

OnApplicationShutdown

August 18, 2026

OnApplicationShutdown

Kind: Interface

Source: `packages/common/interfaces/hooks/on-application-shutdown.interface.ts`

Part of: [Common](#)

Interface defining method to respond to system signals (when application gets shutdown by, e.g., SIGTERM)

`OnApplicationShutdown` is a lifecycle hook interface for providers that need to perform cleanup when the application is shutting down. Implement `onApplicationShutdown()` to respond to operating-system signals such as `SIGTERM` and release resources such as database connections, queues, or background workers.

Diagram

```
graph LR
    Signal[OS signal: SIGTERM / SIGINT] --> App[Nest application]
    App --> Lifecycle[Lifecycle shutdown lifecycle hooks]
    Lifecycle --> Hook[Hook OnApplicationShutdown]
    Hook --> Cleanup[Clean up resources and cleanup]
```

Usage

```
import { Injectable, OnApplicationShutdown } from '@nestjs/common';

@Injectable()
export class DatabaseService implements OnApplicationShutdown {
  async onApplicationShutdown(signal?: string) {
    console.log(`Application shutting down via ${signal ?? 'unknown signal'}`);

    // Close database connections, stop workers, flush telemetry, etc.
    await this.closeDatabaseConnection();
  }
}
```

```
private async closeDatabaseConnection() {  
  // Cleanup implementation  
}  
}  
  
// Enable shutdown hooks during application bootstrap:  
// const app = await NestFactory.create(AppModule);  
// app.enableShutdownHooks();
```

AI Coding Instructions

- Implement `onApplicationShutdown(signal?: string)` in injectable providers that own resources requiring graceful cleanup.
- Enable shutdown hooks with `app.enableShutdownHooks()`; the hook is not invoked for OS signals unless shutdown hooks are enabled.
- Treat the optional `signal` parameter as informational, since shutdown can also be initiated programmatically.
- Make cleanup operations idempotent and safe to run during partial application initialization.
- Await asynchronous cleanup work so connections, queues, and telemetry are properly flushed before process exit.

Used by

3 references from 3 files. Each is a place in this repository where the symbol is actually used — go read one rather than trusting an example.

Imported by (3)

- `ServerGateway` — `integration/websockets/src/server.gateway.ts` :6
- `ClientsModule` — `packages/microservices/module/clients.module.ts` :17
- `callAppShutdownHook` — `packages/core/hooks/on-app-shutdown.hook.ts` :45