

ClientContext

August 19, 2026

ClientContext

Kind: Interface**Source:** `src/adapters/aws-lambda/types.ts`**Part of:** [Adapter](#)

`ClientContext` defines the request-scoped context passed through the AWS Lambda adapter. It groups the platform client API, environment values, and an open-ended `Custom` field for application-specific context.

Properties

Property	Type
<code>client</code>	<code>ClientContextClient</code>
<code>Custom</code>	<code>any</code>
<code>env</code>	<code>ClientContextEnv</code>

Diagram

```
graph LR
  Context[Context] --> Client[Client]
  Context --> Env[Env]
  Context --> Custom[Custom]
```

Usage

```
import type { ClientContext } from "./types";

function handleRequest(context: ClientContext) {
  const region = context.env.region;

  context.client.log({
    message: `Handling request in ${region}`,
  });

  const tenantId = context.Custom?.tenantId;

  return {
    tenantId,
    region,
  };
}
```

AI Coding Instructions

- Pass `ClientContext` through Lambda request handling code instead of separately passing `client`, `env`, and custom values.
- Access AWS adapter APIs through `context.client` and configuration through `context.env`.
- Treat `Custom` as application-defined data; narrow or validate its shape before reading properties.
- Keep custom context values request-scoped to avoid sharing mutable state across Lambda invocations.

How it works

`ClientContext` is an exported TypeScript interface that describes the `clientContext` value on an AWS Lambda `LambdaContext`. That enclosing property is optional and may be `undefined`. [src/adapter/aws-lambda/types.ts:8-13](#) [src/adapter/aws-lambda/types.ts:35-47](#)

Its required members are:

- `client`, a `ClientContextClient` object containing string fields for `installationId`, `appTitle`, `appVersionName`, `appVersionCode`, and `appPackageName`. [src/adapter/aws-lambda/types.ts:9](#) [src/adapter/aws-lambda/types.ts:15-21](#)
- `env`, a `ClientContextEnv` object containing string fields for `platformVersion`, `platform`, `make`, `model`, and `locale`. [src/adapter/aws-lambda/types.ts:12](#)

[src/adapter/aws-lambda/types.ts:23-29](#)

It also declares an optional, capitalized `Custom` member with the `any` type.

[src/adapter/aws-lambda/types.ts:11](#)

The interface declares no methods, runtime validation, thrown errors, or side effects.

[src/adapter/aws-lambda/types.ts:8-13](#) The AWS Lambda adapter passes its received

`LambdaContext` onward as `context` in `streamHandle` and as `lambdaContext` in `handle` ;

neither shown path reads `clientContext` directly. [src/adapter/aws-](#)

[lambda/handler.ts:147-157](#) [src/adapter/aws-lambda/handler.ts:252-272](#)