

InstrumentationEvent

August 18, 2026

InstrumentationEvent

Kind: Interface

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

Part of: [Microservices](#)

`InstrumentationEvent<T>` defines the standard envelope for instrumentation data emitted through the microservices Kafka integration. It provides a unique event ID, event type, timestamp, and a generic payload so producers and consumers can exchange typed telemetry consistently.

Properties

Property	Type
<code>id</code>	<code>string</code>
<code>type</code>	<code>string</code>
<code>timestamp</code>	<code>number</code>
<code>payload</code>	<code>T</code>

Diagram

```
graph LR
  Producer[Service / Instrumentation Producer] --> Event[InstrumentationEvent<T>]
  Event --> ID[id: unique event identifier]
  Event --> Type[type: event category]
  Event --> Timestamp[timestamp: Unix time]
  Event --> Payload[payload: typed event data]
  Event --> Kafka[Kafka Topic]
  Kafka --> Consumer[Instrumentation Consumer]
```

Usage

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

interface HttpRequestPayload {
  method: string;
  path: string;
  statusCode: number;
  durationMs: number;
}

const event: InstrumentationEvent<HttpRequestPayload> = {
  id: crypto.randomUUID(),
  type: 'http.request.completed',
  timestamp: Date.now(),
  payload: {
    method: 'GET',
    path: '/health',
    statusCode: 200,
    durationMs: 14,
  },
};

// Send the event through the configured Kafka producer.
await kafkaProducer.send({
  topic: 'instrumentation-events',
  messages: [{ key: event.type, value: JSON.stringify(event) }],
});
```

AI Coding Instructions

- Use a specific payload type for `T` ; avoid `any` so event producers and consumers remain type-safe.
- Generate a unique `id` for every event, typically with `crypto.randomUUID()` or the project's ID utility.
- Set `timestamp` with `Date.now()` and treat it as a Unix timestamp in milliseconds.
- Use stable, descriptive `type` values such as `service.operation.completed` so consumers can route events reliably.
- Serialize the complete event envelope when publishing to Kafka, and validate or parse the payload type when consuming it.