.NET Unit Testing with AutoFixture

GETTING STARTED WITH AUTOFIXTURE

Jason Roberts
.NET DEVELOPER

@robertsjason dontcodetired.com

The Benefits of AutoFixture

Increase productivity

Reduce test maintenance

Improve test readability

Less test code

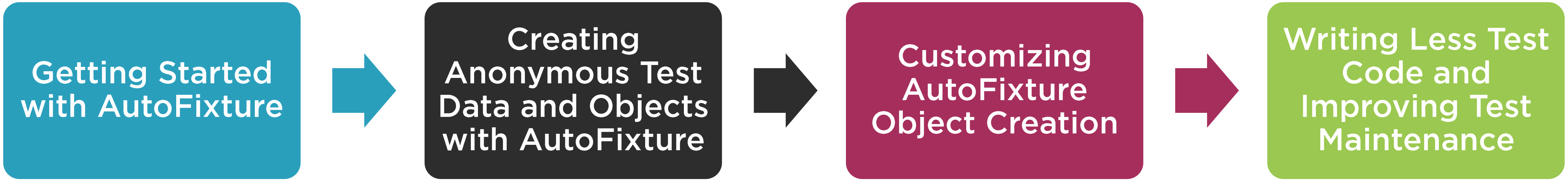
The Benefits of AutoFixture

```
[Fact]
public void SubtractWhenZeroTest()
{
    // Arrange
    var sut = new Calculator();

    // Act
    sut.Subtract(1);

    // Assert
    Assert.True(sut.Value < 0);
}
```

The Benefits of AutoFixture

```
[Theory, AutoData]
public void SubtractWhenZeroTest(int aPositiveNumber,
                                 Calculator sut)
{
    // Act
    sut.Subtract(aPositiveNumber);

    // Assert
    Assert.True(sut.Value < 0);
}
```

Course Overview



Getting Started with AutoFixture

Creating Anonymous Test Data and Objects with AutoFixture

Customizing AutoFixture Object Creation

Writing Less Test Code and Improving Test Maintenance

Overview

Simplify the Arrange phase of tests

Introduce anonymous test data concept

Supported frameworks

Introduce the Fixture class

Create a test project and install AutoFixture

Write an initial test with non-anonymous test data

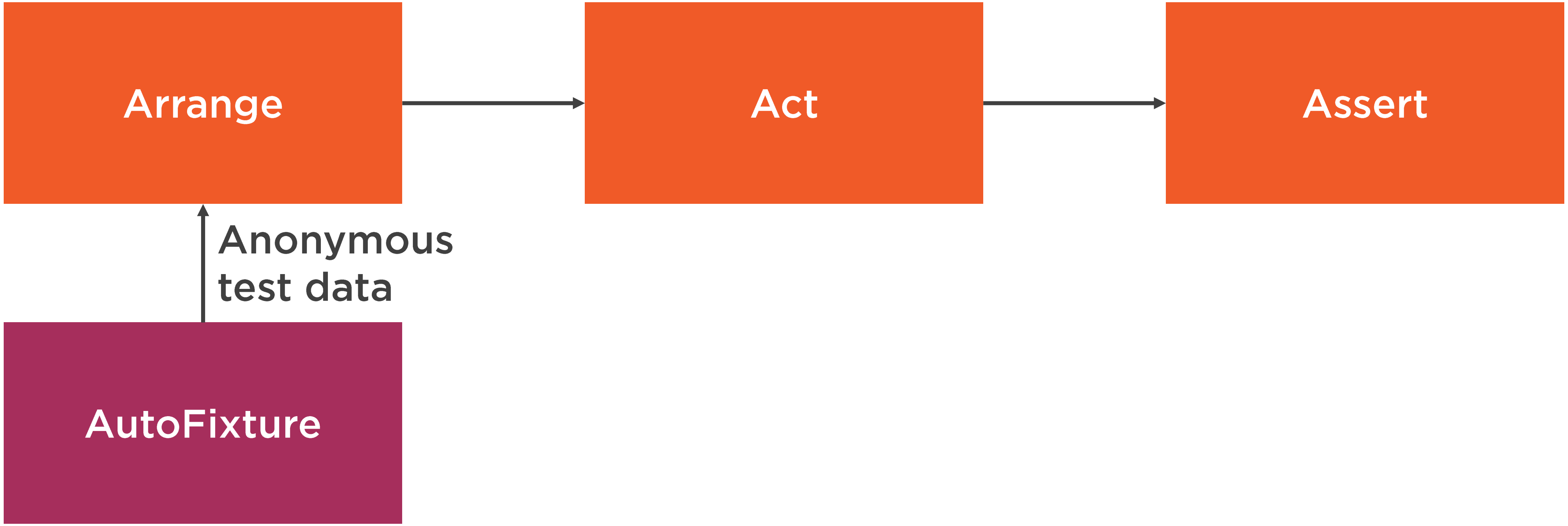
Write a test with manual anonymous test data

