazon SageMaker to improve ad effectiveness using dynamic creative optimization, fraud detection, and contextual analysis.

About the Experts

Gerry Louw is Worldwide Technical Leader for the Advertising & Marketing Industry at AWS. Prior to joining AWS, Gerry was CTO at Smaato, a global ad tech company.



Clark Fredricksen is Head of Worldwide Marketing for the Advertising and Marketing Industry at Amazon Web Services (AWS). Before joining AWS, Clark spent a decade at research firm eMarketer, where he sat on the company's executive management team and held leadership roles in product development, marketing, and communications.

Dmitri Tchikatilov is Head of Worldwide Business Development for the Advertising & Marketing Industry. He has been with AWS for 6 years in business development and solution architecture roles covering advertising technology. Prior to AWS, Dmitri held leadership positions at Microsoft and earned a Ph.D. in Engineering from Columbia University.

Online at: *https://amzn.to/AWS-AM-TradeDesk*

The Trade Desk (https://www.thetradedesk.com/) is an advertising technology company that provides a self-service platform through which media buyers can purchase digital advertising. The company recently began shifting its real-time bidding workload—which receives 10 million queries per second, 800 billion queries per day, and requires an average response-time of less than 15 milliseconds—to the Amazon Web Services (AWS) Cloud, building four new sites in 2019.

It used to take The Trade Desk take up to six months to build out a bidding site for a new market. But, after migrating its bidding workload to AWS, the company has cut build-time to less than a week and production deployment to about three weeks. “There’s much more agility. If we need to tune a site, we can make changes much more quickly. We’ll do more iteration in a single week than what we would do in a full year with a managed, physical site,” says Zak Stengel, senior vice president of engineering for The Trade Desk.

(See the next section for The Trade Desk’s reference architecture for real-time bidding in the cloud.)

Real-world example

The Trade Desk: Real-time Ad Bidding in the Cloud with AWS Global Accelerator

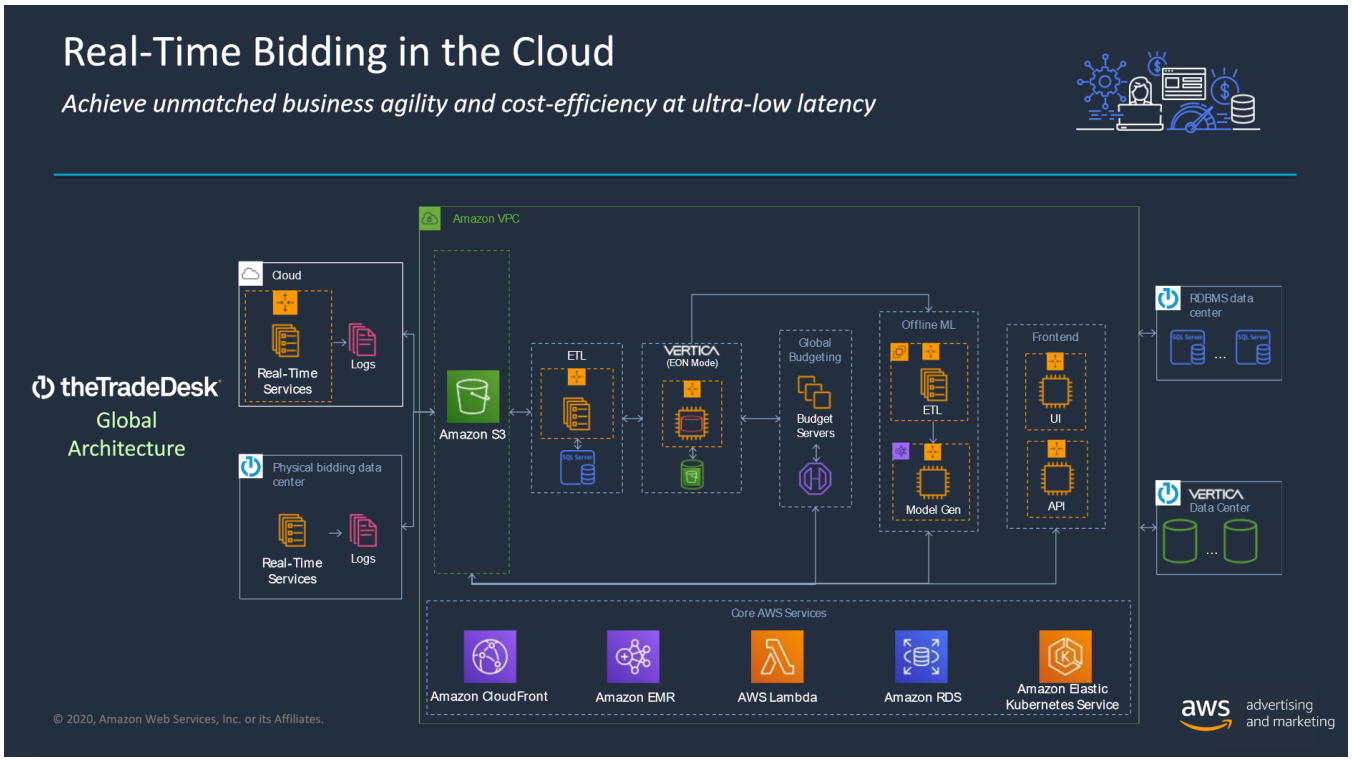
Learn how The Trade Desk built a global, real-time ad bidding system on AWS without having to implement major architectural changes, thanks to its use of AWS Global Accelerator. The Trade Desk processes millions of messages per second while keeping latency low, and can scale to handle many gigabits of data per second on the back end. You'll also learn how the company uses Amazon S3-backed Vertica with Eon Mode for long-term storage and analytics with petabytes of data.

