

InstanceLink

August 18, 2026

InstanceLink

Kind: Interface

Source: `packages/core/injector/instance-links-host.ts`

Part of: [Core](#)

`InstanceLink` describes a resolved dependency entry managed by the injector. It connects an injection `token` to its `InstanceWrapper`, the module-level wrapper `collection` where it was found, and the owning `moduleId` for module-aware resolution.

Properties

Property	Type
<code>token</code>	<code>InjectionToken</code>
<code>wrapperRef</code>	<code>InstanceWrapper<T></code>
<code>collection</code>	<code>Map<any, InstanceWrapper></code>
<code>moduleId</code>	<code>string</code>

Diagram

```
graph LR
    Token[InjectionToken] --> Link[InstanceLink]
    Link --> Wrapper[InstanceWrapper<T>]
    Link --> Collection[Map<any, InstanceWrapper>]
    Link --> ModuleId[moduleId: string]
    Collection --> Wrapper
```

Usage

```

import { InstanceLink } from './instance-links-host';
import { InstanceWrapper } from './instance-wrapper';

function inspectLink<T>(link: InstanceLink<T>) {
  console.log(`Resolved token from module: ${link.moduleId}`);
  console.log('Token:', link.token);
  console.log('Wrapper:', link.wrapperRef);

  // The collection contains all provider wrappers for the owning module.
  const registeredWrapper = link.collection.get(link.token);

  return registeredWrapper ?? link.wrapperRef;
}

// Typically created internally by the injector during dependency resolution.
const link: InstanceLink<MyService> = {
  token: MyService,
  wrapperRef: myServiceWrapper,
  collection: moduleProviders,
  moduleId: 'AppModule',
};

```

AI Coding Instructions

- Treat `wrapperRef` as the authoritative wrapper selected during dependency resolution; use `collection` when module-level lookup context is needed.
- Preserve the original `token`, including string, symbol, class, and custom injection-token values.
- Keep `moduleId` associated with the link so cross-module resolution and diagnostics retain module context.
- Do not assume `collection.get(token)` always returns the same wrapper as `wrapperRef`; aliases and lookup rules may affect resolution.

Relationships

- IMPORTS → `InjectionToken`
- IMPORTS → `isFunction`