



From 0-1000: A Test Driven Approach to Tools Development

David Paris
Senior Engineer, Playground Games
Twitter: @davidjpparis

GAME DEVELOPERS CONFERENCE
MARCH 16-20, 2020 | #GDC20

Today's Journey

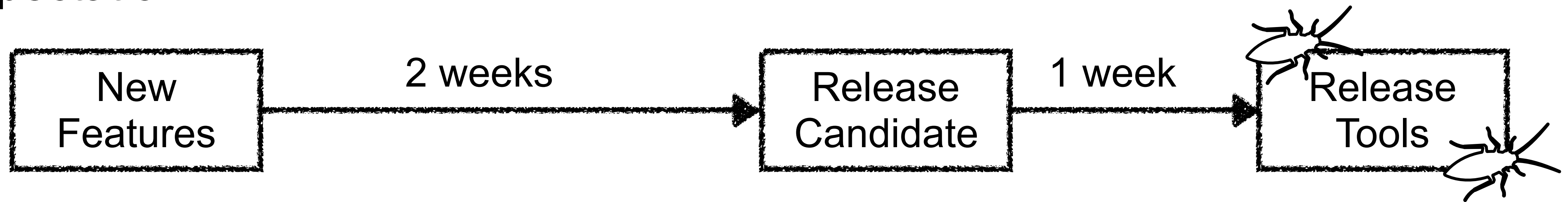
- What is a test driven culture?
- Steps to cultivate a test driven culture
- Future plans
- Key takeaways

What is a Test Driven Culture?



Path to Stability

Expectation:



Reality:



What is a Test Driven Culture?

A culture where **developers, stakeholders** and **users** feel ***confident*** tool changes work and retain ***stability***, without hindering ***creativity*** and ***iteration***

Automated Testing

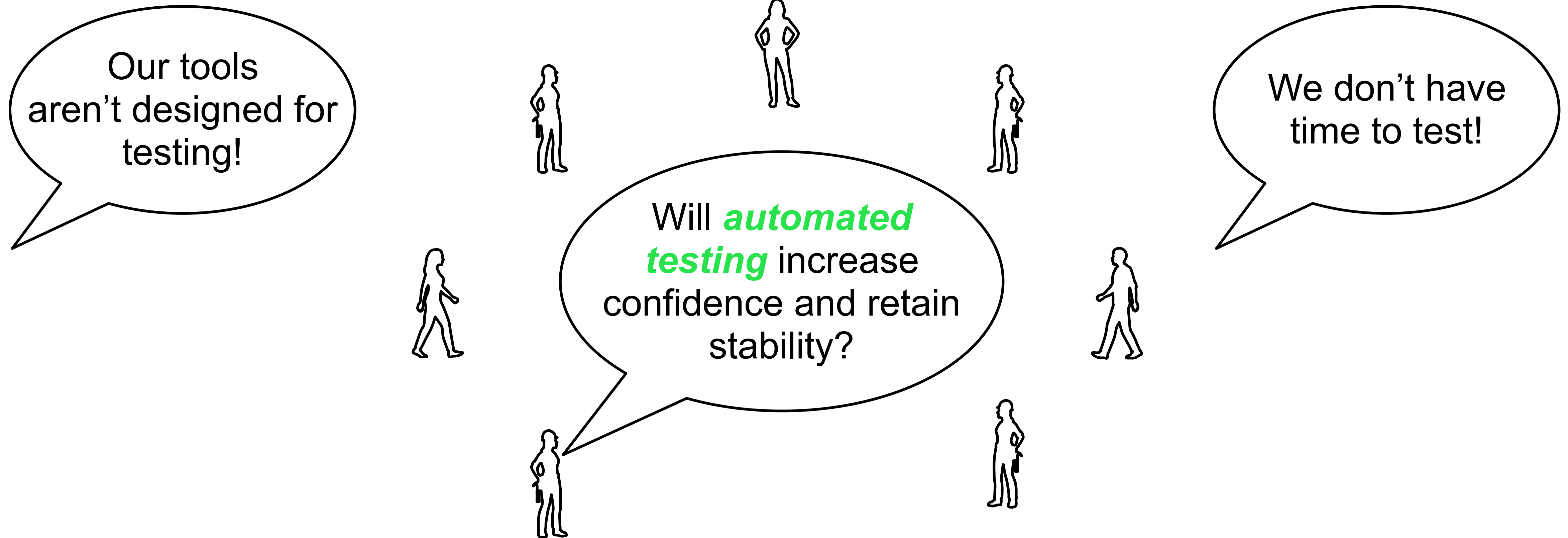
How do we change our culture?

Any culture shift takes ***time, mistakes*** and ***constant evaluation***

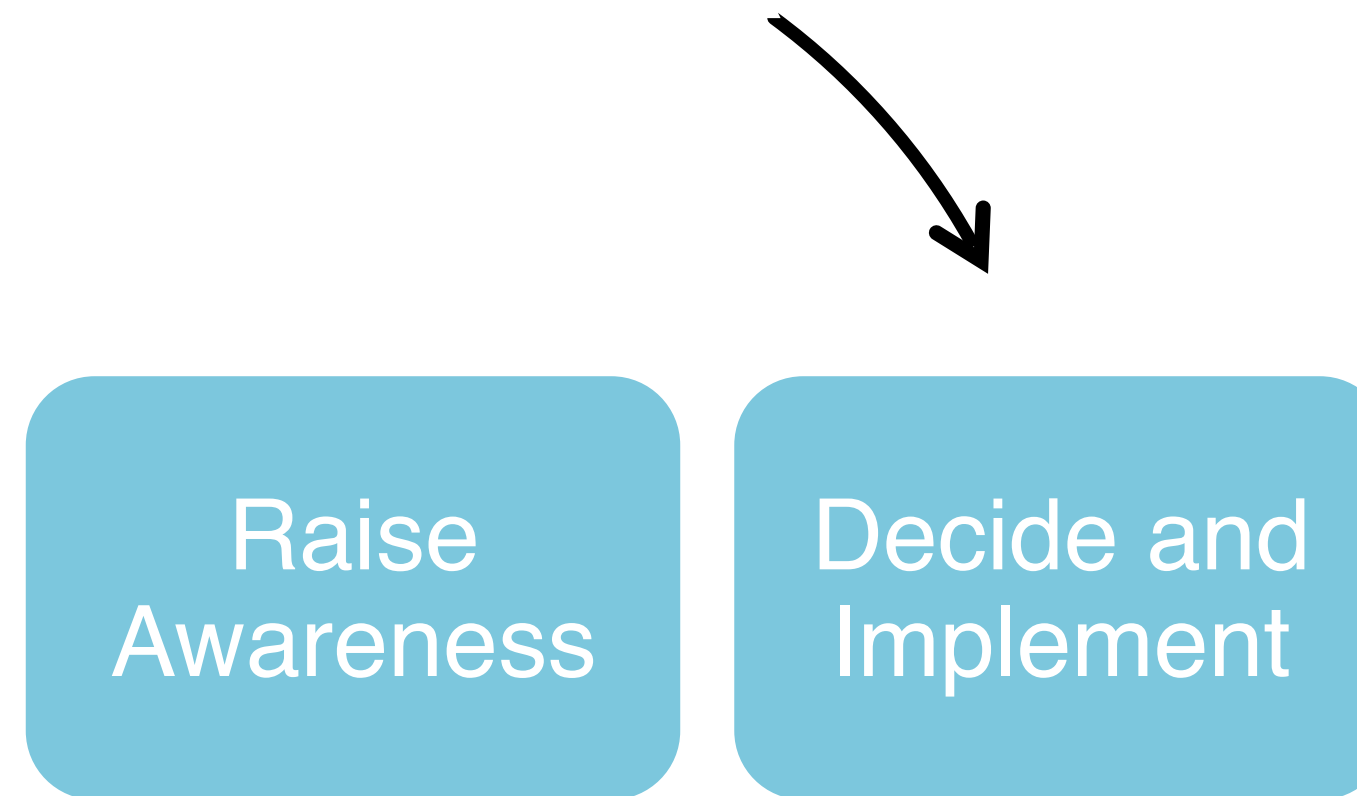
Steps to Cultivate a Test Driven Culture

Raise
Awareness

Step 1: Raise Awareness



Steps to Cultivate a Test Driven Culture

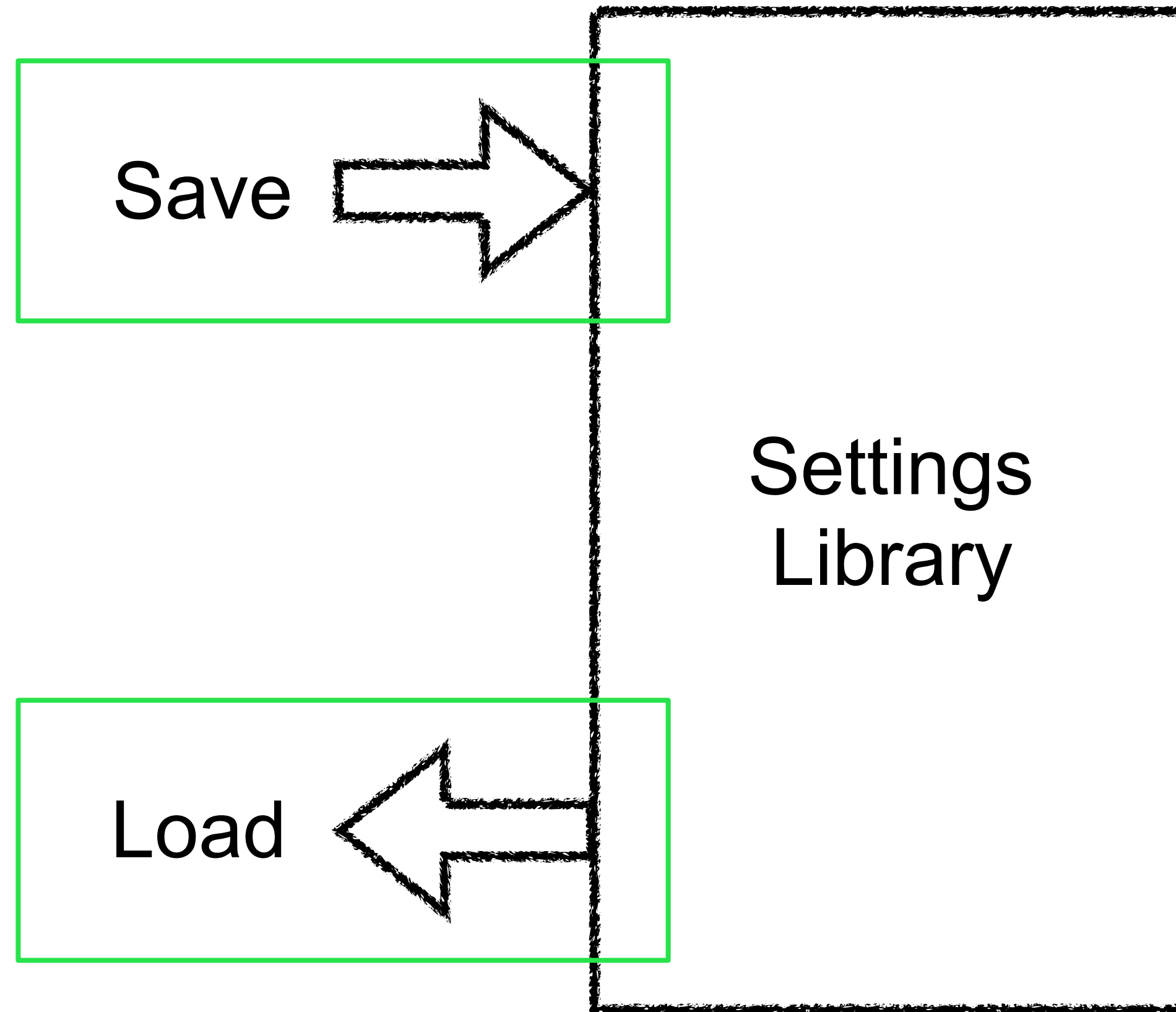


Step 2: Decide and Implement



You're more likely to succeed if you ***start small***

Step 2: Decide and Implement



Write test, see it ***fail***.
Write code, see test ***pass***.
Refactor.

