Athena FAQs

General

Q: What is Amazon Athena?

Amazon Athena is an interactive query service that makes it easy to analyze data in Amazon S3 using standard SQL. Athena is serverless, so there is no infrastructure to setup or manage, and you can start analyzing data immediately. You don’t even need to load your data into Athena, it works directly with data stored in S3. To get started, just log into the Athena Management Console, define your schema, and start querying. Amazon Athena uses Presto with full standard SQL support and works with a variety of standard data formats, including CSV, JSON, ORC, Apache Parquet and Avro. While Amazon Athena is ideal for quick, ad-hoc querying and integrates with Amazon QuickSight for easy visualization, it can also handle complex analysis, including large joins, window functions, and arrays.

Q: What can I do with Amazon Athena?

Amazon Athena helps you analyze data stored in Amazon S3. You can use Athena to run ad-hoc queries using ANSI SQL, without the need to aggregate or load the data into Athena. Amazon Athena can process unstructured, semi-structured, and structured data sets. Examples include CSV, JSON, Avro or columnar data formats such as Apache Parquet and Apache ORC. Amazon Athena integrates with Amazon QuickSight for easy visualization. You can also use Amazon Athena to generate reports or to explore data with business intelligence tools or SQL clients, connected via a [JDBC driver](http://docs.aws.amazon.com/athena/latest/ug/connect-with-jdbc.html).

Q: How do I get started with Amazon Athena?

To get started with Amazon Athena, simply log into the AWS Management Console for Athena and create your schema by writing DDL statements on the console or by using a create table wizard. You can then start querying data using a built-in query editor. Athena queries data directly from Amazon S3 so there’s no loading required.

Q: How do you access Amazon Athena?

Amazon Athena can be accessed via the AWS Management Console, an API, or a JDBC driver. You can programmatically run queries, add tables or partitions using the [JDBC driver](http://docs.aws.amazon.com/athena/latest/ug/connect-with-jdbc.html).

Q: What are the service limits associated with Amazon Athena?

Please [click here](http://docs.aws.amazon.com/athena/latest/ug/known-limitations.html) to learn more about service limits

Q: What is the underlying technology behind Amazon Athena?

Amazon Athena uses [Presto](https://aws.amazon.com/big-data/what-is-presto/) with full standard SQL support and works with a variety of standard data formats, including CSV, JSON, ORC, Avro, and Parquet. Athena can handle complex analysis, including large joins, window functions, and arrays. Because Amazon Athena uses Amazon S3 as the underlying data store, it is highly available and durable with data redundantly stored across multiple facilities and multiple devices in each facility. Learn more about Presto [here](https://aws.amazon.com/big-data/what-is-presto/).

Q: How does Amazon Athena store table definitions and schema?

Amazon Athena uses a managed Data Catalog to store information and schemas about the databases and tables that you create for your data stored in Amazon S3. In regions where AWS Glue is available, you can upgrade to using the AWS Glue Data Catalog with Amazon Athena. In regions where AWS Glue is not available, Athena uses an internal Catalog.

You can modify the catalog using DDL statements or via the AWS Management Console. Any schemas you define are automatically saved unless you explicitly delete them. Athena uses schema-on-read technology, which means that your table definitions applied to your data in S3 when queries are being executed. There’s no data loading or transformation required. You can delete table definitions and schema without impacting the underlying data stored on Amazon S3.

Q: Why should I upgrade to AWS Glue Data Catalog?

[AWS Glue](https://aws.amazon.com/glue/) is a fully managed ETL service. Glue has three main components: 1) a crawler that automatically scans your data sources, identifies data formats and infers schemas, 2) a fully managed ETL service that allows you to transform and move data to various destinations, and 3) a Data Catalog that stores metadata information about databases & tables either stored in S3 or a JDBC-compliant data store. To use the benefits of Glue, you must upgrade from using Athena’s internal Data Catalog to the Glue Data Catalog.

The benefits of upgrading to the Glue Data Catalog are:

1. Unified Metadata Repository: AWS Glue is integrated across a wide range of AWS services. AWS Glue supports data stored in Amazon Aurora, Amazon RDS MySQL, Amazon RDS PostreSQL, Amazon Redshift, and Amazon S3, as well as MySQL and PostgreSQL databases in your Virtual Private Cloud (Amazon VPC) running on Amazon EC2. AWS Glue provides out-of-the-box integration with Amazon Athena, Amazon EMR, Amazon Redshift Spectrum, and any Apache Hive Metastore-compatible application.
2. Automatic schema and partition recognition: AWS Glue automatically crawls your data sources, identifies data formats, and suggests schemas and transformations. Crawlers can help automate table creation and automatic loading of partitions.
3. Easy to build pipelines: AWS Glue’s ETL engine generates Python code that is customizable, reusable, and portable. You can edit the code using your favorite IDE or notebook and share it with others using GitHub. Once your ETL job is ready, you can schedule it to run on AWS Glue's fully managed, scale-out Spark infrastructure. AWS Glue is serverless, so it handles provisioning, configuration, and scaling of the resources required to run your ETL jobs, allowing you to tightly integrate ETL in your workflow.