Use AutoFixture to create anonymous test data

Simplify and reduce

Simplify and reduce arrange phase code

The Arrange, Act, and Assert Phases



Arrange

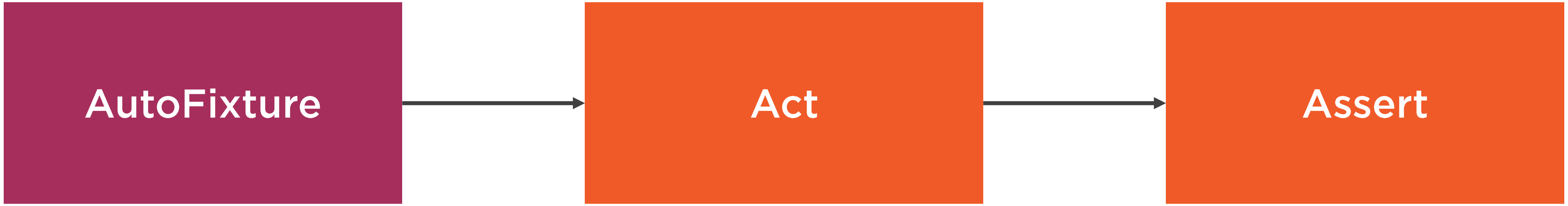
Act

Assert

Anonymous test data

AutoFixture

The Arrange, Act, and Assert Phases



Anonymous Test Data

Data that is required to be present for the test to be able to execute, but where the value itself is unimportant.

The Benefits of AutoFixture

```
[Fact]
public void SubtractWhenZeroTest()
{
    // Arrange
    var sut = new Calculator();

    // Act
    sut.Subtract(1);

    // Assert
    Assert.True(sut.Value < 0);
}
```

Known test data

The Benefits of AutoFixture

```
[Theory, AutoData]
public void SubtractWhenZeroTest(int aPositiveNumber,
                                 Calculator sut)
{
    // Act
    sut.Subtract(aPositiveNumber);

    // Assert
    Assert.True(sut.Value < 0);
}
```

Anonymous test data

"...library for .NET designed to minimize the 'Arrange' phase of your unit tests in order to maximize maintainability. Its primary goal is to allow developers to focus on what is being tested rather than how to setup the test scenario..."