https://amzn.to/AWS-AM-TD-TMA

Reference Architecture:

The Trade Desk:
Real-Time Bidding in the Cloud

If you are reading this on Kindle, and would like to see a larger version of this architecture, please download the PDF of the July magazine: https://aws.amazon.com/whitepapers/kindle/.



Blog:
Building a Customer Identity Graph with Amazon Neptune

By Rajesh Wunnava and Taylor Riggan

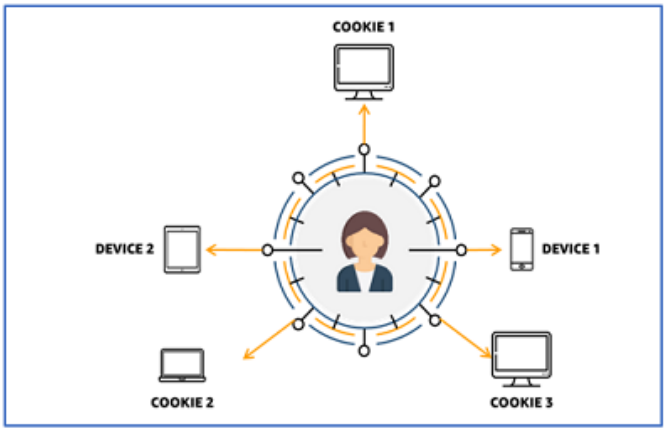
Read the full post online at: *https://amzn.to/AWS-AM-neptune*

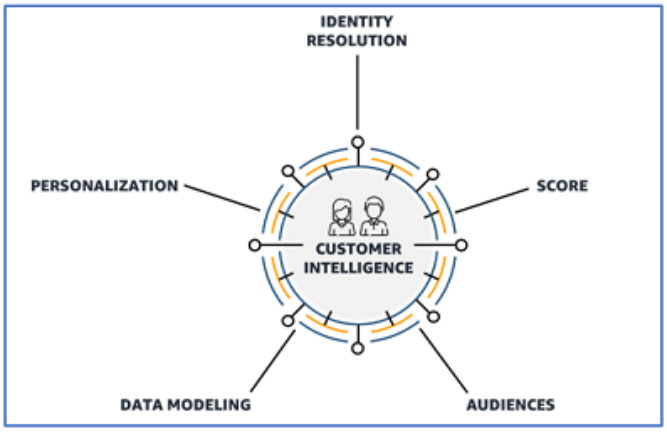
A customer identity graph provides a single unified view of customers and prospects by linking multiple identifiers such as cookies, device identifiers, IP addresses, email IDs, and internal enterprise IDs to a known person or anonymous profile using privacy-compliant methods. It also captures customer behavior and preferences across devices and marketing channels. It acts as a central hub and enables targeted advertising, personalization of customer experiences, and measurement of marketing effectiveness.

This post provides an overview of how to build a customer identity graph on AWS. It reviews key business drivers, challenges, use cases, customer success stories and the benefits of the solution. You also walk through the solution, sample data model, AWS CloudFormation templates, and other technical components that you can use to kick-start your development.