Click [here](https://aws.amazon.com/glue/) to learn more about the Glue Data Catalog.

Q. Is there a step-by-step to upgrade to the AWS Data Catalog?

Yes. Step-by-Step guide can be found [here](http://docs.aws.amazon.com/athena/latest/ug/glue-athena.html).

Q: What regions is Amazon Athena available in?

Please refer to [Regional Products and Services](https://aws.amazon.com/about-aws/global-infrastructure/regional-product-services/) for details of Amazon Athena service availability by region.

When to Use Athena vs Other Big Data Services

Q: What is the difference between Amazon Athena, Amazon EMR, and Amazon Redshift?

Query services like Amazon Athena, data warehouses like Amazon Redshift, and sophisticated data processing frameworks like Amazon EMR, all address different needs and use cases. You just need to choose the right tool for the job. Amazon Redshift provides the fastest query performance for enterprise reporting and business intelligence workloads, particularly those involving extremely complex SQL with multiple joins and sub-queries. Amazon EMR makes it simple and cost effective to run highly distributed processing frameworks such as Hadoop, Spark, and Presto when compared to on-premises deployments. Amazon EMR is flexible - you can run custom applications and code, and define specific compute, memory, storage, and application parameters to optimize your analytic requirements. Amazon Athena provides the easiest way to run ad-hoc queries for data in S3 without the need to setup or manage any servers.

Q: When should you use a full featured enterprise data warehouse, like Amazon Redshift vs. a query service like Amazon Athena?

A data warehouse like Amazon Redshift is your best choice when you need to pull together data from many different sources – like inventory systems, financial systems, and retail sales systems – into a common format, and store it for long periods of time, to build sophisticated business reports from historical data; then a data warehouse like Amazon Redshift is the best choice.

Data warehouses collect data from across the company and act as the “single source of truth” for report generation and analysis. Data warehouses pull data from many sources, format and organize it, store it, and support complex, high speed queries that produce business reports. The query engine in Amazon Redshift has been optimized to perform especially well on this use case - where you need to run complex queries that join large numbers of very large database tables. TPC-DS is a standard benchmark designed to replicate this use case, and Redshift runs these queries up to 20x faster than query services that are optimized for unstructured data. When you need to run queries against highly structured data with lots of joins across lots of very large tables, you should choose Amazon Redshift.

By comparison, query services like Amazon Athena make it easy to run interactive queries against data directly in Amazon S3 without worrying about formatting data or managing infrastructure. For example, Athena is great if you just need to run a quick query on some web logs to troubleshoot a performance issue on your site. With query services, you can get started fast. You just define a table for your data and start querying using standard SQL.

You can also use both services together. If you stage your data on Amazon S3 before loading it into Amazon Redshift, that data can also be registered with and queried by Amazon Athena.

Q: When should I use Amazon EMR vs. Amazon Athena?

Amazon EMR goes far beyond just running SQL queries. With EMR you can run a wide variety of scale-out data processing tasks for applications such as machine learning, graph analytics, data transformation, streaming data, and virtually anything you can code. You should use Amazon EMR if you use custom code to process and analyze extremely large datasets with the latest big data processing frameworks such as Spark, Hadoop, Presto, or Hbase. Amazon EMR gives you full control over the configuration of your clusters and the software installed on them.

You should use Amazon Athena if you want to run interactive ad hoc SQL queries against data on Amazon S3, without having to manage any infrastructure or clusters.

Q: Can I use Amazon Athena to query data that I process using Amazon EMR?

Yes, Amazon Athena supports many of the same data formats as Amazon EMR. Athena’s data catalog is Hive metastore compatible. If you're using EMR and already have a Hive metastore, you simply execute your DDL statements on Amazon Athena, and then you can start querying your data right away without impacting your Amazon EMR jobs.

Creating Tables, Data Formats and Partitions

Q: How do I create tables and schemas for my data on Amazon S3?

Amazon Athena uses Apache Hive DDL to define tables. You can run DDL statements using the Athena console, via a JDBC driver, via the API, or using the Athena create table wizard. If you use the AWS Glue Data Catalog with Athena, you can also use Glue crawlers to automatically infer schemas and partitions. An AWS Glue crawler connects to a data store, progresses through a prioritized list of classifiers to extract the schema of your data and other statistics, and then populates the Glue Data Catalog with this metadata. Crawlers can run periodically to detect the availability of new data as well as changes to existing data, including table definition changes. Crawlers automatically add new tables, new partitions to existing table, and new versions of table definitions. You can customize Glue crawlers to classify your own file types.

