

Unit Testing HTTP Interactions with ngMock



Bradley Braithwaite

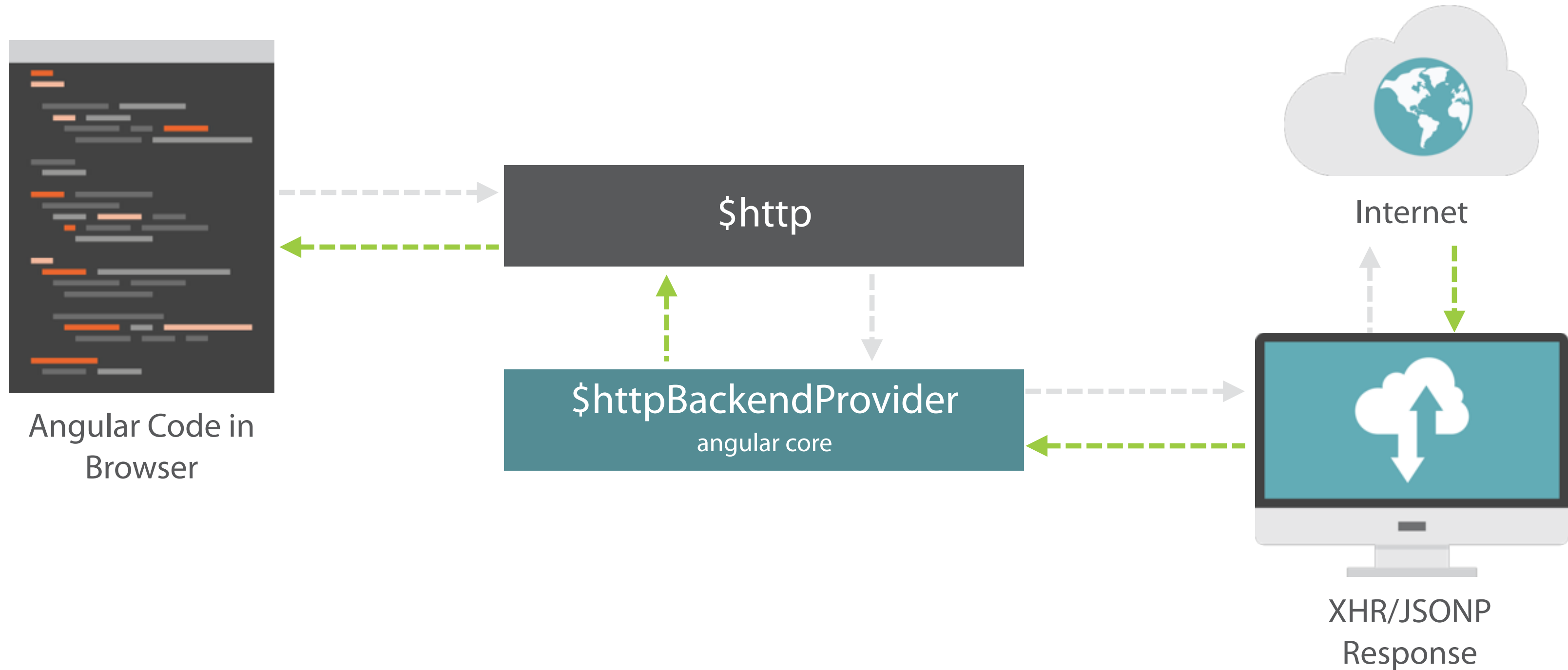
@bradoncode | www.bradoncode.com

How do we use Angular's `$http` service
in our unit tests?