Step 2: Decide and Implement

```
33
34 [TestMethod()]
35 0 references
36 public void SettingExistsReturnsFalseWhenNoFileExistsTest()
37 {
38     SettingsEditor manager = new SettingsEditor(m_ResourcesPath, c_TestSettingsFolderName);
39     bool settingFound = manager.SettingExists(TestSettingsContext.TEST_SETTING_NO_APPLICATION);
40     Assert.IsFalse(settingFound);
41 }
42 #endregion
43
44 #region read to string tests
45
46 [TestMethod()]
47 0 references
48 public void GetSetting1AsAStringTest()
49 {
50     SettingsEditor manager = new SettingsEditor(m_ResourcesPath, c_TestSettingsFolderName);
51
52     string setting;
53     bool settingFound = manager.GetSetting(TestSettingsContext.TEST_SETTING_STRING_1, out setting);
54
55     Assert.IsTrue(settingFound);
56     Assert.AreEqual("settingValue1", setting);
57 }
```

Test Explorer

Run All | Run... | Playlist: All Tests

FuelAll (1769 tests)

- PG.Core.SettingsTests (115) 458 ms
 - PG.Core.SettingsTests (115) 458 ms
 - SessionSettingsEditorTests (58) 72 ms
 - SettingsEditorTests (57) 385 ms
 - DeleteSettingDeletesAChildPathTest 7 ms
 - DeleteSettingDeletesAParentPathTest 7 ms
 - DeleteSettingsDoesNothingIfFileDoesNotExistTest 1 ms
 - GetNotSetSettingAsAStringWithDefaultTest < 1 ms
 - GetNotSetSettingAsBoolWithDefaultTest < 1 ms
 - GetNotSetSettingAsDictionaryWithDefaultTest < 1 ms
 - GetNotSetSettingAsEnumWithDefaultTest 1 ms
 - GetNotSetSettingAsEnumerableWithDefaultTest < 1 ms
 - GetNotSetSettingAsFloatWithDefaultTest 1 ms
 - GetNotSetSettingAsIntWithDefaultTest < 1 ms
 - GetSetting1AsADictionaryReturnsFalseTest < 1 ms
 - GetSetting1AsAEnumReturnsFalseTest < 1 ms
 - GetSetting1AsAEnumerableReturnsTrueTest < 1 ms
 - GetSetting1AsAStringTest < 1 ms
 - GetSetting1AsAStringWithDefaultReturnsSetting1Test < 1 ms
 - GetSetting1AsAnIntReturnsFalseTest < 1 ms
 - GetSetting2AsAStringTest 1 ms
 - GetSettingAsADictionaryTest 1 ms
 - GetSettingAsAEnumNameTest 2 ms
 - GetSettingAsAEnumValueTest < 1 ms
 - GetSettingAsAEnumerableTest < 1 ms
 - GetSettingAsAFloatTest < 1 ms

