Kinesis Data Streams vs Apache Kafka

|  |  |  |
| --- | --- | --- |
| **Concepts** | **Apache Kafka** | **AWS Kinesis Data Streams** |
| **Data Storage** | Partitions | Shards |
| **Data Ordering** | In partition level | In shard level |
| **Data Retention** | No maximum (configurable) | 1 to 7 days (default is 24 hours) |
| **Data Size Per Blob** | Default 1MB (but can be configured) | Maximum 1 MB |
| **Partition/Shard Modification** | Increase only and does not repartition existing data | Re-shard by merging or splitting shards |
| **Partition/Shard Limitation** | No limit. Optimal partitions depend on the use case | 500 shards in US East (N. Virginia), US West (Oregon), and EU (Ireland) regions. 200 shards in all other regions. |
| **Data Replication/DR** | Cluster mirroring | Automatically across 3 Availability Zones |
| **Message Delivery Semantics** | Kafka guarantees at-least-once delivery by default. Kafka supports exactly-once delivery in Kafka Streams | Kinesis Data Streams has at least once semantics |
| **Security** | Either SSL or SASL and authentication of connections to Kafka Brokers from clients; authentication of connections from brokers to ZooKeeper; data encryption with SSL/TLS | Data can be secured at-rest by using server-side encryption and AWS KMS master keys on sensitive data within KDS. Access data privately via your Amazon Virtual Private Cloud (VPC) |
| **Monitoring** | Yammer Metrics for metrics reporting in the server | AWS CloudWatch and CloudTrail |
| **Dependency** | ZooKeeper | DynamoDB |
| **Cost** | Requires a lot of human support on installation, set up, configuration and clusters management. Setup in weeks | [Pay](https://aws.amazon.com/kinesis/data-streams/pricing/) and use. Setup in a couple Of hours |

Conclusion

Both Apache Kafka and AWS Kinesis Data Streams are good choices for real-time data streaming platforms. If you need to keep messages for more than 7 days with no limitation on message size per blob, Apache Kafka should be your choice. However, Apache Kafka requires extra effort to set up, manage, and support.  If your organization lacks Apache Kafka experts and/or human support, then choosing a fully-managed AWS Kinesis service will let you focus on the development. AWS Kinesis is catching up in terms of overall performance regarding throughput and events processing. When moving from Apache Kafka to AWS cloud service, you can set up Apache Kafka on AWS EC2. To avoid any challenge — such as setup and scale — and to manage clusters in production, AWS offers [Managed Streaming for Kafka (MSK)](https://aws.amazon.com/msk/) with settings and configuration based on Apache Kafka’s best deployment practices. MSK is public preview now and will GA in the first quarter of this year.