




ISOLATION (MOCK OBJECT) FRAMEWORKS



Mock Object framework

- a reusable library that can create and configure stub and mock objects at **runtime**.
 - These objects are usually referred to as **dynamic stubs** and **dynamic mocks**.
- 

Isolation framework

- DEFINITION: An *isolation framework* is a set of programmable APIs that make creating mock and stub object much easier. Isolation framework save the developer from the need to write repetitive code to test or simulate object interaction
- C++: mockpp
- Java: jMock and EasyMock
- .net: Nmock, Moq, Typemock Isolator and Rhino Mocks.

OK...Let's take a look at an Mock object implementation

Suppose the mock object needs to implement the following interfaces

```
namespace AOUT.CH5.LogAn
{
    public interface IComplicatedInterface
    {
        void Method1(string a, string b, bool c, int x, object o);
        void Method2(string b, bool c, int x, object o);
        void Method3(bool c, int x, object o);
    }
}
```

An simple Mock object that simply save the states

```
public class
MytestableComplicatedInterface:IComplicatedInterface
{
    public string meth1_a;
    public string meth1_b,meth2_b;
    public bool meth1_c,meth2_c,meth3_c;
    public int meth1_x,meth2_x,meth3_x;
    public int meth1_o,meth2_o,meth3_o;

    public void Method1(string a, string b,
bool c, int x, object o)
    {
        meth1_a = a;
        meth1_b = b;
        meth1_c = c;
        meth1_x = x;
        meth1_o = o;
    }
}
```

```
public void Method2(string b, bool c, int x,
object o)
{
    meth2_b = b;
    meth2_c = c;
    meth2_x = x;
    meth2_o = o;
}

public void Method3(bool c, int x, object
o)
{
    meth3_c = c;
    meth3_x = x;
    meth3_o = o;
}
}
```

Can you see the problems?

- Mock object can be time-consuming to implement
- Complicated mock object may require
 - To test that a method is called several times?
 - Return a specific value based on the parameter it receives
 - To remember all the values for all the method calls

Dynamically creating a fake object

- DEFINITION: A *dynamic fake object* is any stubs or mock that is created at runtime without needing to use a handwritten implementation of an interface or subclass

Isolation frameworks (.net)

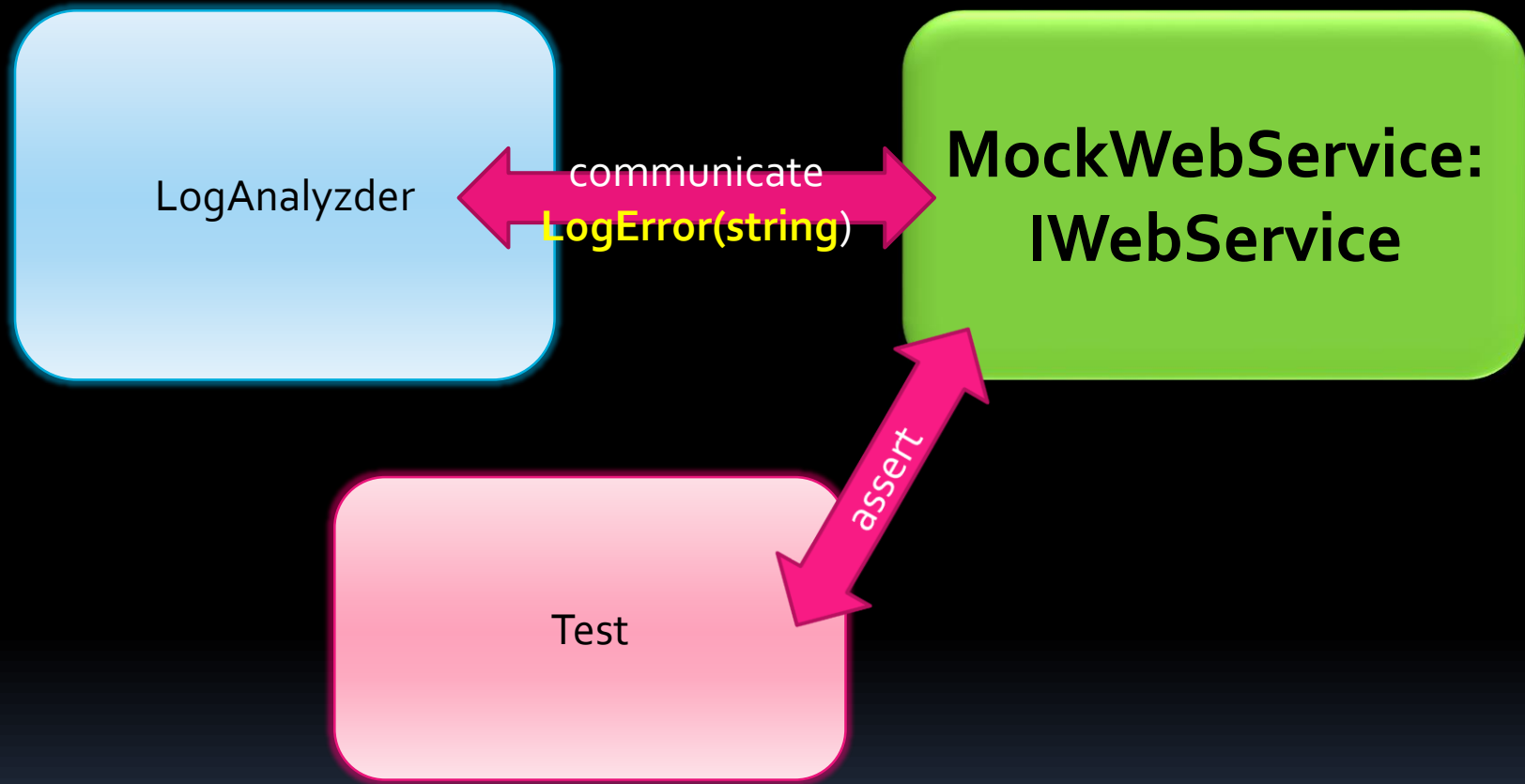
- Rhino mocks
- Moq
- Handwritten stubs
- Type mock isolator
- Nmocks
- jMock
- Google mocking framework
- http://en.wikipedia.org/wiki/List_of_mock_object_frameworks#C.2B.2B

Rhino Mocks

- <http://ayende.com/blog/tags/rhino-mocks>
- API documentation:
<http://ayende.com/Wiki/Rhino+Mocks+Documentation.aspx?AspxAutoDetectCookieSupport=1>



LogAnalyzer with a Mock: A reminder



The CUT communicates with the mock object and all communication is recorded in the mock. The test uses the mock object to verify that the test passes

The example of using Rhino mocks in the tests

[Test]

```
public void CreateMock_RecordSimpleExpectation_  
{  
    MockRepository mockEngine = new MockRepository();  
    IWebService simulatedService = mockEngine.DynamicMock<  
using(mockEngine.Record())  
{  
    simulatedService.LogError("Filename too short:abc.ext");  
}  
  
    LogAnalyzer log = new LogAnalyzer(simulatedService);  
    string tooShortFileName = "abc,ext" ;  
    log.Analyze(tooShortFileName);  
  
    mockEngine.VerifyAll();  
}
```

Set
expectation

Invoke
LogAnalyzer

Assert the
expectation
have been met

