

# Contents

<b>Amazon Ads System Design</b>	<b>1</b>
Overview . . . . .	1
Upcoming Change: Third-Party Cookie Deprecation . . . . .	1
Workarounds & Adaptations . . . . .	1
Contextual Ads vs Behavioral Ads . . . . .	1
Key Components and Flow . . . . .	2
Data & Supporting Systems . . . . .	3
Auction & Delivery Flow . . . . .	3
Metrics & Goals . . . . .	3
Architecture Diagram . . . . .	4

## Amazon Ads System Design

### Overview

This document outlines the architecture and data flow of Amazon Ads, supporting real-time bidding and ad delivery across Amazon properties (Fire TV, IMDb, Alexa, Amazon.com). The system enables targeting, auctioning, and creative rendering in under 100ms using a modular architecture with strong data compliance and ML-based personalization.

### Upcoming Change: Third-Party Cookie Deprecation

As third-party cookies are deprecated (already blocked in Safari and Firefox, and rolling out in Chrome through 2025), traditional methods of cross-site user tracking and identity resolution via cookies will no longer be reliable. This significantly impacts identity mapping, retargeting, frequency capping, and conversion attribution for ads served across different Amazon properties and third-party contexts.

### Workarounds & Adaptations

- **First-party identity solutions:** Rely more heavily on Amazon’s authenticated user graph and deterministic identifiers like hashed emails.
- **Cohort-based targeting:** Use context-aware groupings (similar to Google’s Topics API) rather than individual tracking.
- **Server-side identity resolution:** Shift tracking and ID resolution to server-side pipelines (e.g., via CDPs or internal ID graphs).
- **Contextual targeting:** Match ads to content metadata rather than user profiles when identity signals are weak or unavailable.
- **Telemetry enhancement:** Expand usage of server-logged events (clicks, conversions) to model attribution without relying on third-party cookies.

### Contextual Ads vs Behavioral Ads

As cookie deprecation limits behavioral targeting, it’s important to distinguish between contextual and behavioral advertising approaches:

Feature	Behavioral Ads	Contextual Ads
Targeting Basis	Past behavior & user profiles	Current page or app content
Requires Tracking	Yes (cookies, device IDs, etc.)	No
Personalization Level	High	Moderate
Privacy Concerns	High (subject to consent laws)	Low (no personal data used)
Resilience to Changes	Vulnerable to signal loss	Resilient and future-proof

- **Behavioral Ads** rely on identity graphs and user tracking to personalize ads across websites and apps.
- **Contextual Ads** match ads to the surrounding content without needing user history or identifiers, making them privacy-friendly and suitable in a post-cookie world.

This distinction should guide ad targeting strategies and influence how ML models are trained and evaluated in the absence of persistent user-level signals.

## Key Components and Flow

### 1. User Device / Publisher

- Initiates the ad request from Fire TV, Alexa, IMDb, or other Amazon-owned properties.
- **Runtime Path:** Yes
- **Core Technologies:** Native apps, browsers, Amazon devices

### 2. API Gateway / Front Door

- Entry point for all ad requests. Performs validation, authentication, and routing (AWS API Gateway or Lambda@Edge).
- **Runtime Path:** Yes
- **Core Technologies:** AWS API Gateway, Lambda@Edge, REST/gRPC

### 3. Ad Request Orchestrator

- Enriches ad requests with user, geo, device, and consent data. Applies targeting rules and compliance filters.
- **Runtime Path:** Yes
- **Core Technologies:** Java, Go, AWS EKS/ECS, Redis, DynamoDB, Kafka/S3, SageMaker

### 4. Bid Request Engine

- Converts enriched context into OpenRTB format and dispatches requests to multiple buyers (DSPs) in parallel.
- **Runtime Path:** Yes
- **Core Technologies:** Java, Rust, OpenRTB (HTTP/gRPC), Kafka/Kinesis

### 5. Demand-Side Platforms (DSPs)

- External/internal systems used by advertisers to receive bid requests and respond with price and creative.
- **Runtime Path:** Yes
- **Core Technologies:** Amazon DSP, Google DV360, The Trade Desk, MediaMath

### 6. Ad Auction Service

- Conducts second-price auctions or priority-based selection across eligible bids.
- **Runtime Path:** Yes
- **Core Technologies:** Java, Scala, Netty/Akka, Redis, DynamoDB, Kafka, CloudWatch, Prometheus

### 7. Bid Management Layer

- Handles campaign pacing, spend throttling, and bid control logic before auction execution.
  - **Runtime Path:** Yes
  - **Core Technologies:** Java, Go, Redis, Kafka, ML models (optional)
8. **Creative Renderer**
- Delivers the winning creative (image, video, or HTML) to the user device by transforming ad markup into renderable assets and interfacing with CDN infrastructure.
  - **Runtime Path:** Yes
  - **Core Technologies:** JavaScript, Node.js, Java, CloudFront/Akamai, HTML sanitization, tracking beacons
9. **Advertiser Dashboard & Developer Experience**
- UI and APIs for advertisers to manage campaigns, budgets, creatives, and reporting.
  - **Runtime Path:** No
  - **Core Technologies:** React, REST/GraphQL APIs, S3, Aurora, DynamoDB