The following diagrams illustrate the collection of data around a given user, such as device identifiers, cookies, browsers, and behavior in a customer identity graph platform to enable identity resolution, scoring, and creation of audience segments for personalization.

The following diagrams illustrate the collection of data around a given user, such as device identifiers, cookies, browsers, and behavior in a customer identity graph platform to enable identity resolution, scoring, and creation of audience segments for personalization.





You build the solution in Amazon Neptune, a purpose-built graph database for the cloud. It's ideal for storing and navigating billions of interconnected relationships and supports millisecond latency for real-time advertising and marketing applications. The solution also uses Amazon SageMaker, a fully-managed platform for building, training, and developing machine learning models. For this solution, you use Amazon SageMaker for its ability to provide hosted Jupyter notebooks for loading customer identity graph data, and query it for a few common use cases.

Privacy-compliant customer experiences

Marketers, advertisers, and digital platforms must identify, understand, and anticipate customer needs and personalize experiences at scale using privacy-compliant methods.

Delivering on these expectations is challenging on many fronts. From a business standpoint, it entails aggregating data from enterprise silos across marketing, sales, loyalty, and others. From a technology standpoint, it requires a secure and flexible database platform that can scale globally to continually maintain a real-time customer identity and behavior graph for billions of interconnected relationships between devices, customer identifiers, channels, and preferences.

Building a customer identity graph solution on AWS

The customer identity graph solution provides a reference application so you can build a cost-efficient, scalable, secure, and highly available customer data platform with your own proprietary business rules. You can respond to customer signals in real time to automate your advertising and marketing applications and customer journey orchestration.

The solution enables marketers, ad-tech, mar-tech, gaming, media, and entertainment companies to capture and activate insights in real time from billions of relationships for millions of customer profiles. Customers like Zeta Global, NBCUniversal, and Activision Blizzard use Amazon Neptune to build identity graphs and capture consumer journeys to personalize advertising, content, and in-game experiences for millions of users.

This solution includes a sample data model, CloudFormation template, and Amazon SageMaker notebooks to query the database for common use cases. A complete customer identity graph solution usually consists of an ingestion pipeline, data validation, cleansing, identity resolution algorithms, identity graph database, and audience segmentation. This post focuses on ingesting data into a Neptune database, data modeling to capture interconnected profiles, and query mechanisms to support cross-device graphs, audience segmentation and other use cases.

Use cases

The following are some common use cases for this solution:

- **Cross-device and interest graph** – Find a given user's interests by analyzing the customer journey and time spent across devices to personalize advertising
- **Convince undecided consumers** – Identify ecommerce site visitors based on prior website visits
- **Audience segmentation by brand** – Create specific audiences based on brand and category interest or affinity scores
- **Interest-based advertising** – Target ads based on prior interest in specific websites
- **Early adopter path to purchase insights** – Analyze the customer's journey on a website from initial site visit to product purchase confirmation
- **Identify look-alike customers** – Query for common audience characteristics for a given purchased product

Graph databases are ideal for building customer identity graphs to capture and link billions of interconnected relationships to support these use cases. Although traditional Relational Database Management Systems (RDBMS) are ideal for building enterprise applications that require transactional integrity, they aren't designed to capture highly connected datasets such as customer device graphs and support millisecond latency at scale. Similarly, SaaS solutions provide limited flexibility to capture and model multiple relationships. In contrast, graph databases are easy to model one-to-many and many-to-many relationships, flexible to redesign, and store relationships at a physical storage level to support low-latency queries.

The customer identity graph solution is built on Neptune—a fast, reliable, fully-managed graph database service that makes it easy to build and run applications that work with highly connected datasets. At the core of Neptune is a purpose-built, high-performance graph database engine optimized for storing billions of relationships and querying the graph with millisecond latencies.

Solution overview

Creating a customer identity graph in Neptune requires three primary components:

- **The customer identity graph data model** – You first have to collect and transform your data to a graph data model in a format that you can load into your graph database (Neptune). This post also discusses the data elements required for a knowledge graph and the potential sources for that data.

- **Jupyter notebook and Python library** – You need a way to easily query your graph, explore the data, and potentially visualize the results. For this you use Jupyter notebooks, which is a common framework that many data engineers and data scientists use. Amazon SageMaker provides you with a fully managed Jupyter notebook environment made available through its notebook instance feature.
- **Creating a Neptune cluster and Amazon SageMaker notebook instance:** You create a Neptune cluster, load your data, and connect your Amazon SageMaker notebook instance with a pre-built CloudFormation stack.

