

Configuration Stub Expectations

You can monitor stubs execution to check against certain expectations, such as the number of calls to a given stub or the sequence of stub calls. Expectations must be defined inside a test cases and are verified after the test case execution is finished. If the expectation is not met, then a suitable message is generated and output to the results. All expectations are removed at the end of test case execution after the outcome has been verified.

i Stub expectations must precede all calls to the tested function to ensure that all stub calls are counted.

i You can configure stub expectations for subs that are created with a test configuration where one of the following options is enabled:

- **Enable Stub Callbacks**
- **Enable Stub API (deprecated)**

See [Dynamic Stubs Configuration](#) for additional information.

The following table describes the available expectation types:

Expectation	Description
CPPTEST_EXPECT_NCALLS(<func-id>, <value>)	The number of calls to the <func-id> stub is expected to be equal to <value>.
CPPTEST_EXPECT_NCALLS_MORE_THAN(<func-id>, <value>)	The number of calls to the <func-id> stub is expected to be greater than <value>.
CPPTEST_EXPECT_NCALLS_LESS_THAN(<func-id>, <value>)	The number of calls to the <func-id> stub is expected to be less than <value>.
CPPTEST_EXPECT_NCALLS_IN_RANGE(<func-id>, <min>, <max>)	The number of calls to the <func-id> stub is expected to be greater than <min> but less than <max>.
CPPTEST_EXPECT_CALL_SEQUENCE() [Add<func-id> AddNTimes<func-id>]	Specified the expected sequence for calls made to the stubs

Examples

The following examples present typical ways to set stub definition expectations.

Number of Stub Calls Expectations

i The API that sets expectations for the number of stub calls is the same for C and C++.

- **Defining an expectation for the exact number of calls to the stub:**
In the following example, the "foo" stub is expected to be called exactly 3 times.
`CPPTEST_EXPECT_NCALLS("foo", 3);`
- **Defining an expectation for the stub call number to be greater than value:**
In the following example, the "foo" stub is expected to be called more than 3 times.
`CPPTEST_EXPECT_NCALLS_MORE_THAN("foo", 3);`
- **Defining an expectation for the stub call number to be lower than value:**
In the following example, the "foo" stub is expected to be called less than 3 times.
`CPPTEST_EXPECT_NCALLS_LESS_THAN("foo", 3);`
- **Defining an expectation for the stub call number to be in range:**
In the following example, the "foo" stub is expected to be called between 3 and 7 times.
`CPPTEST_EXPECT_NCALLS_IN_RANGE("foo", 3, 7);`

Sequence of Stub Calls Expectation

Defining an expectation for the stub call sequence:

The following example sets expectations for a `foo -> goo -> foo (3times) -> goo` stub call sequence:

```
[C] CPPTEST_EXPECT_CALL_SEQUENCE()->Add("foo")->Add(goo)->AddNTimes("foo", 3)->Add("goo");
[C++] CPPTEST_EXPECT_CALL_SEQUENCE().Add("foo").Add(goo).AddNTimes("foo", 3).Add("goo");
```

The defined expectation as shown in this example is violated in the following situations:

- Stub specified in the sequence was not called in the predicted order
- At the end of the test case, some functions in the expected sequence were not called