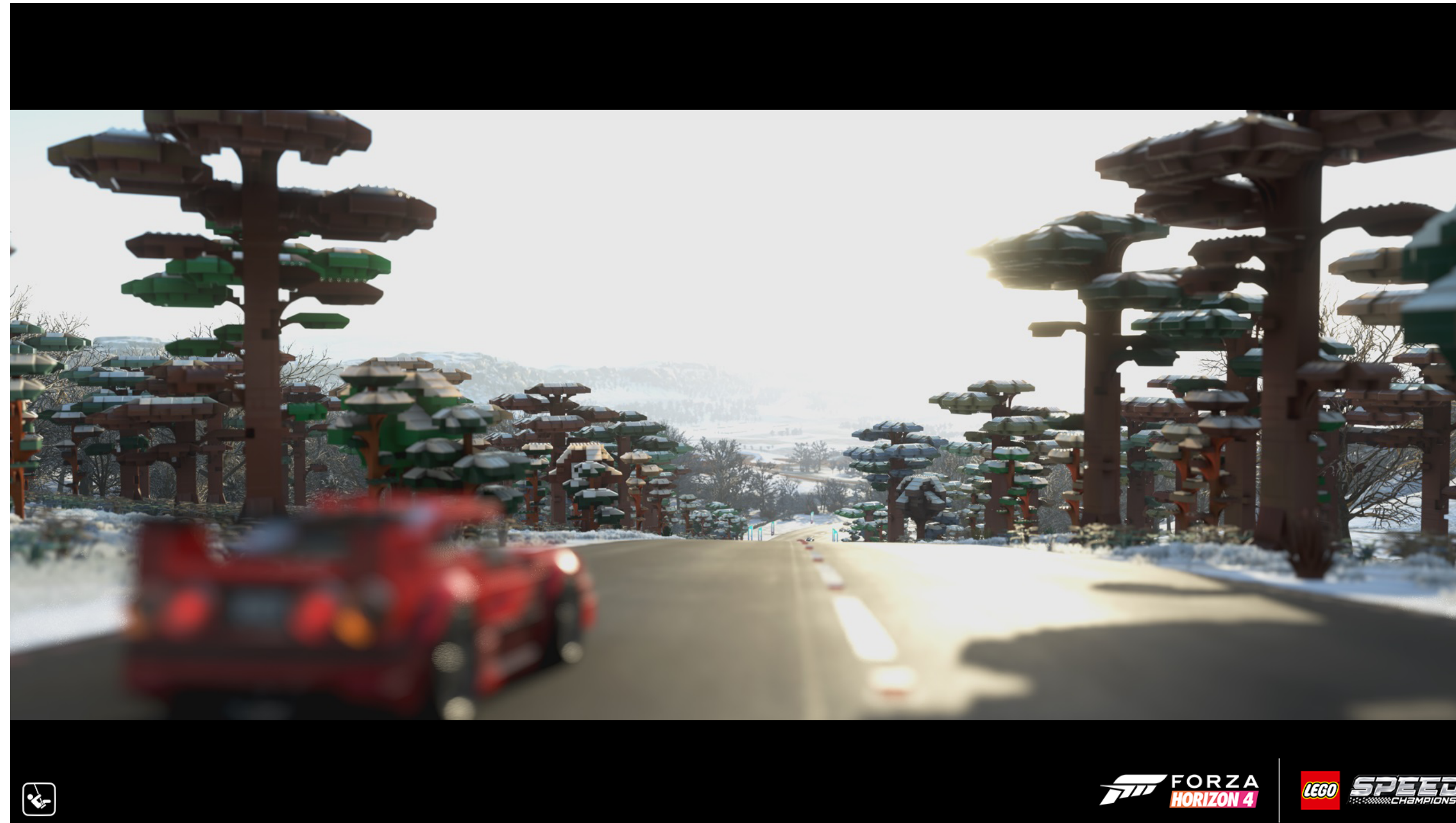
Group Summary Copy All

Grouped by Hierarchy: PG.Core.SettingsTests

Duration: 0:00:00.4587056

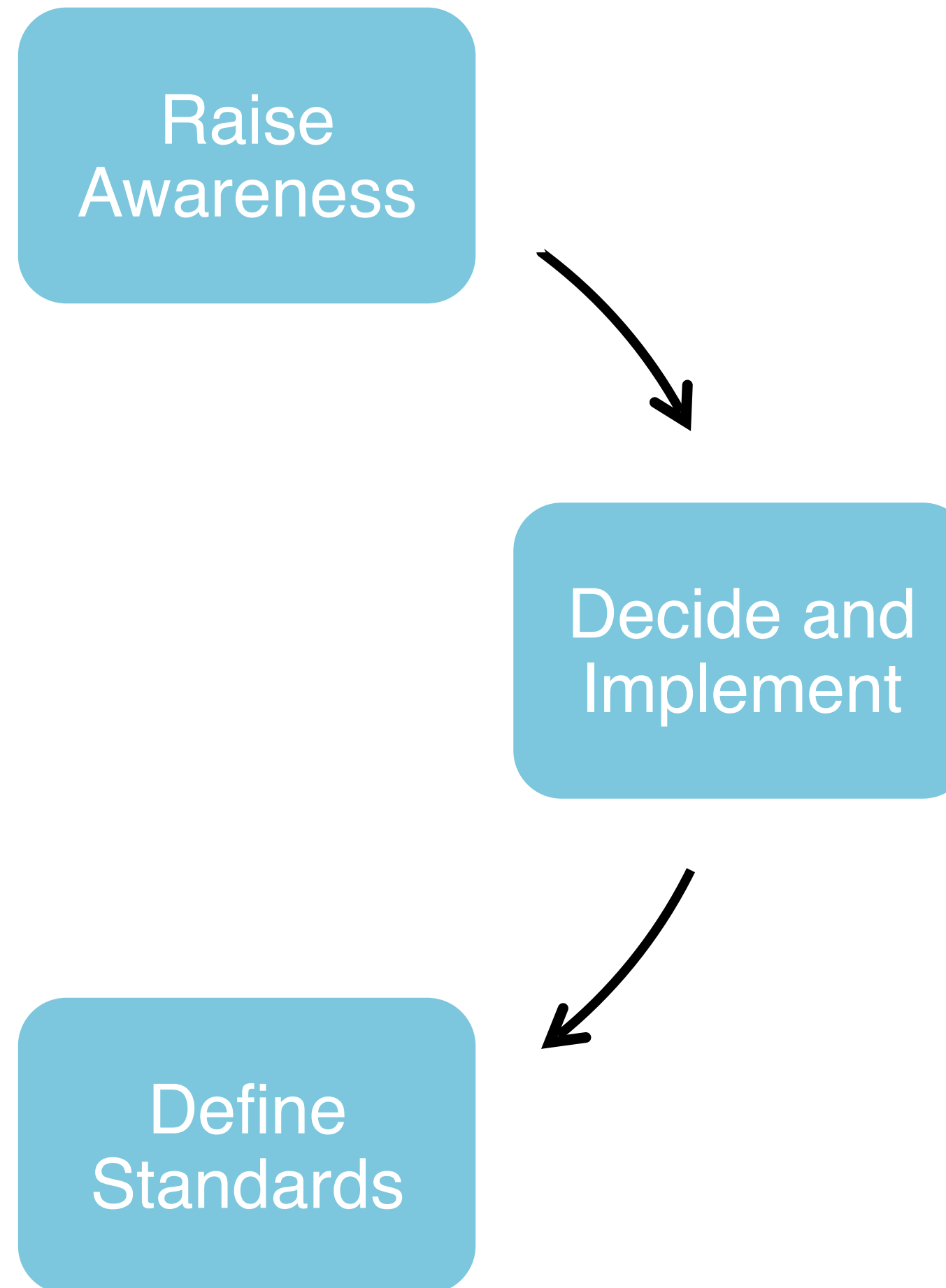
✓ 115 Tests Passed

Step 2: Decide and Implement



Take *initiative* to solve a problem

Steps to Cultivate a Test Driven Culture

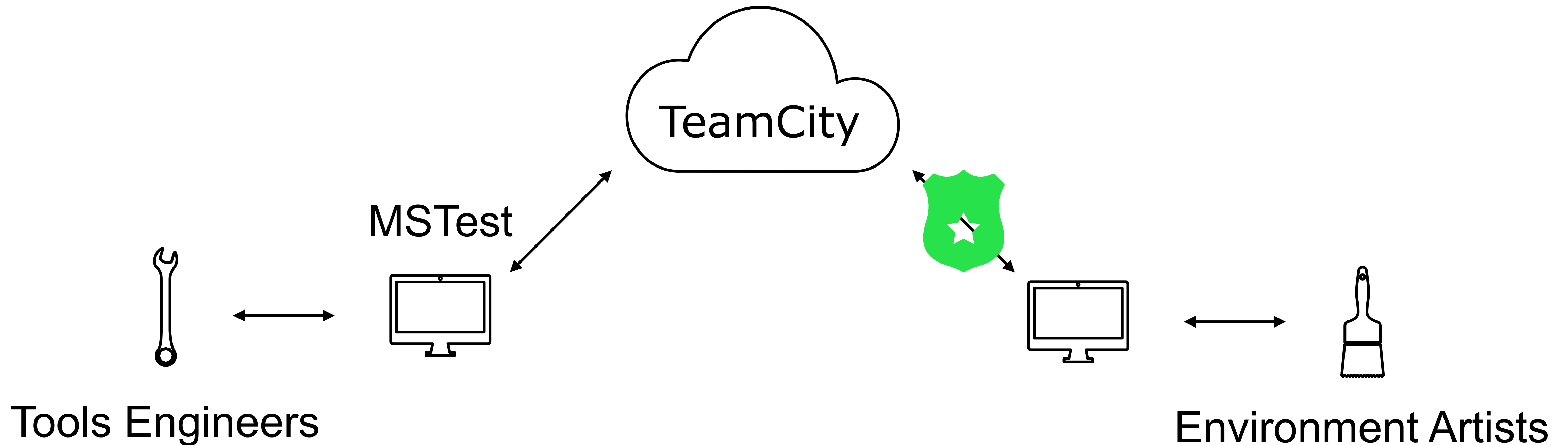


Step 3: Define Standards

Minimise the *barrier to entry*.

Step 3: Define Standards

Focus on the workflow *as a whole*.



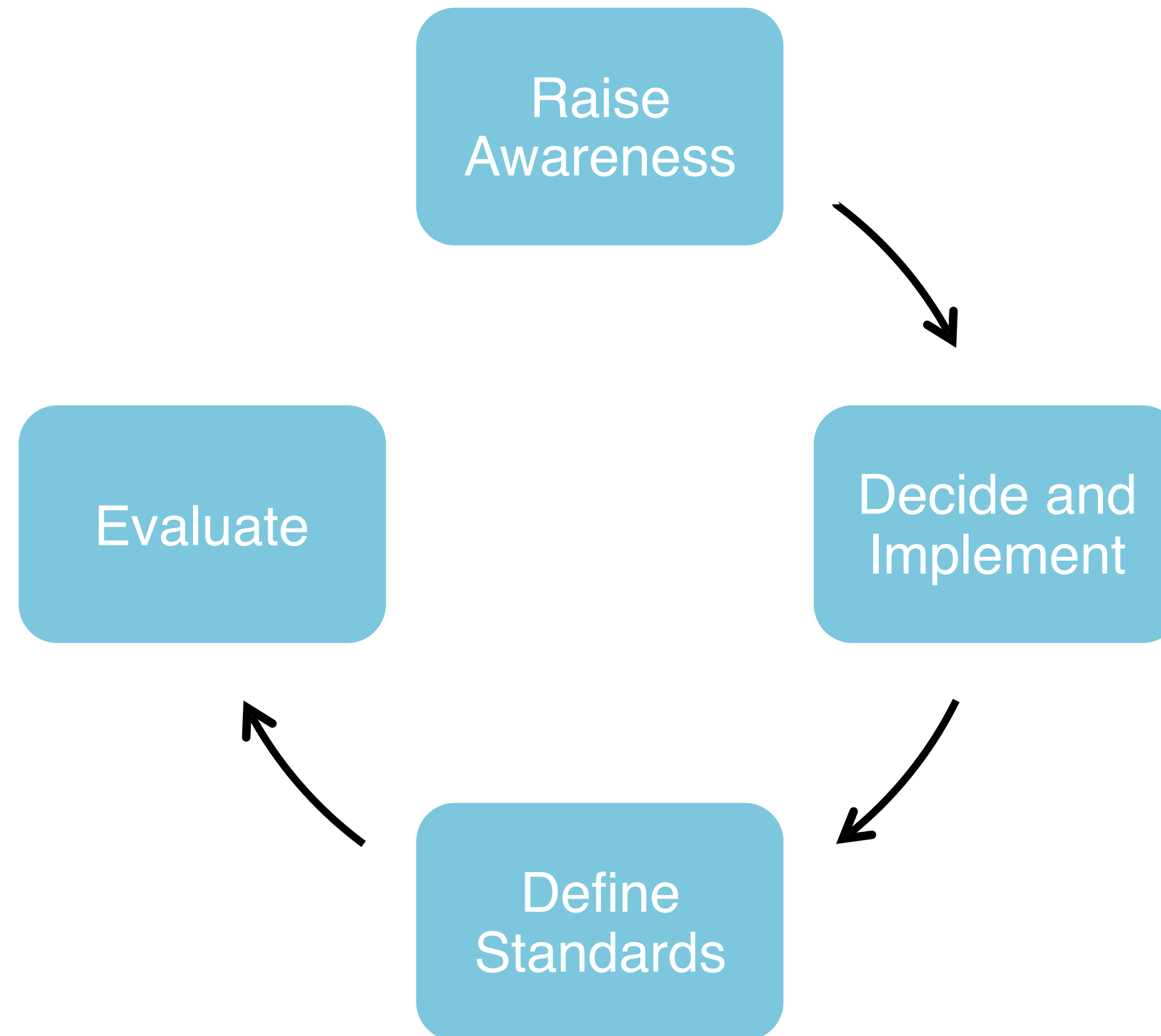
Step 3: Define Standards

Test to ***prevent bugs*** rather than to find bugs.

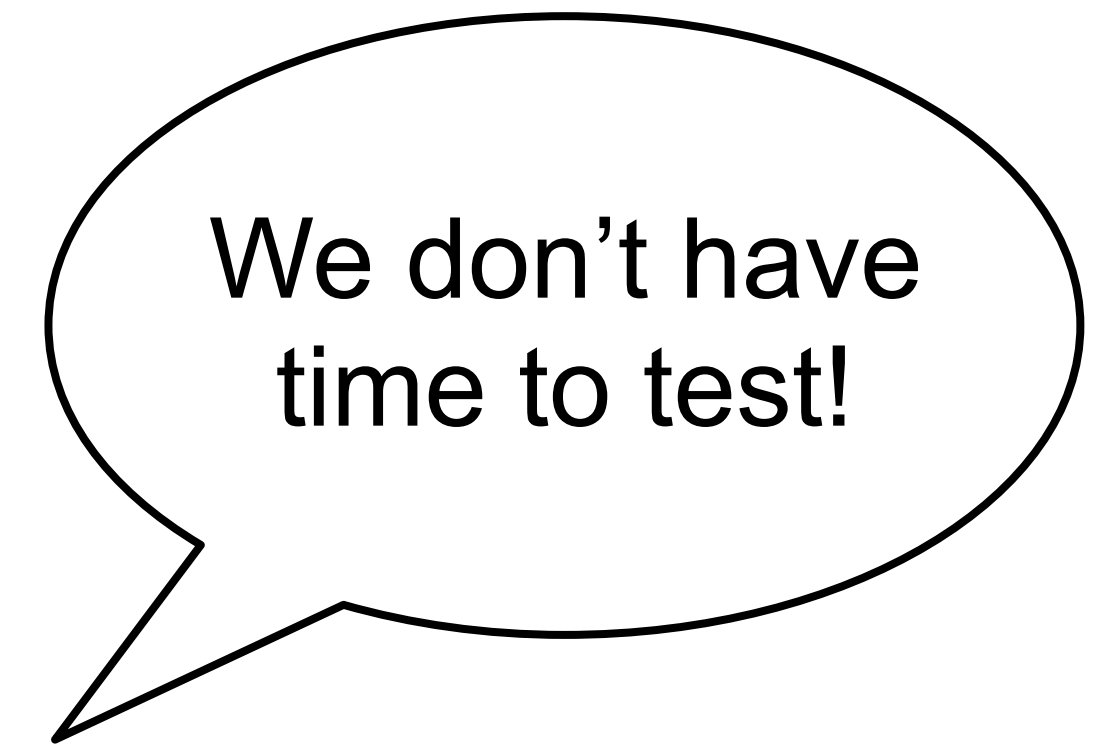
Step 3: Define Standards

- Use external resources
 - Working Effectively with Legacy Code (Michael C. Feathers)
 - Martin Fowler's online blog (martinfowler.com)