Continue reading the full post online at: *https://amzn.to/AWS-AM-neptune*

Real-world example

Nike: A Social Graph at Scale with Amazon Neptune

Getting a graph database to be performant and easy to use is very different than making a NoSQL database high performing. In this episode we talk about how Nike powers a number of applications via a social graph, built on Amazon Neptune, that effectively maps millions of relationships between its users. We take a closer look at the underlying property graph that represents highly connected data, which allows users to select their interests such as basketball or training. These interest selections then drive personalized recommendations and curated content for consumers, based on entries in their graph.

https://amzn.to/AWS-AM-Nike-TMA

By Rodrigo Asensio (co-written with Lucas Ceballos, CTO of Smadex)

Read online at: *https://amzn.to/AWS-AM-ads-redis*

Introduction

Showing ads may seem to be a simple task, but it's not. Showing the right ad to the right user is an incredibly complex challenge that involves multiple disciplines such as artificial intelligence, data science, and software engineering. Doing it one million times per second with a 100-ms constraint is even harder.

In the ad-tech business, speed and infrastructure costs are the keys to success. The less the final user waits for an ad, the higher the probability of that user clicking on the ad. Doing that while keeping infrastructure costs under control is crucial for business profitability.

About Smadex

Smadex is the leading mobile-first programmatic advertising platform specifically built to deliver best user acquisition performance and complete transparency.

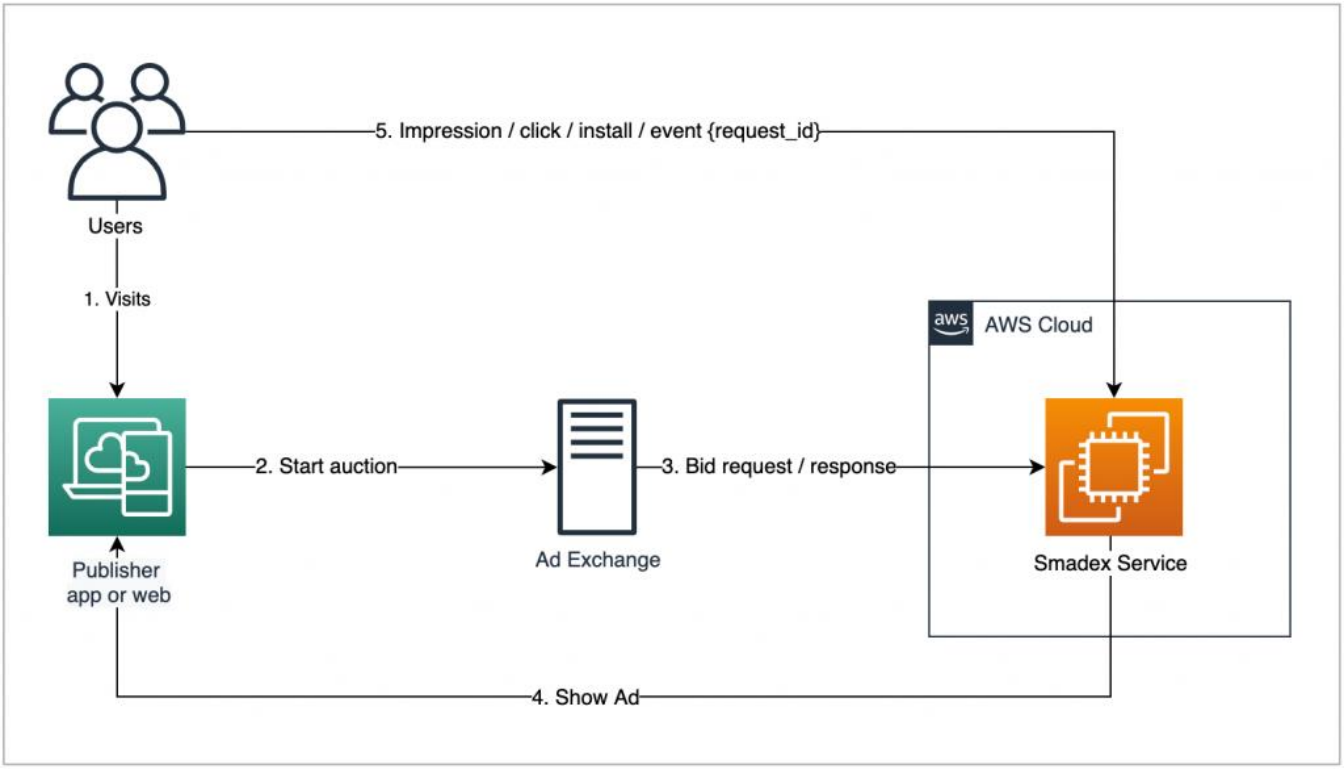
Its Demand Side Platform (DSP) technology provides advertisers with the tools they need to achieve their goals and ROI, with measurable results from web forms, post-app install events, store visits, and sales.

