

NatsRequestJSONDeserializer

August 17, 2026

NatsRequestJSONDeserializer

Kind: Class

Source: `packages/microservices/deserializers/nats-request-json.deserializer.ts`

Part of: [Microservices](#)

`NatsRequestJSONDeserializer` converts raw NATS request messages into Nest microservice `IncomingRequest` or `IncomingEvent` objects. It parses the JSON payload, reads the correlation ID from NATS headers, and attaches a response function backed by the original NATS message.

Extends: `IncomingRequestDeserializer`

Methods

Method	Signature	Returns
<code>deserialize</code>	<code>deserialize(value: Uint8Array, options: Record<string, any>)</code>	<code>IncomingRequest</code>

Diagram

```
graph LR
  A[NATS message] --> B[NatsRequestJSONDeserializer.deserialize]
  B --> C[Parse JSON payload]
  B --> D[Read correlation ID header]
  B --> E[Bind message.respond]
  C --> F[IncomingRequest or IncomingEvent]
  D --> F
  E --> F
```

Usage

```
import { NatsRequestJSONDeserializer } from '@nestjs/microservices';

const deserializer = new NatsRequestJSONDeserializer();

const natsMessage = {
  data: Buffer.from(
    JSON.stringify({
      pattern: 'users.findOne',
      data: { id: '42' },
    })
  ),
  headers: {
    get: (key: string) =>
      key === 'Nats-Correlation-Id' ? 'request-123' : undefined,
  },
  respond: (payload: Uint8Array) => {
    // Send a response through NATS
  },
};

const request = deserializer.deserialize(natsMessage);

console.log(request.pattern); // "users.findOne"
console.log(request.data); // { id: "42" }
console.log(request.id); // "request-123"
```

AI Coding Instructions

- Preserve the expected NATS message shape: payload data, optional headers, and a `respond` function.
- Keep payloads JSON-serializable because `deserialize()` parses the message body as JSON.
- Do not remove the bound `response` function; request-response handlers depend on it to publish replies.
- Ensure clients include the NATS correlation ID header when request identity or response matching is required.
- Treat the deserialized result as either an `IncomingRequest` or `IncomingEvent`, depending on the payload shape and transport flow.

Relationships

- IMPORTS → `loadPackage`