

transformPatternToRoute

August 18, 2026

transformPatternToRoute

Kind: Function

Source: `packages/microservices/utils/transform-pattern.utils.ts`

Part of: [Microservices](#)

Transforms the Pattern to Route safely.

`transformPatternToRoute` converts a microservice message pattern into a route-like string that can be used for transport-level routing or identification. String patterns are returned unchanged, while object patterns are flattened into slash-separated key/value segments; unsupported pattern values safely resolve to an empty string.

Signature

```
function transformPatternToRoute(pattern: MsPattern, depth, maxDepth, maxKeys): string
```

Parameters

Name	Type
<code>pattern</code>	<code>MsPattern</code>
<code>depth</code>	<code>any</code>
<code>maxDepth</code>	<code>any</code>
<code>maxKeys</code>	<code>any</code>

Returns: `string`

Diagram

```
graph LR
  A[Microservice Pattern] --> B{Pattern type}
  B -->|String| C[Return pattern unchanged]
  B -->|Object| D[Convert each key/value to segment]
  D --> E[Join segments with /]
  E --> F[Route string]
  B -->|Invalid or unsupported| G[Return empty string]
```

Usage

```
import { transformPatternToRoute } from '@nestjs/microservices/utils/transform-pattern.utils';

const stringRoute = transformPatternToRoute('users.create');
// "users.create"

const objectRoute = transformPatternToRoute({
  service: 'users',
  action: 'create',
});
// "service/users/action/create"

const invalidRoute = transformPatternToRoute(null as any);
// ""
```

AI Coding Instructions

- Preserve string patterns exactly; do not normalize, trim, or add route separators to them.
- When passing object patterns, use stable and meaningful key ordering because property order determines the generated route.
- Handle potentially invalid or non-object pattern values before relying on the returned route in downstream transport logic.
- Keep this utility transport-agnostic; transport-specific routing behavior should be implemented by the caller or adapter.

Relationships

- IMPORTS → `isNumber`
- IMPORTS → `isObject`
- IMPORTS → `isString`