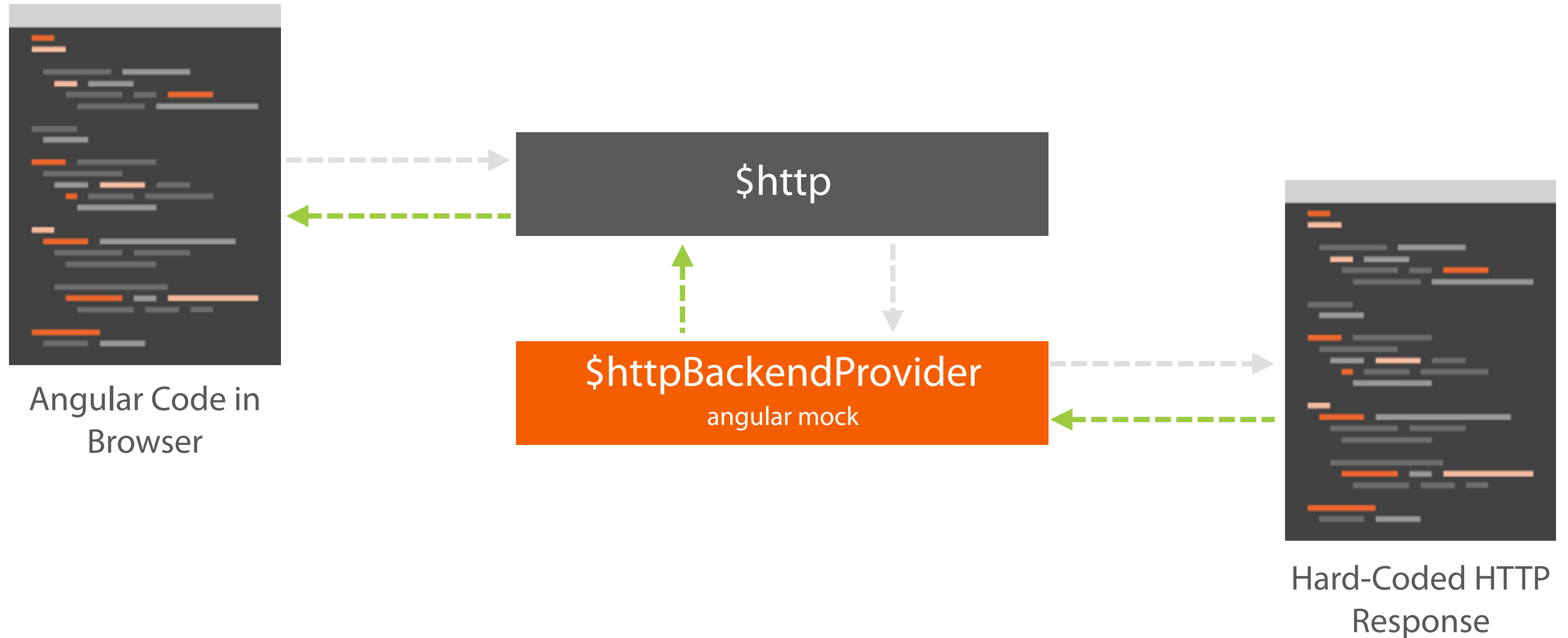
\$httpBackend

Fake HTTP backend implementation suitable for unit testing applications that use the \$http service.

Using `$http` in Angular Application Code



Using `$http` in Angular Unit Tests



\$httpBackend functions

when

Specifies a backend definition

expect

Specifies a request expectation

when Function

Specifies Backend Definition

```
$httpBackend  
  .when( 'GET', 'http://localhost/api' )  
  .respond( 200, { foo: 'bar' } );
```

expect Function

Specifies a Request Expectation

```
$httpBackend
```

```
  .expect( 'GET', 'http://localhost/api' )
```

```
  .respond( 200, { foo: 'bar' } );
```


Supported HTTP Verbs

GET

POST

PUT / PATCH

DELETE

HEADER

JSONP

when vs. expect

when

- Black-box Testing
- Best for Working with Data
- Use in Any Order
- Re-use Allowed

expect

- Strict Usage Testing
- Best for Testing Exact Usage
- Order Matters
- Re-use **Not** Allowed

when vs. expect

when

http request A

http request B

http request C

expect

http request A

http request B

http request C

when vs. expect

when

http request A

http request B

http request C

expect

http request A

http request B

http request C

when vs. expect

when

http request A

http request B

http request C

expect

http request B

http request C

when vs. expect

when

http request A

http request B

http request C

expect

http request C

when vs. expect

when

http request A

http request B

http request C



expect



when vs. expect

when

http request A

http request B

http request C

expect

http request A

http request B

http request C

when vs. expect

when

http request A

http request B

http request C

expect

http request B

http request C

when vs. expect

when

http request A

http request B

http request C



expect

http request B

http request C