Smadex advertising architecture

What does showing ads look like under the hood? At Smadex, our technology works based on the OpenRTB (Real-Time Bidding) protocol.

RTB is a means by which advertising inventory is bought and sold on a per-impression basis, via programmatic instantaneous auction, which is similar to financial markets.

To show ads, we participate in auctions deciding in real time which ad to show and how much to bid trying to optimize the cost of every impression.



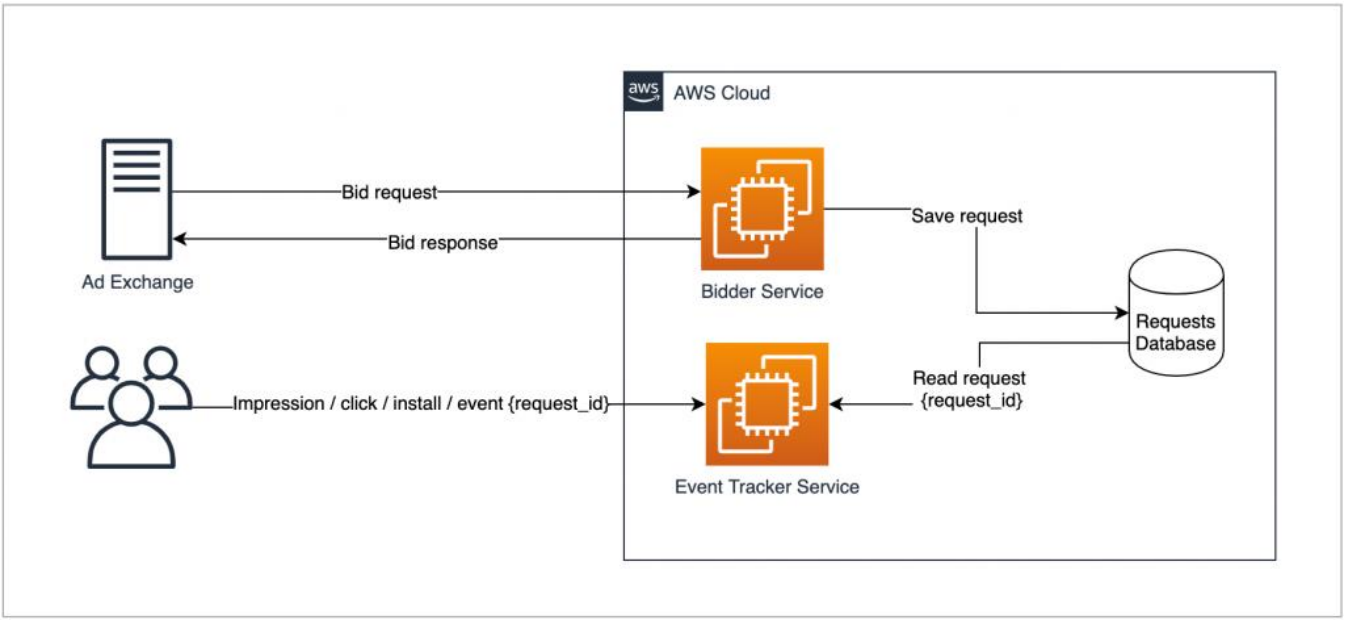
1. The final user browses the publisher's website or app.
2. Ad-exchange is called to start a new auction.
3. Smadex receives the bid request and has to decide which ad to show and how much to offer in just 100 ms (and this is happening one million times per second).
4. If Smadex won the auction, the ad must be sent and rendered on the publisher's website or app.
5. In the end, the user interacts with the ad sending new requests and events to Smadex platform.

Flow of data

As you can see in the previous diagram, showing ads is just one part of the challenge. After the ad is shown, the final user interacts with it in multiple ways, such as clicking it, installing an application, subscribing to a service, etc. This happens during a determined period that we call the "attribution window." All of those interactions must be tracked and linked to the original bid transaction (using the request_id parameter).

