

ModuleOverride

August 18, 2026

ModuleOverride

Kind: Interface

Source: `packages/core/interfaces/module-override.interface.ts`

Part of: [Core](#)

`ModuleOverride` defines a replacement mapping between an existing `ModuleDefinition` and a new implementation. It is used by the module resolution or dependency injection system to substitute one module definition with another during configuration or application setup.

Properties

Property	Type
<code>moduleToReplace</code>	<code>ModuleDefinition</code>
<code>newModule</code>	<code>ModuleDefinition</code>

Diagram

```
graph LR
  A[Original ModuleDefinition] --> B[ModuleOverride]
  C[Replacement ModuleDefinition] --> B
  B --> D[Module Resolution / DI System]
  D --> E[Uses newModule instead of moduleToReplace]
```

Usage

```
import type { ModuleOverride } from '@your-scope/core';

const moduleOverride: ModuleOverride = {
  moduleToReplace: {
    name: 'NotificationModule',
```

```

    providers: [EmailNotificationService],
  },
  newModule: {
    name: 'NotificationModule',
    providers: [MockNotificationService],
  },
};

// Pass the override into the application's module configuration.
createApplication({
  moduleOverrides: [moduleOverride],
});

```

AI Coding Instructions

- Set `moduleToReplace` to the exact `ModuleDefinition` targeted by the application configuration.
- Ensure `newModule` is a valid `ModuleDefinition` with compatible exports, providers, and dependencies.
- Use overrides for environment-specific implementations, such as mocks in tests or alternate infrastructure adapters.
- Avoid mutating either module definition after creating the override; treat override configuration as immutable setup data.
- Verify that the module resolution integration consumes overrides before modules are instantiated.

Used by

1 reference from 1 file. Each is a place in this repository where the symbol is actually used — go read one rather than trusting an example.

Imported by (1)

- `TestingModuleOptions` — `packages/testing/testing-module.builder.ts` :29