

KafkaJSNumberOfRetriesExceededMetadata

August 17, 2026

KafkaJSNumberOfRetriesExceededMetadata

Kind: Interface

Source: `packages/microservices/external/kafka.interface.ts`

Part of: [Microservices](#)

`KafkaJSNumberOfRetriesExceededMetadata` describes retry details when a KafkaJS operation exceeds its configured retry limit. It records the number of retry attempts made and the total retry time, allowing error handlers and observability tooling to report retry exhaustion accurately.

Properties

Property	Type
<code>retryCount</code>	<code>number</code>
<code>retryTime</code>	<code>number</code>

Diagram

```
graph LR
  A[KafkaJS operation] --> B{Retry attempt fails?}
  B -->|Yes| C[Increment retryCount]
  C --> D[Accumulate retryTime]
  D --> E{Retries exceeded?}
  E -->|No| A
  E -->|Yes| F[KafkaJSNumberOfRetriesExceededMetadata]
  F --> G[Error handling / logging]
```

Usage

```
import type { KafkaJSNumberOfRetriesExceededMetadata } from './kafka.interface';
```

```
function logRetryExhaustion(
  metadata: KafkaJSNumberOfRetriesExceededMetadata,
) {
  console.error(
    `Kafka retries exhausted after ${metadata.retryCount} attempts ` +
    `and ${metadata.retryTime}ms.`,
  );
}

const retryMetadata: KafkaJSNumberOfRetriesExceededMetadata = {
  retryCount: 5,
  retryTime: 12_000,
};

logRetryExhaustion(retryMetadata);
```

AI Coding Instructions

- Populate `retryCount` with the total number of retry attempts that occurred before failure.
- Store `retryTime` as a duration in milliseconds; do not use timestamps or seconds.
- Use this interface when handling KafkaJS retry-exhaustion errors to keep logging and error reporting type-safe.
- Avoid mutating retry metadata after it is attached to an error; treat it as a snapshot of the failed operation.