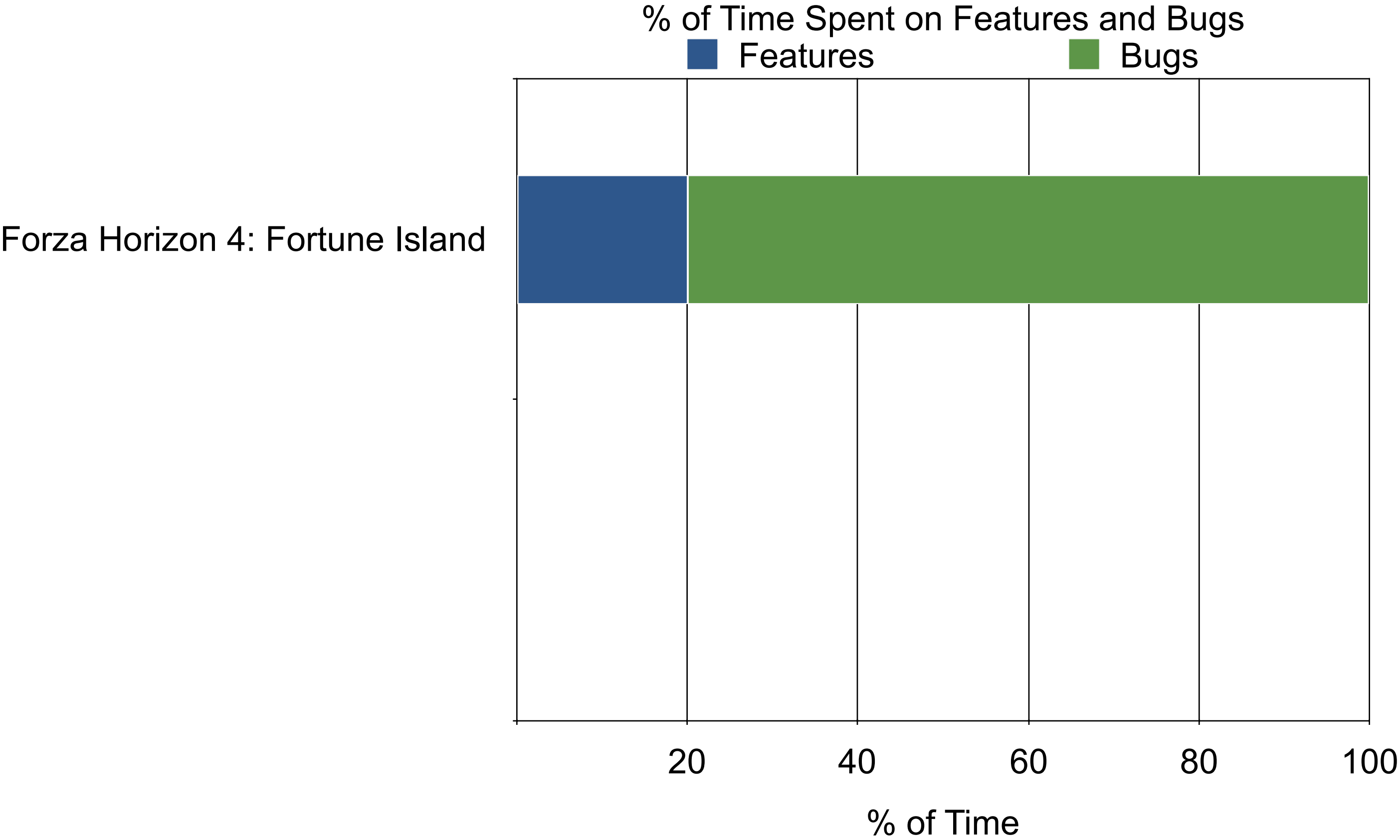
Steps to Cultivate a Test Driven Culture



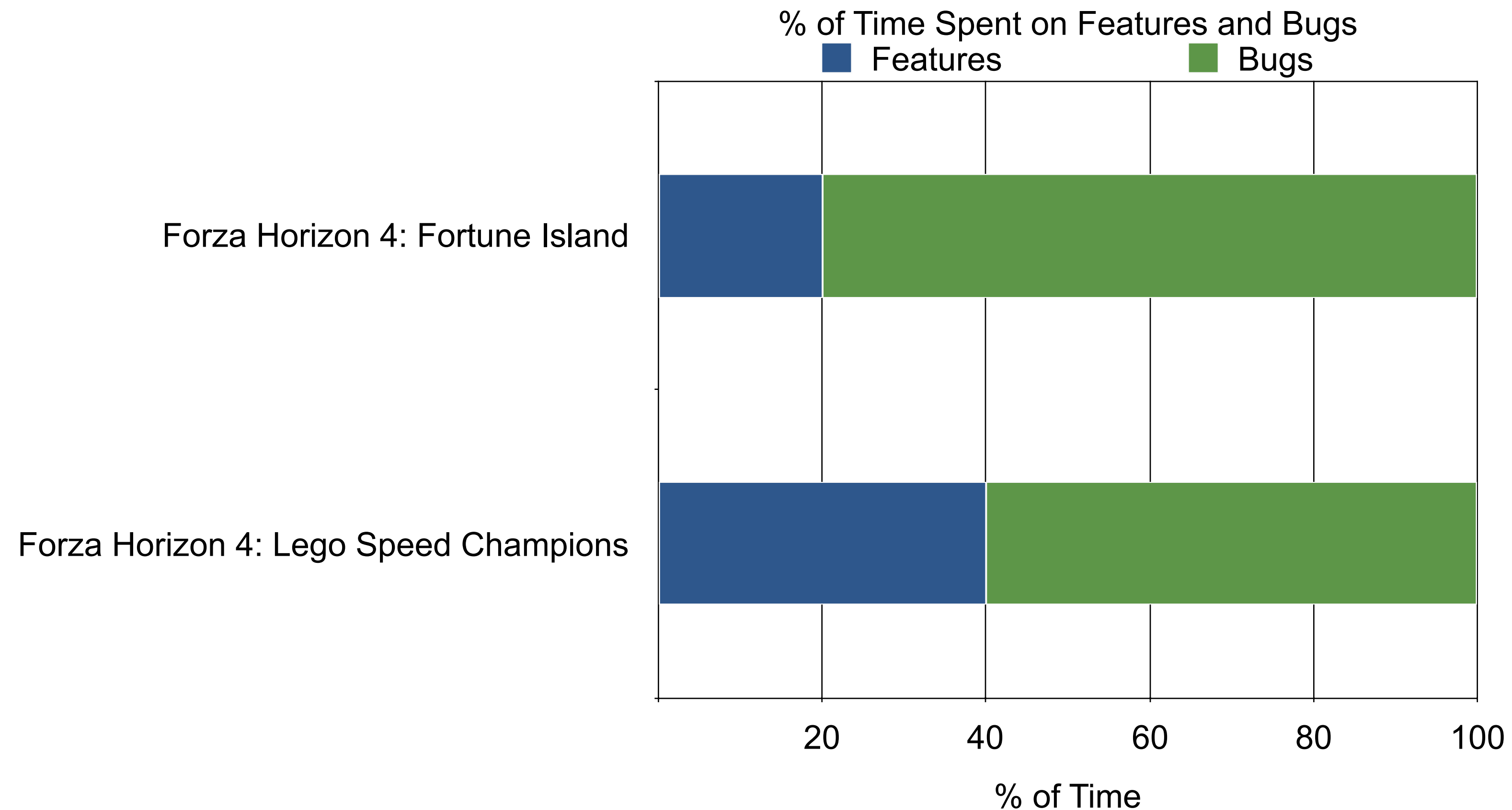
Step 4: Evaluate



Step 4: Evaluate



Step 4: Evaluate



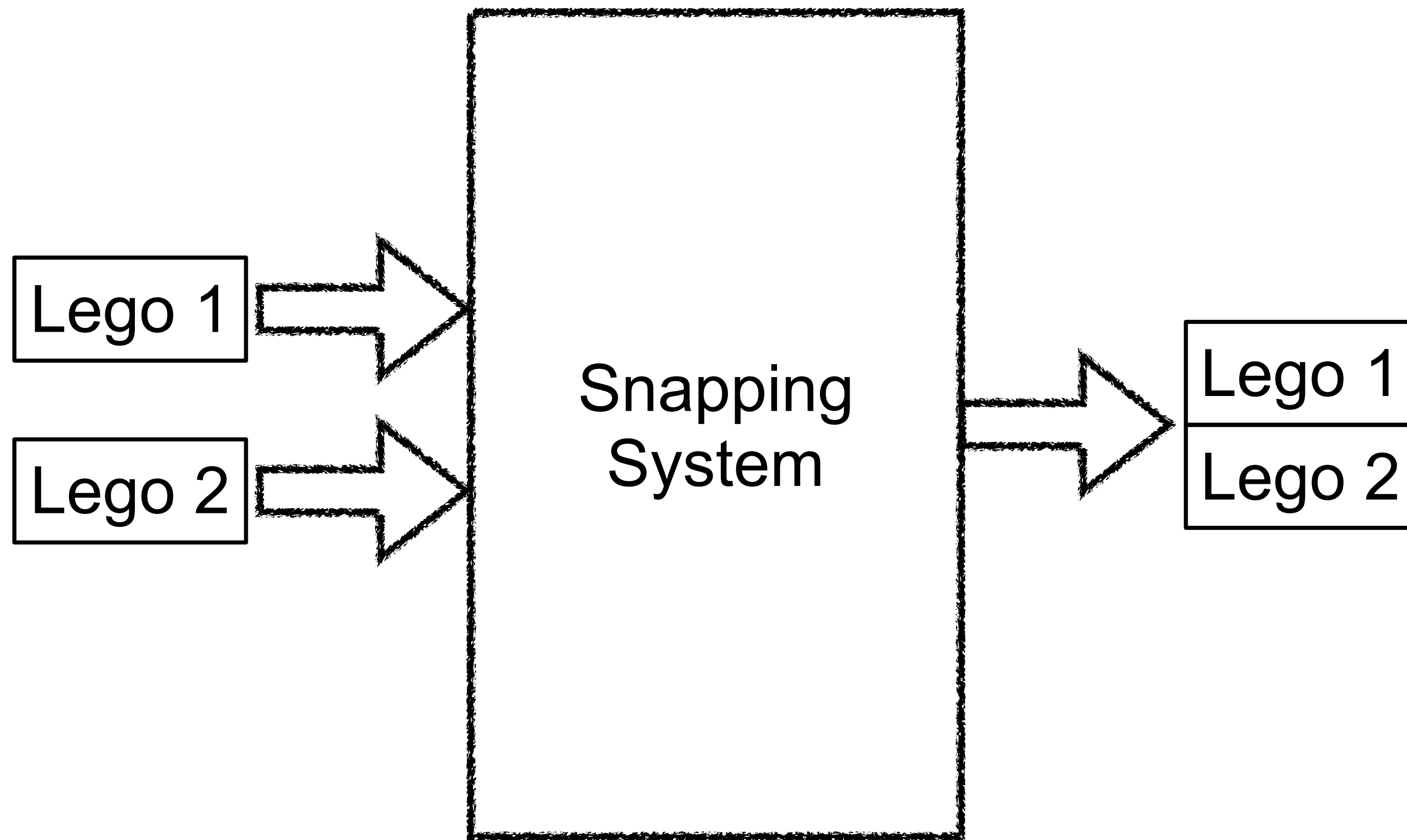
1 month more of features vs. fixing bugs in a 5 month cycle

Step 4: Evaluate



Lego Speed Champions snapping system

Step 4: Evaluate



Write test, see it ***fail***.
Write code, see test ***pass***.
Refactor.

Tests ***do not*** mean you have
written the ***correct system***.

Step 4: Evaluate



Tests lead to *better collaboration*

What does the system do if...