When you create a new table schema in Amazon Athena the schema is stored in the Data Catalog and used when executing queries, but it does not modify your data in S3. Athena uses an approach known as schema-on-read, which allows you to project your schema onto your data at the time you execute a query. This eliminates the need for any data loading or transformation. Learn more about [creating tables](http://docs.aws.amazon.com/athena/latest/ug/creating-tables.html).

Q: What data formats does Amazon Athena support?

Amazon Athena supports a wide variety of data formats like CSV, TSV, JSON, or Textfiles and also supports open source columnar formats such as Apache ORC and Apache Parquet. Athena also supports compressed data in Snappy, Zlib, LZO, and GZIP formats. By compressing, partitioning, and using columnar formats you can improve performance and reduce your costs.

Q: What kind of data types does Amazon Athena support?

Amazon Athena supports both simple data types such as INTEGER, DOUBLE, VARCHAR and complex data types such as MAPS, ARRAY and STRUCT.

Q: Can I run any Hive Query on Athena?

Amazon Athena uses Hive only for DDL (Data Definition Language) and for creation/modification and deletion of tables and/or partitions. Please [click here](http://docs.aws.amazon.com/athena/latest/ug/language-reference.html) for a complete list of statements supported. Athena uses Presto when you run SQL queries on Amazon S3. You can run ANSI-Compliant SQL SELECT statements to query your data in Amazon S3.

Q: What is a SerDe?

SerDe stands for Serializer/Deserializer, which are libraries that tell Hive how to interpret data formats. Hive DLL statements require you to specify a SerDe, so that the system knows how to interpret the data that you’re pointing to. Amazon Athena uses SerDes to interpret the data read from Amazon S3. The concept of SerDes in Athena is the same as the concept used in Hive. Amazon Athena supports the following SerDes:

1. Apache Web Logs: "org.apache.hadoop.hive.serde2.RegexSerDe"
2. CSV: "org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe"
3. TSV: "org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe"
4. Custom Delimiters: "org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe"
5. Parquet: "org.apache.hadoop.hive.ql.io.parquet.serde.ParquetHiveSerDe"
6. Orc: "org.apache.hadoop.hive.ql.io.orc.OrcSerde"
7. JSON: “org.apache.hive.hcatalog.data.JsonSerDe” OR org.openx.data.jsonserde.JsonSerDe

Q: Can I add my own SerDe (Serializer/Deserializer) to Amazon Athena?

Currently, you cannot add your own SerDe to Amazon Athena. We appreciate your feedback, so if there are any SerDes you would like to see added, please contact the Athena team at [Athena-feedback@amazon.com](mailto:Athena-feedback@amazon.com)

Q: I created Parquet/ORC files using Spark/Hive. Will I be able to query them via Athena?

Yes, Parquet and ORC files created via Spark can be read in Athena.

Q: I have data coming from Kinesis Firehose. How can I query it using Athena?

If your Kinesis Firehose data is stored in Amazon S3, you can query it using Amazon Athena. Simply create a schema for your data in Athena and start querying. We recommend that you organize the data into partitions to optimize performance. You can add partitions created by Kinesis Firehose using ALTER TABLE DDL statements. Learn more about [partitions](http://docs.aws.amazon.com/athena/latest/ug/partitions.html).

Q: Does Amazon Athena support data partitioning?

Yes. Amazon Athena allows you to partition your data on any column. Partitions allow you to limit the amount of data each query scans, leading to cost savings and faster performance. You can specify your partitioning scheme using the PARTITIONED BY clause in the CREATE TABLE statement. Learn more about [partitioning data](http://docs.aws.amazon.com/athena/latest/ug/partitions.html).

Q: How do I add new data to an existing table in Amazon Athena?

If your data is partitioned, you will need to run a metadata query (ALTER TABLE ADD PARTITION) to add the partition to Athena once new data becomes available on Amazon S3. If your data is not partitioned, just adding the new data (or files) to the existing prefix automatically adds the data to Athena. Learn more about [partitioning data](http://docs.aws.amazon.com/athena/latest/ug/partitions.html).

Q: I already have large quantities of log data in Amazon S3. Can I use Amazon Athena to query it?

Yes, Amazon Athena makes it easy to run standard SQL queries on your existing log data. Athena queries data directly from Amazon S3 so there’s no data movement or loading required. Simply define your schema using DDL statements and start querying your data right away.

Querying and Data Formats

Q: What kinds of queries does Amazon Athena support?

