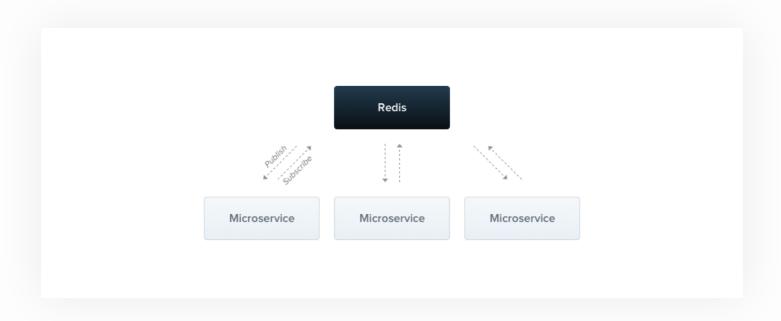


Redis

A second built-in transporter is based on **Redis** database. This transporter takes advantage of **publish/subscribe** feature.



Installation

Before we start, we have to install required package:

```
$ npm i --save redis
```

Overview

In order to switch from TCP transport strategy to Redis **pub/sub**, we need to change an options object passed to the createMicroservice() method.

```
main.ts

const app = await NestFactory.createMicroservice(ApplicationModule, {
   transport: Transport.REDIS,
   options: {
     url: 'redis://localhost:6379',
   },
});
```

Transport enumerator is imported from the @nestjs/microservices package.

Options

There are a bunch of available options that determine a transporter behaviour.

url	Connection url
retryAttempts	A total amount of connection attempts
retryDelay	A connection retrying delay (ms)

Support us

Nest is an MIT-licensed open source project. It can grow thanks to the support by these awesome people. If you'd like to join them, please read more here.

Principal Sponsor



Sponsors / Partners