Before Tests

Open Editor

Place Assets

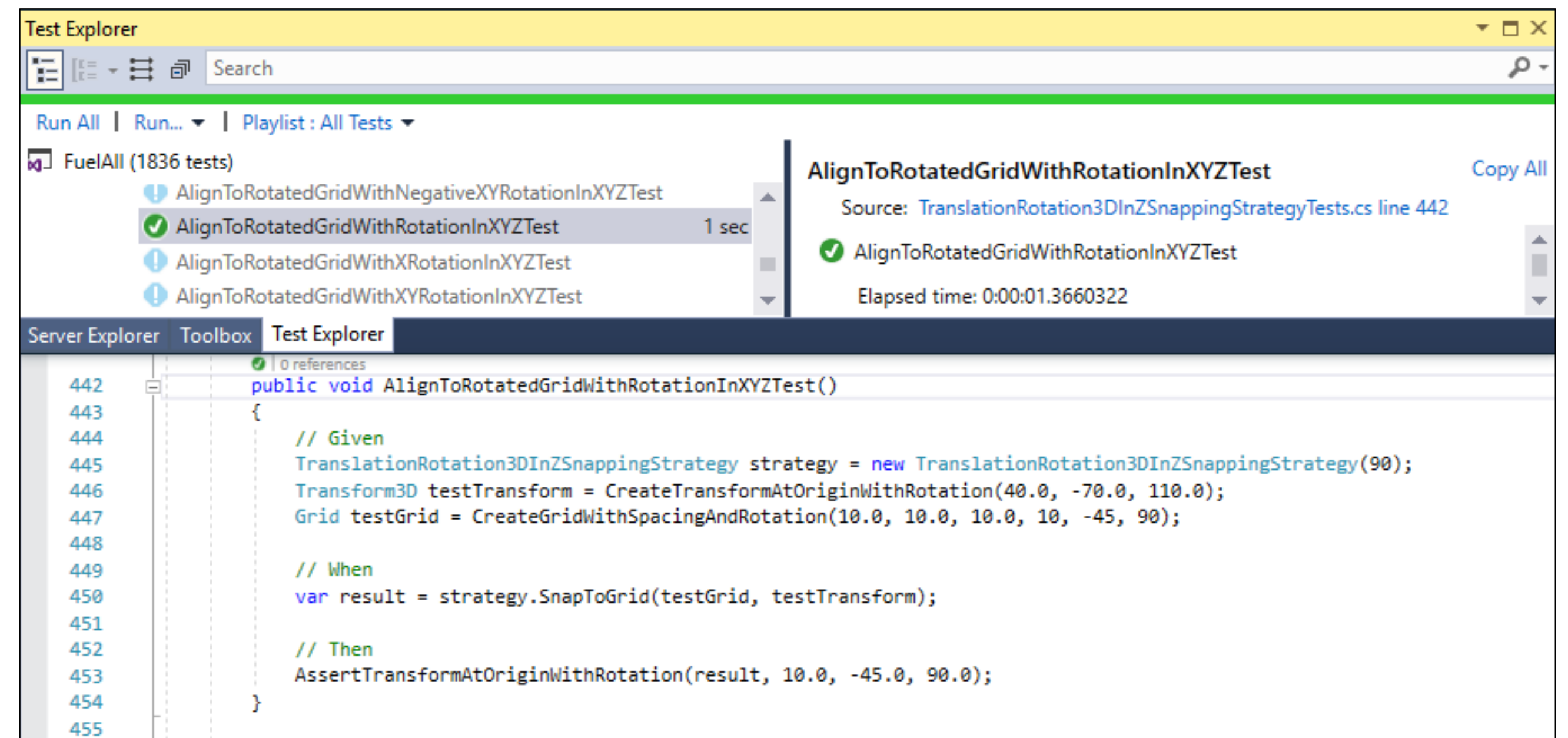
Setup Position

Run Snapping

Examine

Reset Position

After Tests

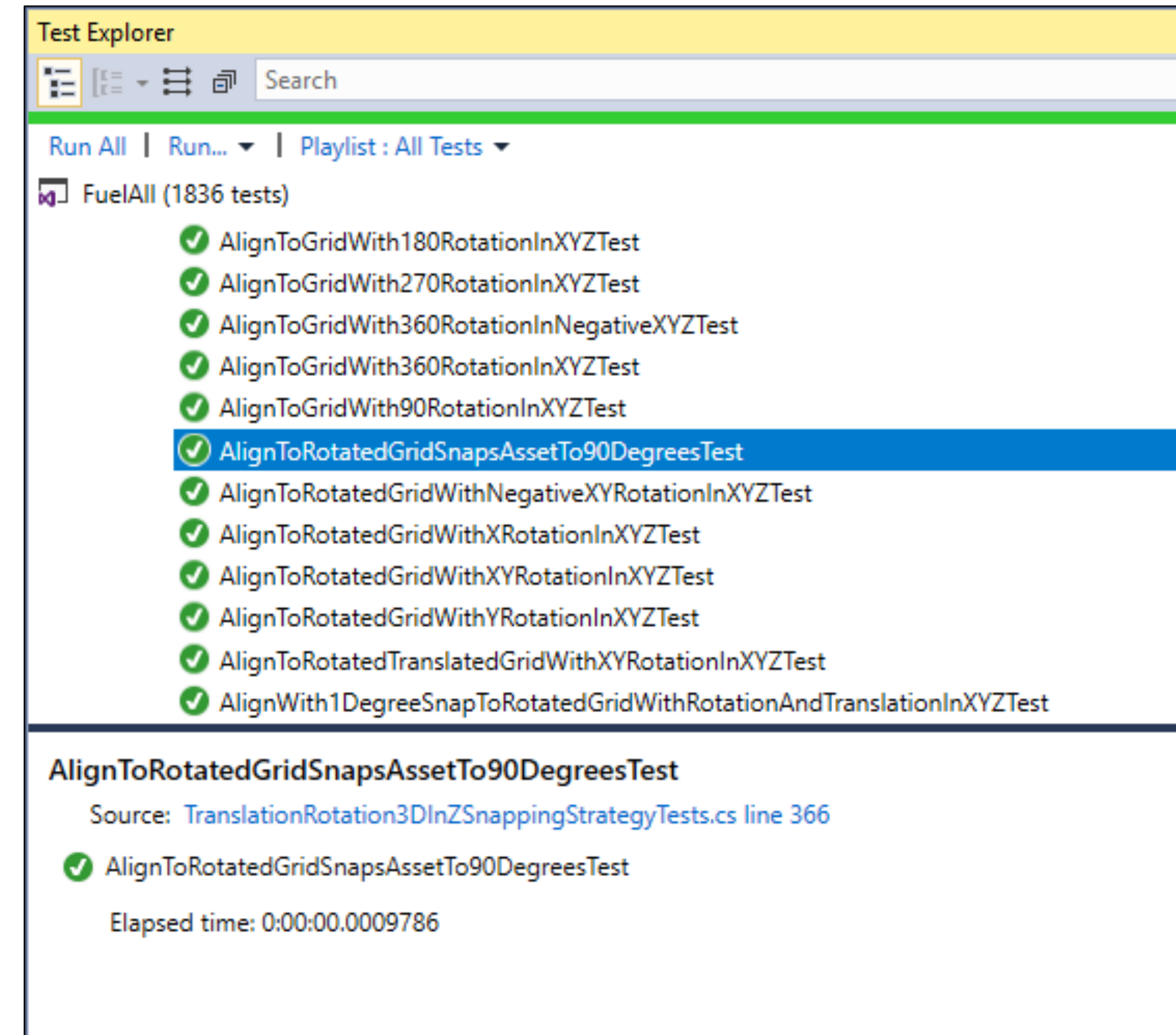
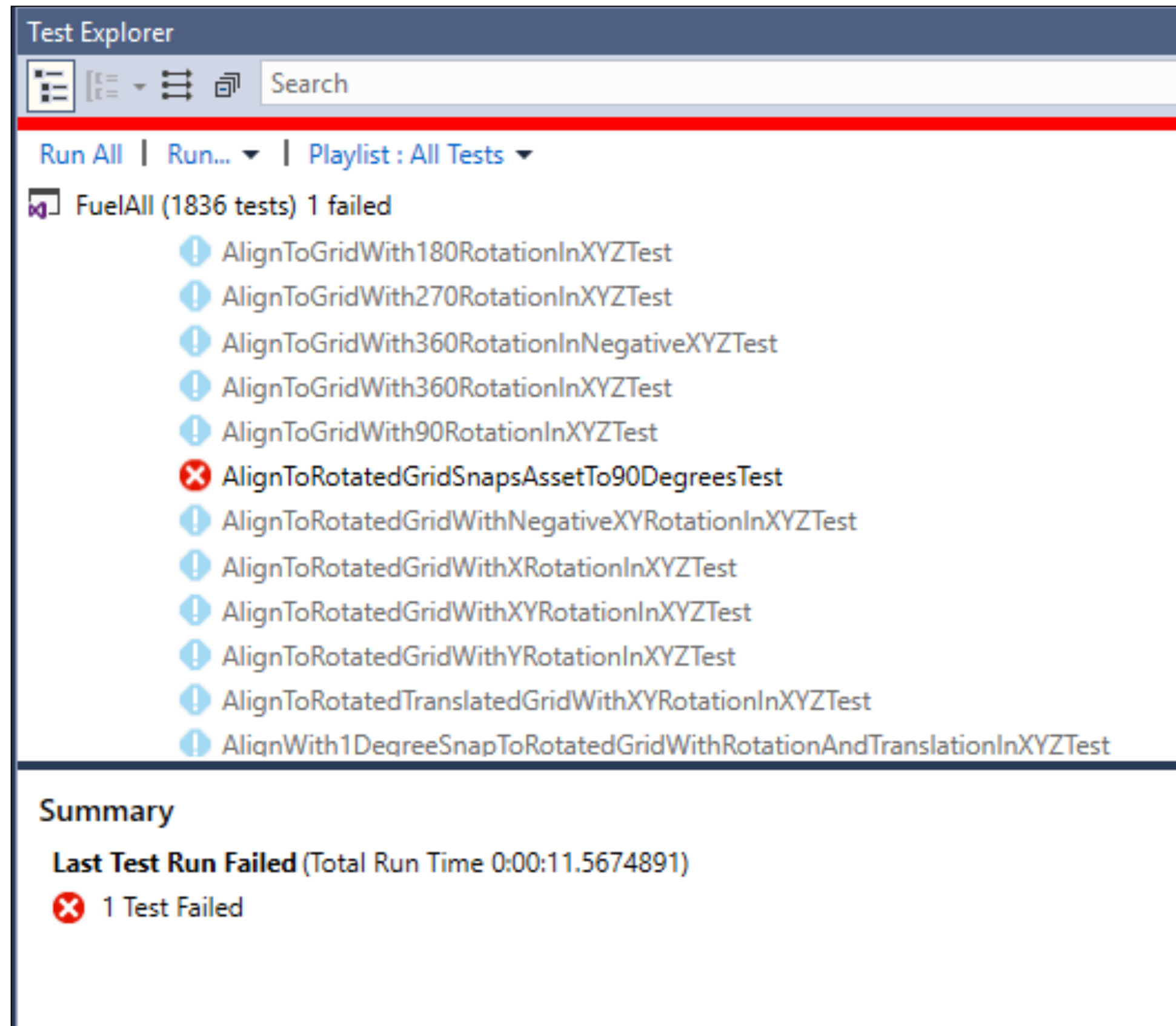


Step 4: Evaluate



Tests make *resolving bugs easier*

There is a bug when...