Doing this is complicated: billions of bid transactions must be stored and available so that they can be quickly accessed every time the user interacts with the ad. The longer we store

the transactions, the longer we can “wait” for an interaction to take place, and the better for our business and our clients, too.



Challenge #1: Cost

The challenge is: What kind of database can store billions of records per day, with at least a 30-day retention capacity (attribution window), be accessed by key-value, and all by spending as little as possible?

The answer is…none! Based on our research, all the available options that met the technical requirements were way out of our budget.

So…how to solve it? Here is when creativity and the combination of different AWS services comes into place.

We started to analyze the time dispersion of the events trying to find some clues. The interesting thing we spotted was that 90% of what we call “post-bid events” (impression, click, install, etc.) happened within one hour after the auction took place.

That means that we can process 90% of post-bid events by storing just one hour of bids.

Under our current workload, in one hour we participate in approximately 3.7 billion auctions generating 100 million bid records of an average 600 bytes each. This adds up to 55 gigabytes per hour, an easier amount of data to process.

Instead of thinking about one single database to store all the bid requests, we decided to split bids into two different categories:

- **Hot Bid**: A request that took place within the last hour (small amount and frequently accessed)
- **Cold Bid**: A request that took place more than our hour ago (huge amount and infrequently accessed)

Amazon ElastiCache for Redis is the best option to store 55 GB of data in memory, which gives us the ability to query in a key-value way with the lowest possible latency.

Continue reading the full post online at: *https://amzn.to/AWS-AM-ads-redis*

Real-world example

NextRoll: Driving OpEx Efficiency for Ad Bidding Engines

Valentino Volonghi from NextRoll explains how they handle over 100 Billion requests per day with EC2 Spot, EBS, ECS, and S3. Learn how to persist business critical data form terminating Spot instances building an ECS-based "Spot Savior" to collect tagged EBS drives and forward important data to a durable datastore like S3.

https://amzn.to/AWS-AM-nextroll

Solution:

Real-Time Web Analytics with Kinesis Data Analytics

View the full solution online at: *https://amzn.to/AWS-AM-kinesis-solution*

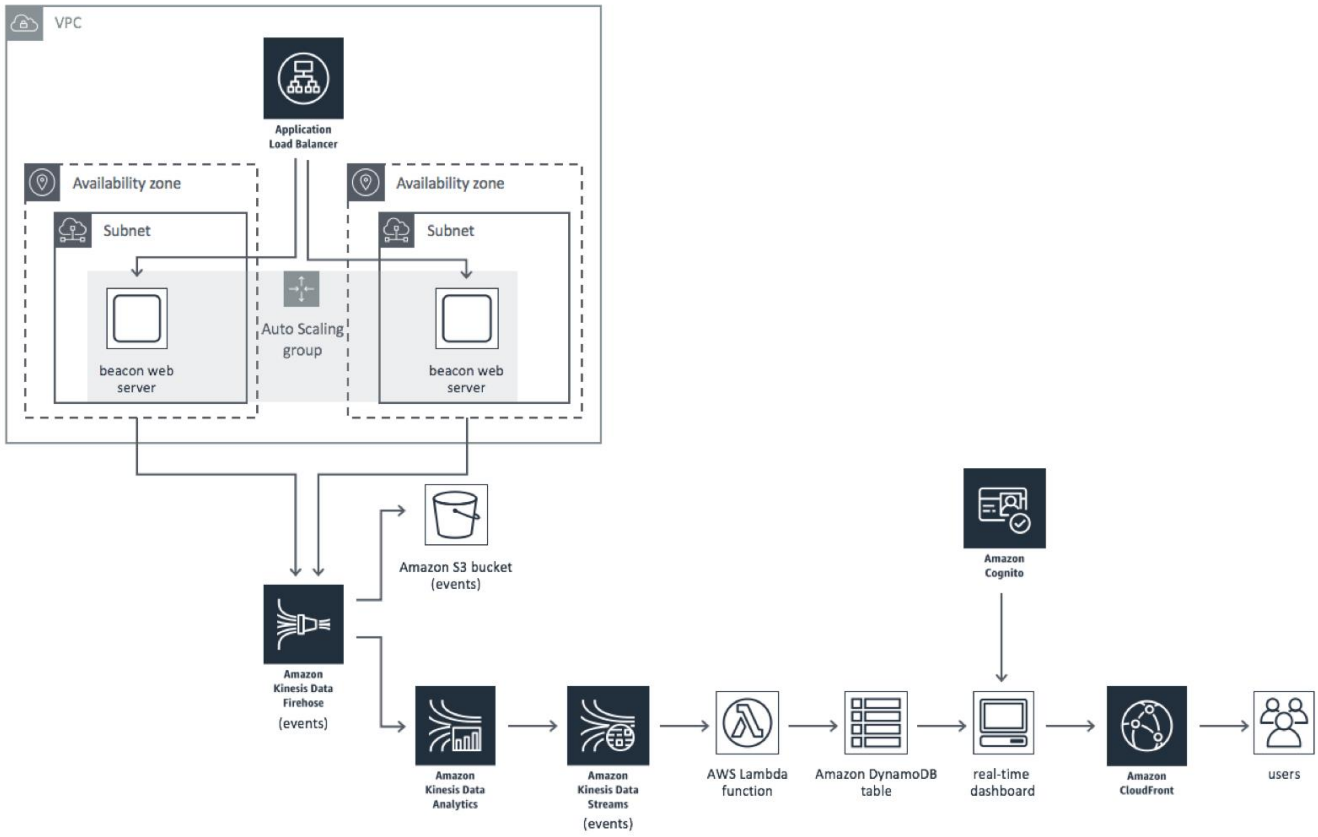
What does this AWS Solutions Implementation do?