Amazon Athena supports ANSI SQL queries. Amazon Athena uses Presto, an open source, in-memory, distributed SQL engine, and can handle complex analysis, including large joins, window functions, and arrays.

Q: Can I use Amazon QuickSight with Amazon Athena?

Yes. Amazon Athena integrates with Amazon QuickSight, allowing you to easily visualize your data stored in Amazon S3.

Q: Does Athena support other BI Tools and SQL Clients?

Yes. Amazon Athena comes with a JDBC driver you can use with other business intelligence tools and SQL clients. Learn more about using a [JDBC driver](http://docs.aws.amazon.com/athena/latest/ug/connect-with-jdbc.html) with Athena.

Q: Does Athena support User Defined Functions (UDFs)?

Currently, Athena does not support custom UDFs. If you need custom UDF support, please email us at [athena-feedback@amazon.com](mailto:athena-feedback@amazon.com)so we can learn more about your requirements.

Q: How do I access the functions supported by Amazon Athena?

Click [here](http://docs.aws.amazon.com/athena/latest/ug/language-reference.html) to learn more about functions supported by Amazon Athena.

Q: How do I improve the performance of my query?

You can improve the performance of your query by compressing, partitioning, or converting your data into columnar formats. Amazon Athena supports open source columnar data formats such as Apache Parquet and Apache ORC. Converting your data into a compressed, columnar format lowers your cost and improves query performance by enabling Athena to scan less data from S3 when executing your query.

Security & Availability

Q: How do I control access to my data?

Amazon Athena allows you to control access to your data by using AWS Identity and Access Management (IAM) policies, Access Control Lists (ACLs), and Amazon S3 bucket policies. With IAM policies, you can grant IAM users fine-grained control to your S3 buckets. By controlling access to data in S3, you can restrict users from querying it using Athena.

Can Athena query encrypted data in Amazon S3?

Yes, you can query data that’s encrypted using Server-Side Encryption with Amazon S3-Managed Encryption Keys, Server-Side Encryption with AWS Key Management Service (KMS) – Managed Keys, and Client-Side Encryption with keys managed by KMS. Amazon Athena also integrates with KMS and provides you an option to encrypt your result sets.

Q: Is Athena highly available?

Yes. Amazon Athena is highly available and executes queries using compute resources across multiple facilities, automatically routing queries appropriately if a particular facility is unreachable. Athena uses Amazon S3 as its underlying data store, making your data highly available and durable. Amazon S3 provides durable infrastructure to store important data and is designed for durability of 99.999999999% of objects. Your data is redundantly stored across multiple facilities and multiple devices in each facility.

Q: Can I provide cross-account access to someone else’s S3 bucket?

Yes, you can provide cross-account access to Amazon S3.

Pricing & Billing

Q: How is Amazon Athena priced?

Amazon Athena is priced per query and charges based on the amount of data scanned by the query. You can store data in a variety of formats on Amazon S3. If you compress your data, partition, or convert it to columnar storage formats, you pay less because you scan less data. Converting data to the columnar format allows Athena to read only the columns it needs to process the query. Please see the Athena [pricing page](https://aws.amazon.com/athena/pricing/) for more details

Q: Why do I get charged less when I use a columnar format?

Amazon Athena charges you for the amount of data scanned per query. Compressing your data allows Amazon Athena to scan less data. Converting your data to columnar formats allows Athena to selectively read only required columns to process the data. Partitioning your data also allows Athena to restrict the amount of data scanned. This leads to cost savings and improved performance. See [pricing](https://aws.amazon.com/athena/pricing/) example for details.

Q: How do I lower my costs?

You can save 30%-90% on your query costs and get better performance by compressing, partitioning, and converting your data into columnar formats. Each of these operations reduces the amount of data Amazon Athena needs to scan to execute a query. Amazon Athena supports Apache Parquet and ORC, two of the most popular open-source columnar formats. You can see the amount of data scanned per query, on the Athena console.

Q: Does Amazon Athena charge me for failed queries?

No, you are not charged for failed queries.

Q: Does Amazon Athena charge me for cancelled queries?

Yes, if you cancel a query manually, you are charged for the amount of data scanned up to the point at which you cancelled the query.

Q: Are there any additional charges associated with Amazon Athena?

Amazon Athena queries data directly from Amazon S3, so your source data is billed at S3 rates. When Amazon Athena runs a query, it stores the results in an S3 bucket of your choice and you are billed at standard S3 rates for these result sets. We recommend you monitor these buckets and use lifecycle policies to control how much data gets retained.

Q. Will I be charged for using AWS Glue Data Catalog?

Yes, you are charged separately for using the AWS Glue Data Catalog. Click [here](https://aws.amazon.com/glue/pricing/) to learn more about Glue Data Catalog pricing.