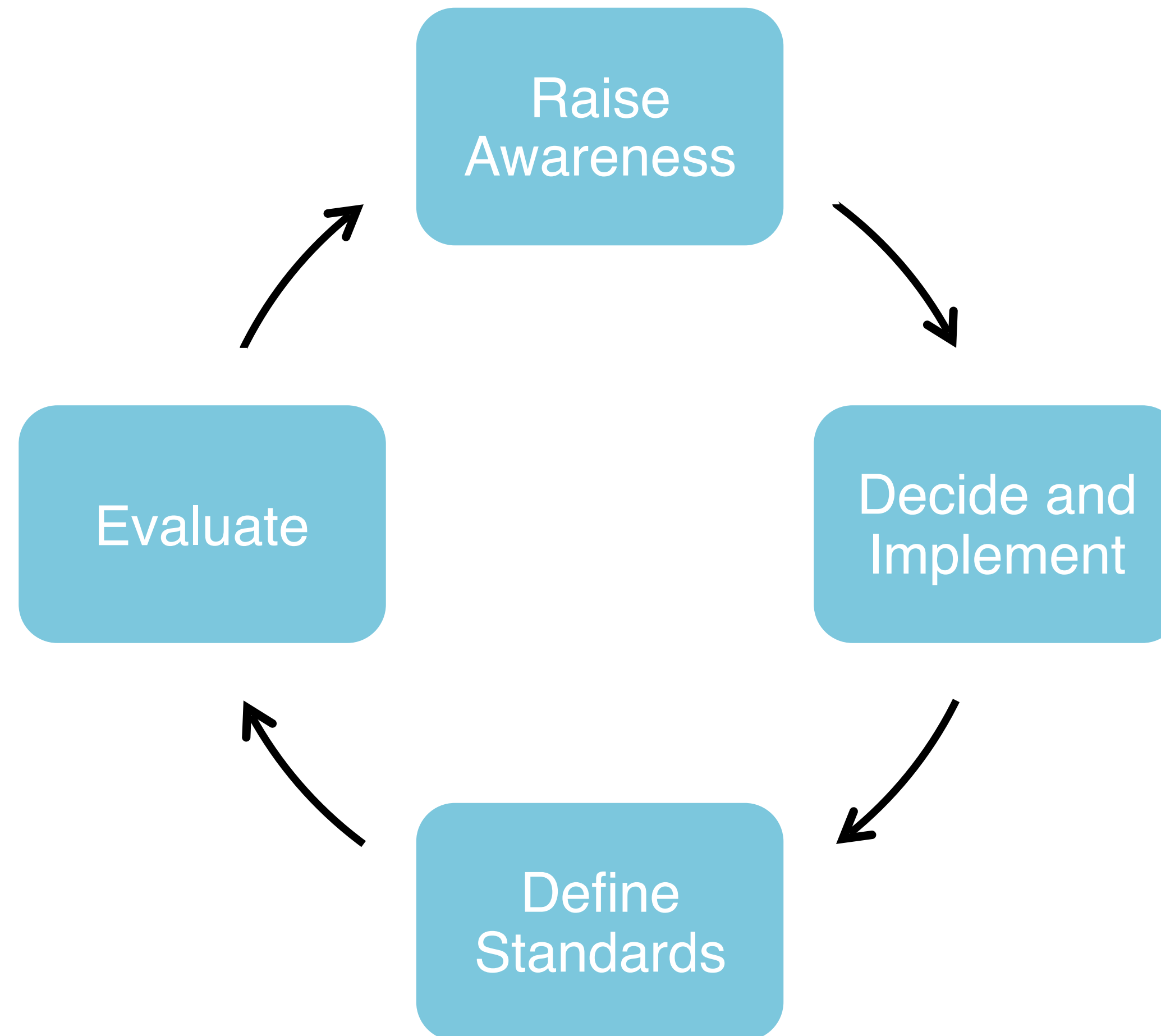


Step 4: Evaluate

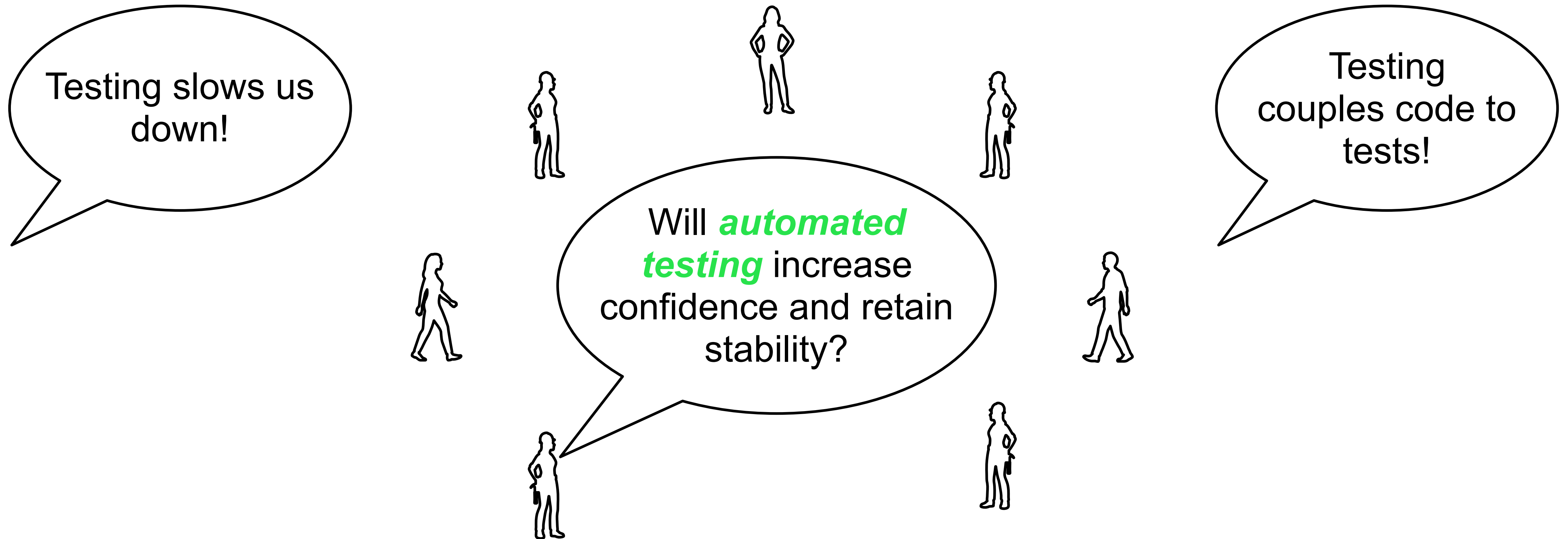


Tests *boost confidence* and *increase team trust*

Steps to Cultivate a Test Driven Culture

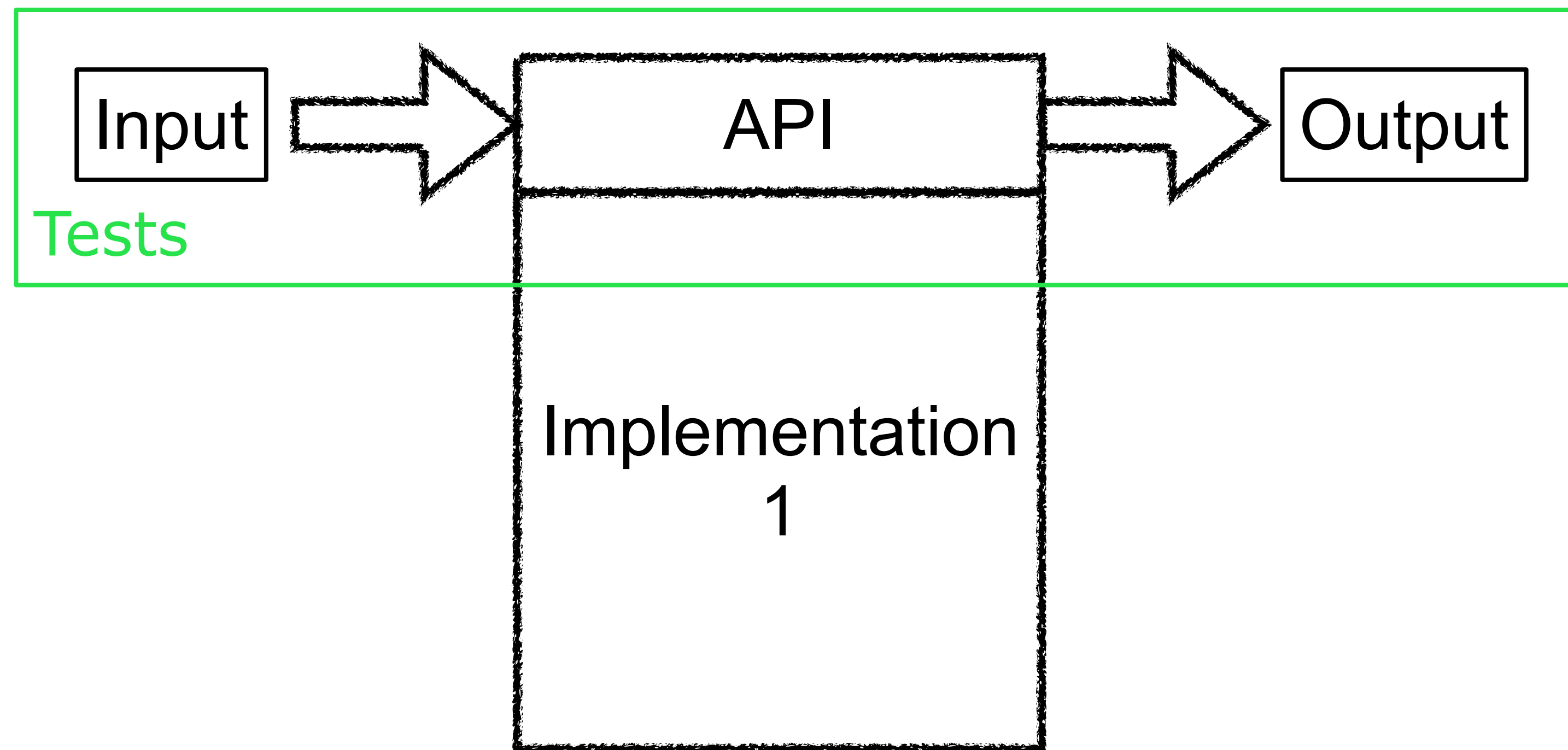


Step 1: Raise Awareness



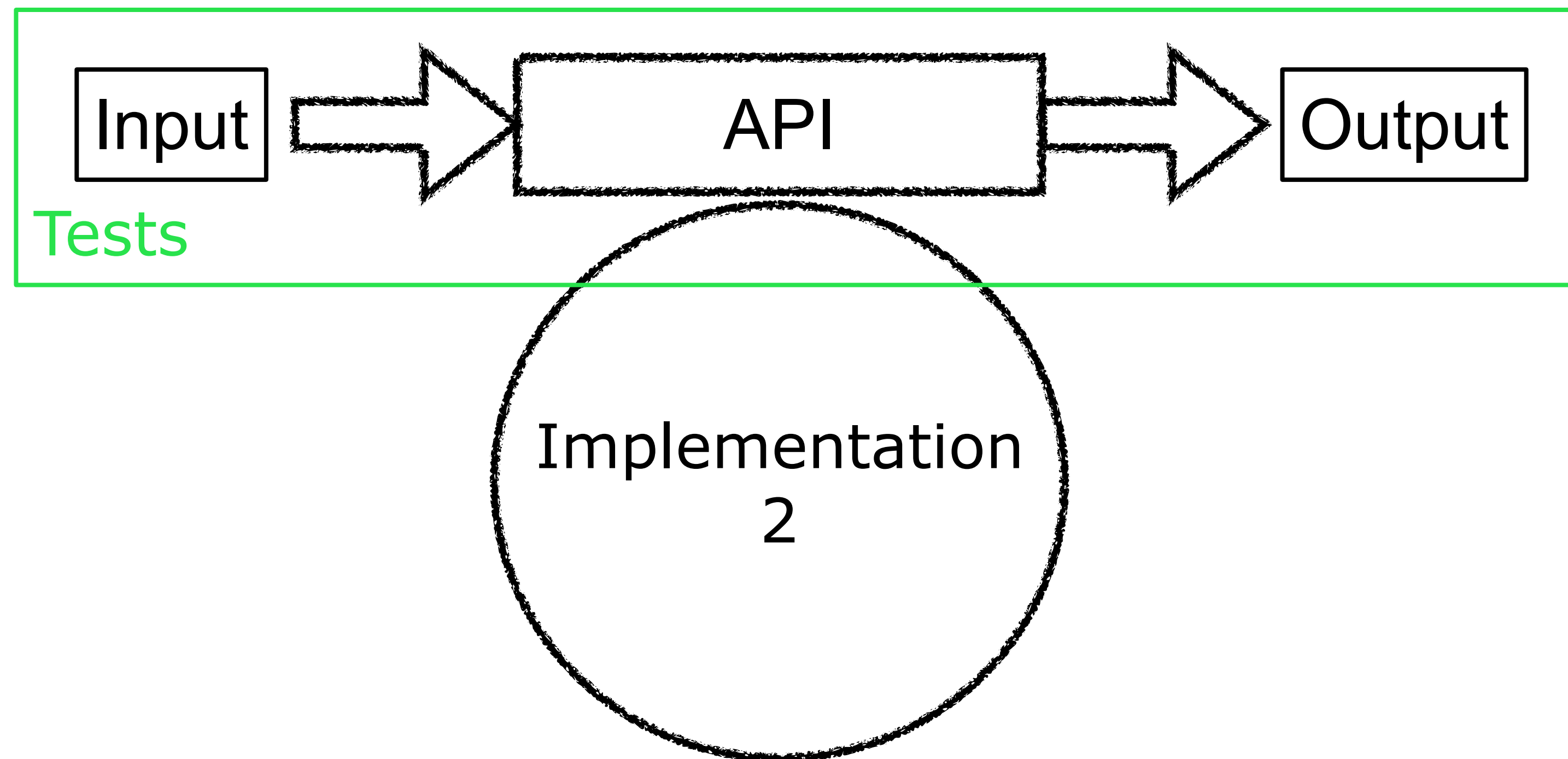
Step 1: Raise Awareness

Code written with *testing in mind is different* to code without.