## Data & Supporting Systems

- **Campaign Management System:** Stores advertiser campaign configurations, including budgets, targeting rules, creative links, and scheduling constraints.
  - **Runtime Path:** No (accessed asynchronously)
  - **Core Technologies:** Aurora, DynamoDB, REST/GraphQL, Kafka/SQS
- **Telemetry / Spend Tracking Systems:** Captures ad delivery logs, auction outcomes, impression tracking, and budget spend events.
  - **Runtime Path:** No (async data collection)
  - **Core Technologies:** Kafka, Kinesis, S3, Redshift, ClickHouse, Flink, Spark, Athena
- **Creative Asset Store / CDN:** Stores final creative assets (images, videos, JS) and delivers them to user devices via CDN.
  - **Runtime Path:** Partial (user device fetches assets after receiving markup)
  - **Core Technologies:** S3, CloudFront, Fastly, Akamai

## Auction & Delivery Flow

- Ad requests are validated and enriched.
- Bid requests are sent to DSPs in OpenRTB format.
- DSPs respond with bids and creatives.
- Ad Auction Service selects the winning bid (second-price or priority-based).
- Creative Renderer validates and delivers the creative to the user device.
- Telemetry systems log delivery and performance data.

## Metrics & Goals

- QPS by request type and service
- Auction win rates and bid response times (p50/p99)
- Revenue per mille (RPM)

- Click-through rate (CTR)
- Creative failure rate
- Budget utilization per campaign
- Maintain latency budget under 100ms
- Support multiple ad types and channels
- Ensure compliance with GDPR/CCPA
- Provide ML-ready telemetry and reporting

## Architecture Diagram

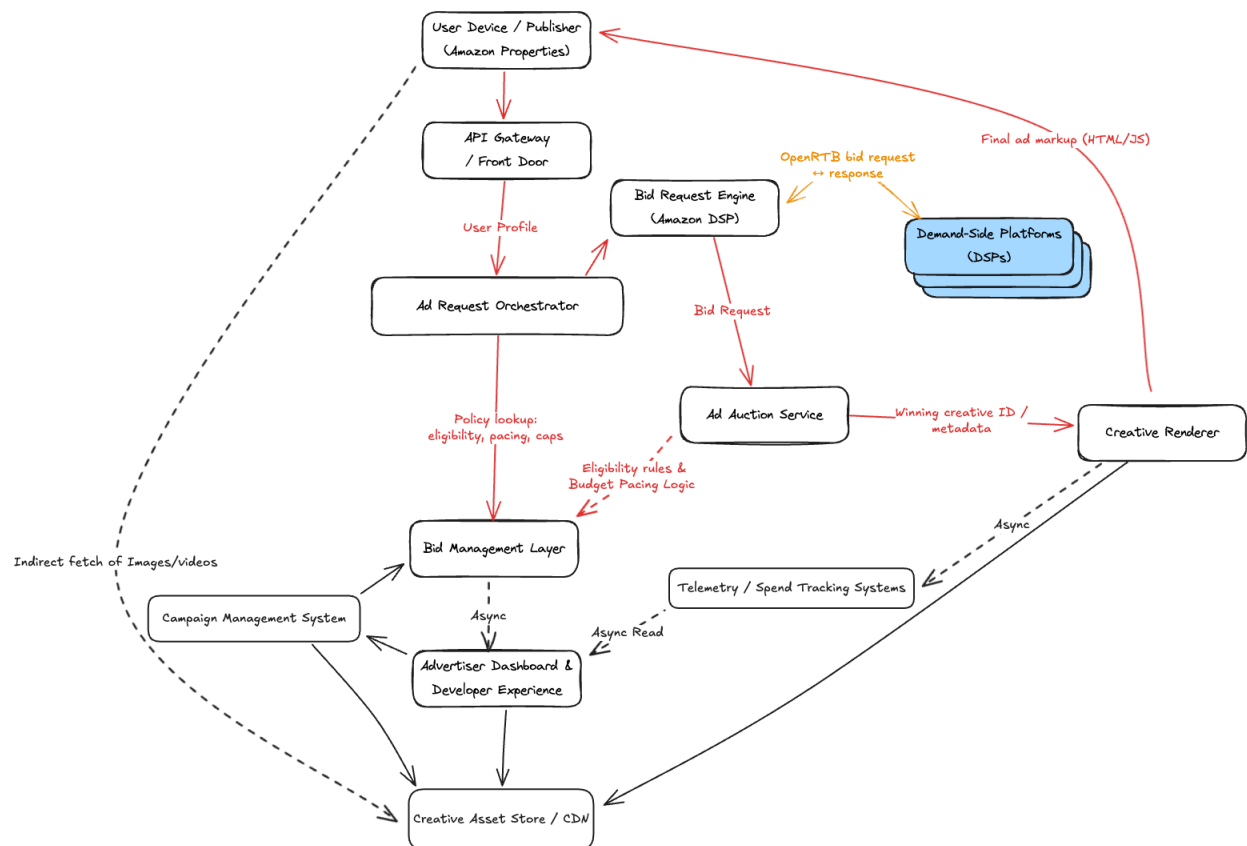


Figure 1: Amazon Ads System Diagram

You can edit this diagram by uploading the PNG to [Excalidraw](#).