The Real-Time Web Analytics with Kinesis Data Analytics solution automatically provisions the services necessary to track and visualize website clickstream data in real-time. This solution is designed to provide a framework for analyzing and visualizing metrics, allowing you to focus on adding new metrics rather than managing the underlying infrastructure.

Version 1.1.2 of the solution uses the most up-to-date Node.js runtime. Version 1.0.2 uses the Node.js 8.10 runtime, which reaches end-of-life on December 31, 2019. To upgrade to version 1.1.2, you must deploy the solution as a new stack. For more information, see the deployment guide (https://amzn.to/AWS-AM-solution-kinesis-deploy).

AWS Solutions Implementation overview

AWS offers a solution that uses beacon web servers to log requests from a user's web browser, Amazon Kinesis Data Firehose to capture website clickstream data, Amazon Kinesis Data Analytics to compute metrics in real-time, and Amazon Simple Storage Service (Amazon S3) and Amazon DynamoDB to durably store metric data. The solution also features a dashboard that visualizes your account activity in real-time. The diagram below presents the architecture you can deploy in minutes using the solution's implementation guide and accompanying AWS CloudFormation template.



Real-Time Web Analytics with Kinesis Data Analytics architecture

Web beacon servers log requests from a user's web browser and send the data to a Kinesis Data Firehose delivery stream. The delivery stream archives the events in an Amazon S3 bucket and sends the data to a Kinesis Data Analytics application for processing.

Once the data is processed, it is sent to Kinesis Data Streams. An AWS Lambda function reads data from the stream and sends the data in real-time to an Amazon DynamoDB table to be stored.

The solution also creates an Amazon Cognito user pool, an Amazon S3 bucket, an Amazon CloudFront distribution, and real-time dashboard to securely read and display the account activity stored in the DynamoDB table.

View the full solution online at: *https://amzn.to/AWS-AM-kinesis-solution*

Real-world example

Depop: Real-Time Data Ingestion and Analysis

Learn how Depop (https://www.depop.com/) built a data ingestion pipeline and analysis system to handle events from its ecommerce marketplace site. We discuss how the Depop team iterated on the initial architecture to take advantage of a flexible data lake pattern using Lambda, S3, Glue, and Athena. To allow Glue to create a different schema for each data type, the company created a separate Kinesis stream for each record or event type.

https://amzn.to/AWS-AM-depop

By Clark Fredricksen

Read online at: *https://amzn.to/AWS-AM-AM-reinvent-announcements*

With more than 77 new service launches and other announcements, AWS re:Invent left a lot to digest—even for teams inside AWS! We've developed this guide to ensure customers in the advertising and marketing industry know the most relevant news from this year's event.