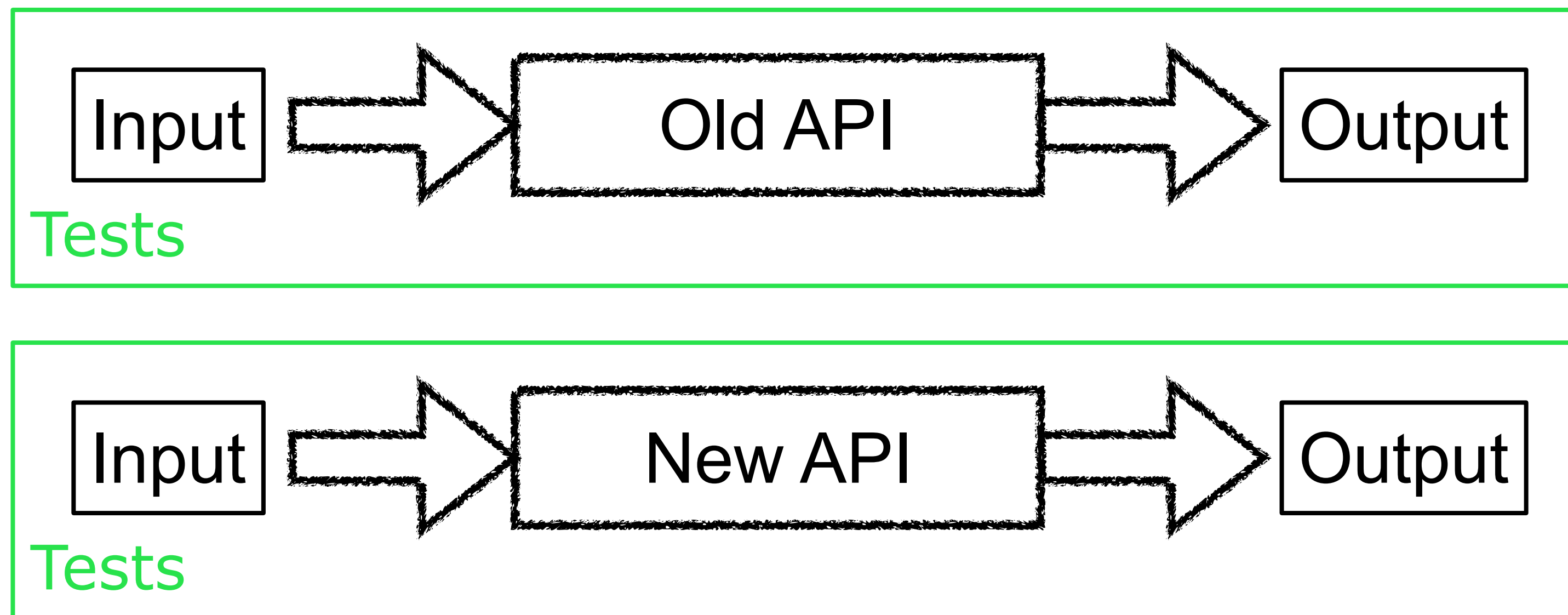
Step 1: Raise Awareness

Code written with *testing in mind is different* to code without.

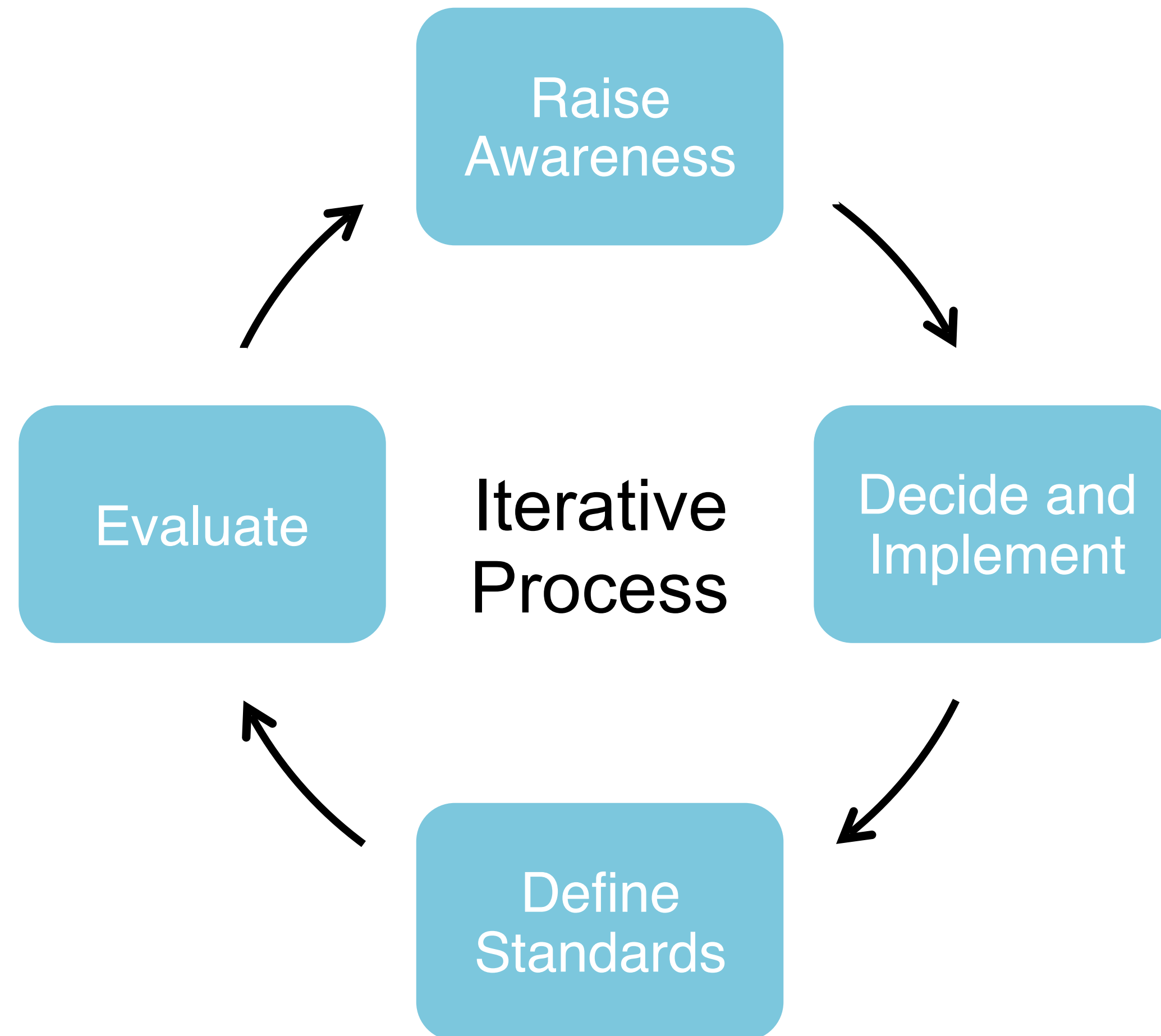


Step 1: Raise Awareness

Testing is a way to ***consistently ensure*** a workflow behaves as expected.



4 Simple Steps to Cultivate a Test Driven Culture



Avoiding Common Pitfalls

Tests are only *as good as the effort* put into them.