AutoFixture documentation

https://github.com/AutoFixture/AutoFixture

Supported Test Frameworks

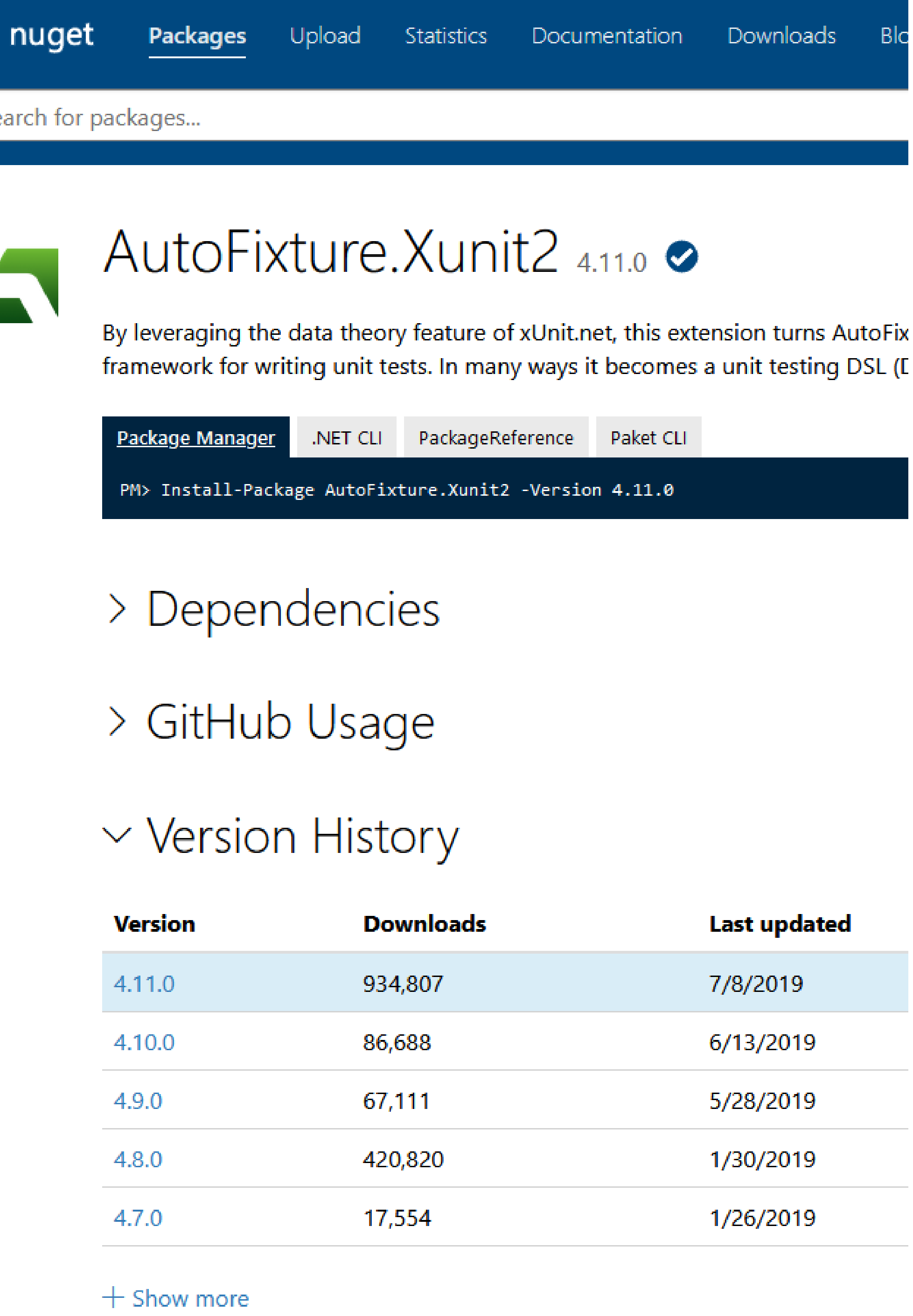
NUnit

xUnit.net

MSTest

Fixie

AutoFixture is independent of the testing framework or test runner.



AutoFixture.Xunit2 4.11.0

By leveraging the data theory feature of xUnit.net, this extension turns AutoFix
framework for writing unit tests. In many ways it becomes a unit testing DSL (D

```
PM> Install-Package AutoFixture.Xunit2 -Version 4.11.0
```

Version	Downloads	Last updated
4.11.0	934,807	7/8/2019
4.10.0	86,688	6/13/2019
4.9.0	67,111	5/28/2019
4.8.0	420,820	1/30/2019
4.7.0	17,554	1/26/2019

Testing framework specific packages:

- AutoFixture.Xunit
- AutoFixture.Xunit2
- AutoFixture.NUnit2
- AutoFixture.NUnit3

Add deeper integration with a specific testing framework

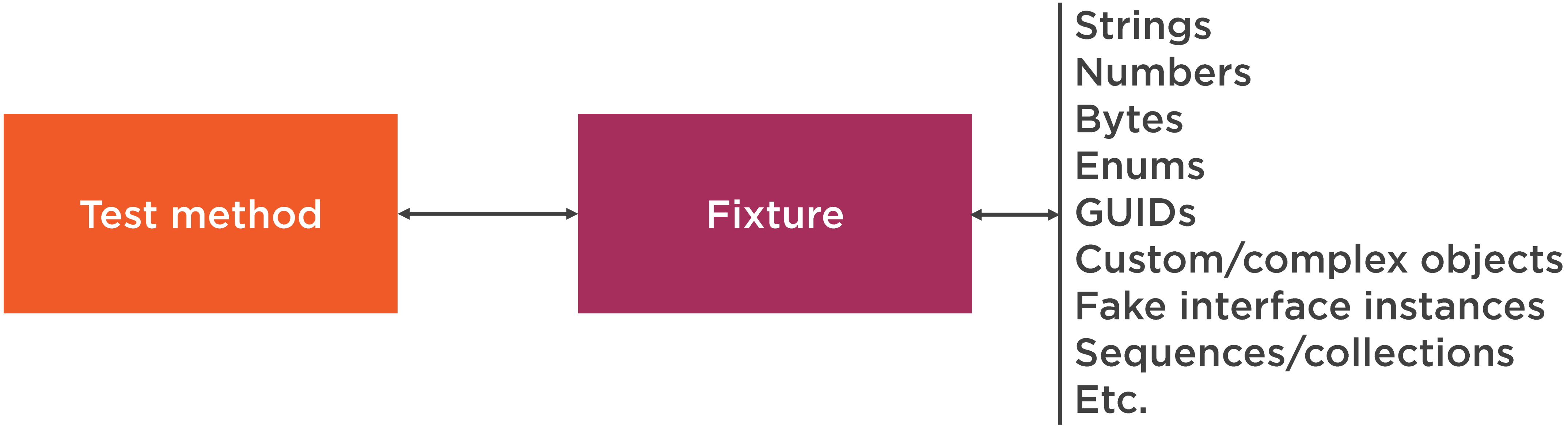
.NET Support

.NET Framework	.NET Standard
4.5.2	1.5
	2.0

NuGet Packages

- AutoFixture
- AutoFixture.SeedExtensions
- AutoFixture.xUnit / AutoFixture.xUnit2
- AutoFixture.NUnit2 / AutoFixture.NUnit3
- AutoFakeItEasy
- AutoFoq
- AutoMoq
- AutoNSubstitute
- AutoRhinoMock
- Idioms
- Idioms.FsCheck

Introducing the Fixture Class



Test method

Fixture

Strings
Numbers
Bytes
Enums
GUIDs
Custom/complex objects
Fake interface instances
Sequences/collections
Etc.

You can also create anonymous test data without having to manually create Fixture instance.

(later in the course)

Summary

Simplify the Arrange phase of tests

Anonymous test data

Supported frameworks

Introduced the Fixture class

Created a test project and installed AutoFixture NuGet package

Wrote an initial test with non-anonymous test data

Wrote a test with manual anonymous test data

fixture.Create<int>()

Up Next:
Creating Anonymous Test Data and Objects with AutoFixture