Of the many announcements, three stood out as game-changers for advertising and marketing industry customers:

- **The Trade Desk** announced four new cloud-based real-time bidding sites running on AWS in Tokyo, Singapore, Beijing and Frankfurt, with another site coming online soon in Hong Kong. Zak Stengel, SVP of Engineering at The Trade Desk, spoke in the Advertising and Marketing Industry Leadership session at re:Invent. Stengel noted the launch of AWS Global Accelerator at AWS re:Invent 2018 was a game-changer for the company's bidding strategy, saying,*"The sheer size of our bidding workload and its latency requirements, along with how dynamic it can be, posed some really significant challenges to load balancing solutions. We began testing Global Accelerator shortly after its public release [at re:Invent 2018] … and we found that it worked for our workload. This was the final development that led us to shift toward a cloud-bidding strategy."*
- **Amazon Sagemaker**: AWS announced the availability of Deep Graph Library, an open source library built for easy implementation of graph neural networks, on Amazon SageMaker. The library will help customers in advertising and marketing improve machine learning on identity graph-friendly workloads such as Data Management Platforms, Customer Data Platforms, and other graph workloads for cross-device and customer event mapping.
- **AWS** made several **Amazon Redshift** announcements that will improve advertising analytics and big data workloads for industry customers, including Amazon Redshift RA3 nodes—featuring high bandwidth networking and performance indistinguishable from bare metal— as well as Amazon Redshift Federated Query (Preview) to query and analyze data across operational databases, data warehouses, and data lakes.

Advertising and marketing industry customer announcements

- **Nielsen Media** talked about their migration of National TV Audience Measurement to AWS, which included development of a 30PB AWS data lake that helped Nielsen expand from measuring 40,000 households to over 30 million.
- **Advertising technology firm Smaato** shared through how they use Apache Spark and Amazon SageMaker to reduce costs on programmatic advertising workloads with machine learning.
- **Annalect** dove deep into their use of containers and AWS analytics tools to reduce costs from $70 per usable TB to under $5 per usable TB while increasing their total queryable data from 100 TB to over 2 PB during the same period.
- Calvin French-Owen, CTO of Segment, shared keys for personalization for marketing technology firms using big data, also as part of the Advertising and Marketing Industry Leadership Session at re:Invent.
- **Zeta Global** announced using Amazon Neptune for cross-device identity resolution in their Customer Intelligence platform, which handles 450 million requests per day and resolves data for 1 billion customer profiles.

Below are some relevant services and features launched at this year's re:Invent grouped by common workloads and use-cases from industry customers.

Cost and performance optimization

Target use-cases: Optimize compute and networking costs at low-latency for scaled data collection and event streaming, machine learning inferencing, and for programmatic advertising workloads such as bidding, auctions, and ad serving.

- AWS Compute Optimizer recommends optimal AWS Compute resources for your workloads to reduce costs and improve performance by using machine learning to analyze historical usage metrics.
- EC2 inf1 instances for machine learning inferencing deliver up to 3x higher throughput and up to 40% lower cost per inference than Amazon EC2 G4 instances, which were already the lowest cost instance for machine learning inference available in the cloud.
- **Containers**:
    - EKS Preview: ARM-Processor EC2 A1 instances are now available in more regions with latest kubernetes versions.

    - Amazon Elastic Container Services (ECS) Cluster Auto Scaling is now available. With ECS Cluster Auto Scaling, your ECS clusters running on EC2 can automatically scale as needed to meet the resource demands of all tasks and services in your cluster, including scaling to and from zero.

- AWS Wavelength: Build applications that deliver single-digit millisecond latencies to mobile devices and end-users. Deploy applications to Wavelength Zones, AWS infrastructure deployments that embed AWS compute and storage services within the telecommunications providers' datacenters at the edge of the 5G networks, and seamlessly access the breadth of AWS services in the region. In advertising and marketing, this means workloads like ad tracking, event collection, identity matching, and ad serving are possible.
- AWS Transit Gateway Network Manager allows you to centrally manage and monitor your global network across AWS and on premises, with network manager. Transit Gateway network manager reduces the operational complexity of managing networks across AWS Regions and remote locations.
- AWS Transit Gateway now supports the ability to establish peering connections between Transit Gateways in different AWS Regions. Transit Gateway is a service that enables customers to connect thousands of Amazon Virtual Private Clouds (Amazon VPCs) and their on-premises networks using a single gateway.

***To continue reading about machine learning, advertising & customer-360 analytics, and data integrations and orchestration, visit the blog online at**: https://amzn.to/AWS-AM-AM-reinvent-announcements.*

Real-world example

HubSpot: Task Queueing System Built on Top of Amazon SQS

HubSpot discusses how it built a task queueing system built on top of Amazon SQS, for delivering retries, high availability, dead-letter management, message tracing, and more. HubSpot leverages this task queueing to support 450+ micro services, delivering features ranging from email delivery, to automated marketing workflows, or even automatic impaired host replacements.

https://amzn.to/AWS-AM-HubSpot