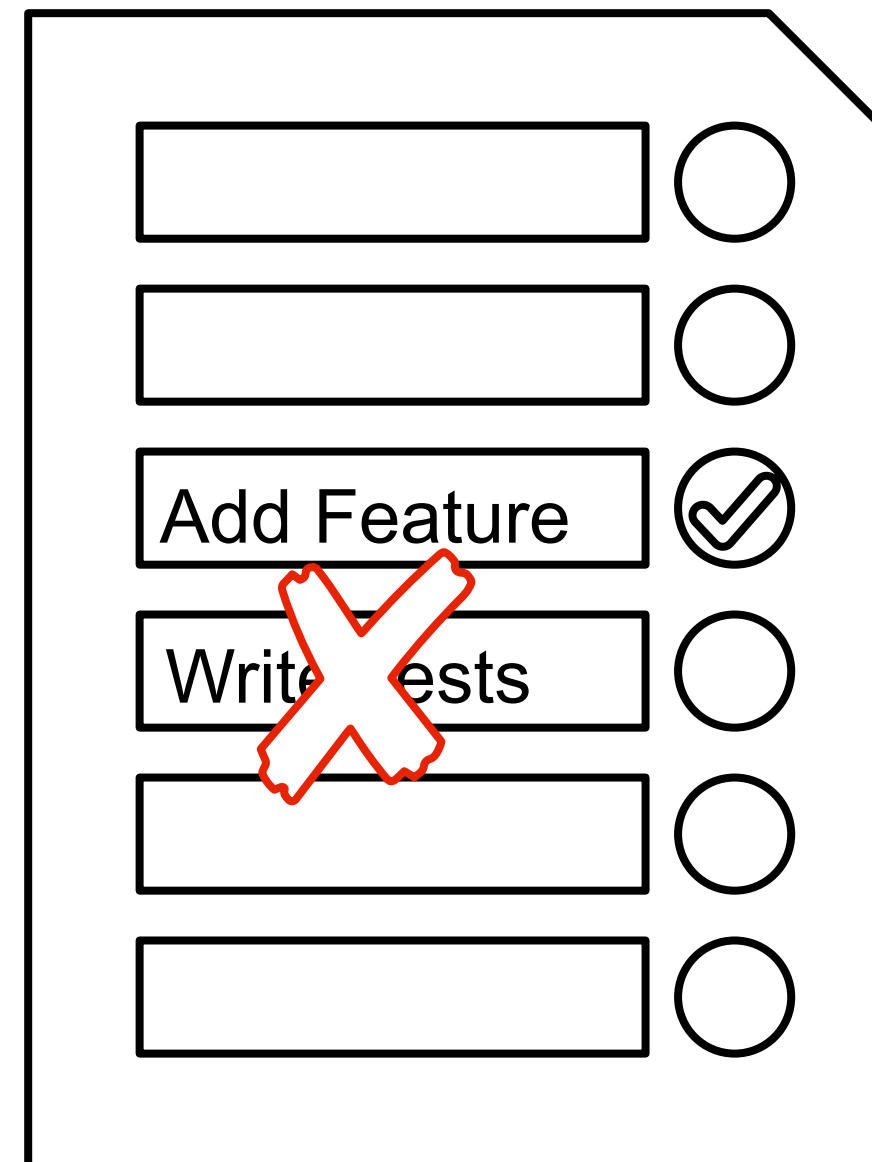
Tests don't improve quality: developers do.

Avoiding Common Pitfalls

Status ▾	Test ▾	Duration ▲
OK	BatchedHeightfieldBrushTests.SettingBrushRadiusCalculatesTheCorrectInnerBrushRadiusWithBrushSize100Test ▾ (VSTest: Turn10.Controls.Tests.Graphics.Techniques)	1ms 
OK	BatchedHeightfieldBrushTests.SettingBrushRadiusCalculatesTheCorrectOuterBrushRadiusWithBrushSize100Test ▾ (VSTest: Turn10.Controls.Tests.Graphics.Techniques)	1ms 
OK	BatchedHeightfieldBrushTests.SettingBrushRadiusCalculatesTheCorrectInnerBrushRadiusWithBrushSize200Test ▾ (VSTest: Turn10.Controls.Tests.Graphics.Techniques)	1ms 
OK	BatchedHeightfieldBrushTests.SettingBrushRadiusCalculatesTheCorrectOuterBrushRadiusWithBrushSize200Test ▾ (VSTest: Turn10.Controls.Tests.Graphics.Techniques)	1ms 
OK	CachedReflectionPropertyGetterTests.GetPropertyFromReturnsNullOnParentclassOfPropertyReflectedTypeTest ▾ (VSTest: Turn10.Controls.Tests.CollectionFilter)	1ms 
OK	ReflectionPropertyGetterTests.GetPropertyFromReturnsNullOnParentclassOfPropertyReflectedTypeTest ▾ (VSTest: Turn10.Controls.Tests.CollectionFilter)	1ms 
OK	AlignmentControlTests.OnValueTransformedDoesNotUpdateGridParentIfShouldUpdateGridTrueAndNotGridParentTransformedTest ▾ (VSTest: Turn10.Controls.Tests.AlignmentGrid)	1ms 
OK	AlignmentControlTests.OnValueTransformedDoesNotSendEngineRequestIfSnapStrategySetShouldUpdateGridTrueAndTransformContextUndoTest ▾ (VSTest: Turn10.Controls.Tests.AlignmentGrid)	1ms 
OK	AlignmentControlTests.OnValueTransformedDoesNotSendEngineRequestIfSnapStrategySetShouldUpdateGridTrueAndTransformContextSnapTest ▾ (VSTest: Turn10.Controls.Tests.AlignmentGrid)	1ms 
OK	AlignmentEngineTests.EngineDoesNotSetSettingIfPresentOnStartupTest ▾ (VSTest: Turn10.Controls.Tests.AlignmentGrid)	1ms 
OK	AlignmentEngineTests.EngineProcessesTwoStrategyTest ▾ (VSTest: Turn10.Controls.Tests.AlignmentGrid)	1ms 
OK	AlignmentEngineTests.EngineDoesNotUpdateTransformIfSameTest ▾ (VSTest: Turn10.Controls.Tests.AlignmentGrid)	1ms 
OK	AlignmentEngineTests.EngineProcessesTwoStrategyWithOneCanSnapFalseTest ▾ (VSTest: Turn10.Controls.Tests.AlignmentGrid)	1ms 
OK	LegoSettingsGridFactoryTest.FactoryDoesNotCreatesSettingsIfTheyExistTest ▾ (VSTest: Turn10.Controls.Tests.AlignmentGrid)	1ms 
OK	LegoSettingsGridFactoryTest.CreateFromTransformedTransformableCreatesGridAtTransformWithOffsetAndSpacingTest ▾ (VSTest: Turn10.Controls.Tests.AlignmentGrid)	1ms 
OK	SnapFilterTests.SnapFilterReturnsTransformableIfContextIsNotNullTest ▾ (VSTest: Turn10.Controls.Tests.AlignmentGrid)	1ms 
OK	SnapFilterTests.SnapFilterReturnsNullIfTransformIsNotNullTest ▾ (VSTest: Turn10.Controls.Tests.AlignmentGrid)	1ms 
OK	SnapFilterTests.SnapFilterReturnsNullIfTransformAndContextIsNotNullTest ▾ (VSTest: Turn10.Controls.Tests.AlignmentGrid)	1ms 

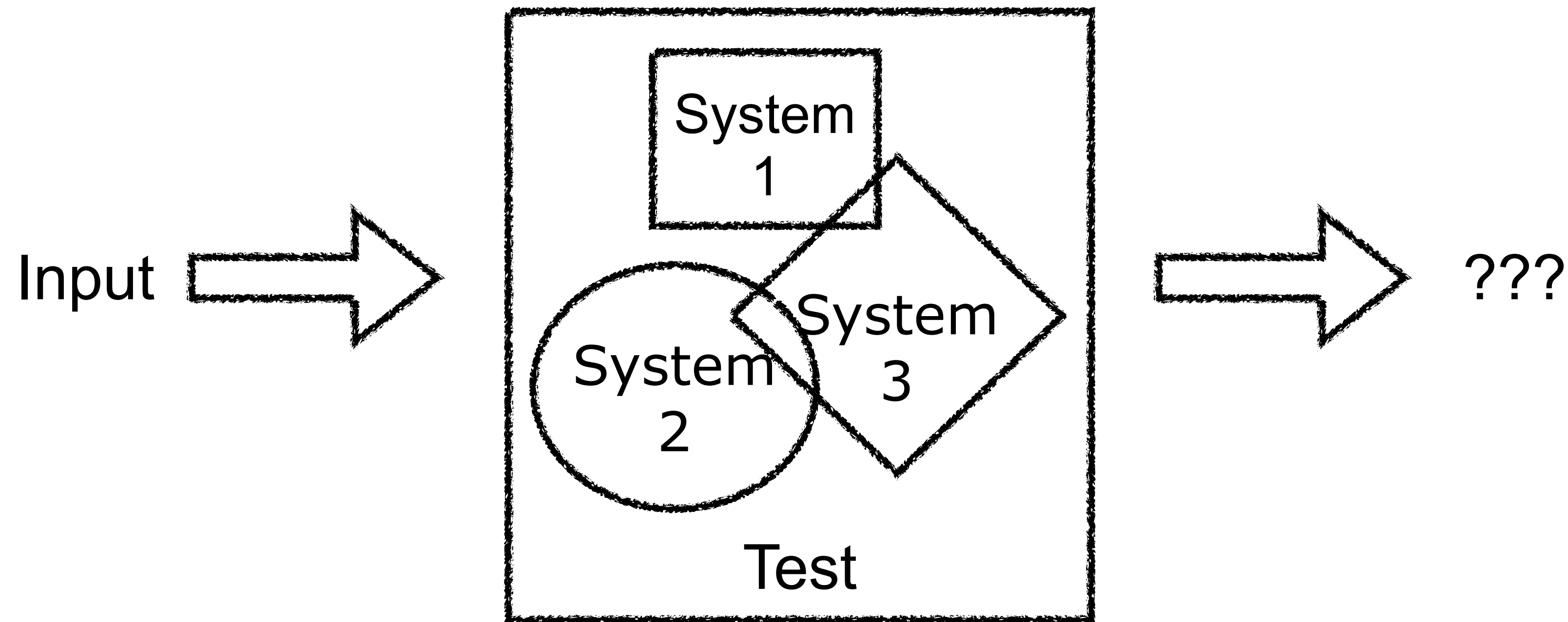
Write *fast* tests

Avoiding Common Pitfalls



Make testing ***part of your estimates***

Avoiding Common Pitfalls



Keep tests ***deterministic***. Consider flaky tests.

Avoiding Common Pitfalls

Learn from your mistakes.

Future Plans

Testing Working Group to ***expand to all disciplines.***

Automate manual tests for ***higher value QA.***

Use metrics (e.g. SonarQube) to ***make tests more impactful.***

Challenge

Find a problem you are passionate about.
Use the 4 steps to find a solution.
Introduce a positive change.

Key Takeaways

- Test driven approach has ***improved tool stability*** and ***increased confidence***:
 - More time for features
 - Reduced bugs
 - Improved collaboration
 - Easier to resolve bugs
 - Increased trust with other teams
- Changing a culture is a long term ***iterative process***:
 - Raise awareness, implement, standardise, evaluate
 - Learn from your mistakes

GDC

Speaker: David Paris

Questions/comments via Twitter: @davidjpparis



<https://www.playground-games.com/careers/>



GAME DEVELOPERS CONFERENCE
MARCH 16-20, 2020 | #GDC20