

# An Introduction to Unit Testing

Brian Henderson  
Programmer Analyst,  
Collaborative Data Services  
bhenders@fhcrc.org

## CDS Seminars & Training Classes

- ◆ CDS Brownbag seminars
  - Nov 28<sup>th</sup> - SharePoint Tips & Tricks  
SharePoint Tips & Tricks
  - Jason Mauer
  
  - Jan 10<sup>th</sup> - Creating Web Apps with Visual Studio 2005 (Repeat)
  - Paul Litwin
  
- ◆ Email List
  - announcement list for seminars/classes
  - contact Paul Litwin ([plitwin@fhcrc.org](mailto:plitwin@fhcrc.org)), or me to get on the list

2

*Intro to Unit Testing, Version 1.0 (10/03/05)*

©2005 CDS - Fred Hutch Cancer Research Center

. All Rights Reserved Worldwide. Unauthorized duplication is strictly prohibited.

## Genesis of Writing Tests

- ◆ "Why Test?"
  - we all have written code that worked just fine in the first place ...
- ◆ Can it break?
  - When? How? Where? What? Why?

*So if your code can't possibly be wrong, has a perfectly simple design, and everyone loves the interface, and if the code and **how to use it** is perfectly clear, then don't test it.*

3

## Agenda

- ◆ Test Techniques
- ◆ Practices
- ◆ Process
- ◆ Writing Unit Tests
- ◆ Tools & Resources

4

*Intro to Unit Testing, Version 1.0 (10/03/05)*

*©2005 CDS - Fred Hutch Cancer Research Center*

*. All Rights Reserved Worldwide. Unauthorized duplication is strictly prohibited.*

## Test Techniques

- ◆ What to Test
  - The Right-BICEP
- ◆ CORRECT Boundary Conditions
- ◆ Properties of Good Tests

5

## What to Test: Right-BICEP

- ◆ Are the Results **Right**?
- ◆ **B**oundary Conditions
- ◆ Check **I**nverse Relationships
- ◆ **C**ross-check using Other Means (Test Oracle)
- ◆ Force **E**rror Conditions (Attacks)
- ◆ **P**erformance Characteristics

6

*Intro to Unit Testing, Version 1.0 (10/03/05)*

©2005 CDS - Fred Hutch Cancer Research Center

. All Rights Reserved Worldwide. Unauthorized duplication is strictly prohibited.

## **CORRECT Boundary Conditions**

- ◆ **C**onformance
- ◆ **O**rdering
- ◆ **R**ange
- ◆ **R**eference
- ◆ **E**xistence
- ◆ **C**ardinality
- ◆ **T**ime
- ◆ **T**ry It Yourself

7

## **Properties of Good Tests**

- ◆ Automatic
- ◆ Thorough
- ◆ Repeatable
- ◆ Independent
- ◆ Professional
- ◆ ... & Testing the Tests

8

*Intro to Unit Testing, Version 1.0 (10/03/05)*

©2005 CDS - Fred Hutch Cancer Research Center

. All Rights Reserved Worldwide. Unauthorized duplication is strictly prohibited.

## Quotes – RE: Unit Tests

“The best way that I know to write code is to shape it from the beginning with tests” – Ron Jeffries

“Fewer defects, less debugging, more confidence, better design, and higher productivity in my programming practice” – Kent Beck

“... a powerful way to produce well designed code with fewer defects” – Martin Fowler

9

## Main Objective

- ◆ Objective: is not to test software
  - to aid programmers & customers with unambiguous requirements
  - provide feedback on stability of the design
- ◆ Benefits: of having Unit Tests
  - reconciling view points earlier (Pgmr/Cust)
  - starting point for QA (Quality Assurance)
  - code is written to be testable (motivation)
- ◆ Values
  - Communication, Simplicity, Feedback, Courage

10

*Intro to Unit Testing, Version 1.0 (10/03/05)*

*©2005 CDS - Fred Hutch Cancer Research Center*

*. All Rights Reserved Worldwide. Unauthorized duplication is strictly prohibited.*

## Example: Stack

- ◆ Specification:
  - A stack is a data structure in which you can access only the item at the top. Within a computer; a Stack is like a stack of dishes – you only add items to the top and remove them from the top.

11

## Example - Continued

- ◆ Specification in C#

```
[Test]
public void PushPop() {
    Stack stack = new Stack();
    string name = "Name";
    stack.Push(name);
    Assert.AreEqual(name, stack.Pop);
}
```

12

## Test Types

- ◆ Programmer Tests
  - aka Unit Tests
  - Technology facing tests
  - Test Tool: xUnit (JUnit, NUnit, etc.)
  
- ◆ Customer Tests
  - aka acceptance tests
  - Business facing tests
  - Test Tool: Fit (<http://fit.c2.com>)

13

## Simple Design

- ◆ "Simplicity is more complicated than you think. But it's well worth it."  
– Ron Jefferies
  
- ◆ Satisfy Requirements
  - No Less
  - No More

14

*Intro to Unit Testing, Version 1.0 (10/03/05)*

©2005 CDS - Fred Hutch Cancer Research Center

. All Rights Reserved Worldwide. Unauthorized duplication is strictly prohibited.

## Test-Driven Process

- ◆ Focused on Tasks (Stories/Senerios)
  - start by brainstorming a list of tests
  - don't get hung up on completeness...  
always can add more later
  - describes completion requirements
- ◆ Test List
- ◆ Red / Green / Refactor

15

## Demo: Test List

- ◆ Develop a programmer Test List for a Stack class that hold objects
- ◆ Methods:
  - void Push(object obj)
  - object Top()
  - bool IsEmpty()

16

*Intro to Unit Testing, Version 1.0 (10/03/05)*

©2005 CDS - Fred Hutch Cancer Research Center

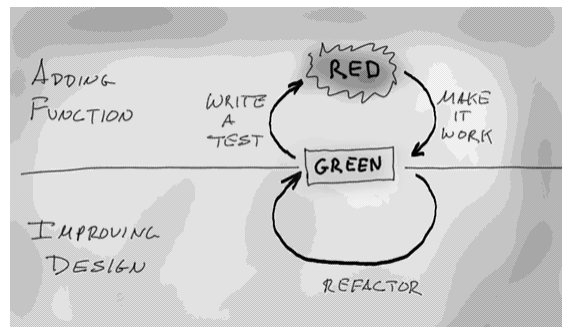
. All Rights Reserved Worldwide. Unauthorized duplication is strictly prohibited.

## Where to Start?

- ◆ The “simplest” test to get you started
  - e.g. calling `IsEmpty()` on a newly created **Stack**
- ◆ The essence of the problem
  - e.g. **Push** an item on a newly created **Stack**, Call **Pop** and verify that the pushed item is equal to the popped item.

17

## Red / Green / Refactor



18

*Intro to Unit Testing, Version 1.0 (10/03/05)*

©2005 CDS - Fred Hutch Cancer Research Center

. All Rights Reserved Worldwide. Unauthorized duplication is strictly prohibited.

## Fail / Pass / Rinse & Repeat

- ◆ Restated:
  - Make It Fail,
    - write no code without a failing automated test
  - Make It Work
    - as simply as possible
  - Make It Better
    - refactor
  - Repeat the cycle...

19

## Implications

- ◆ Developers can write their own test
- ◆ Design organically, running code provides the best feedback
- ◆ Development environment must be capable of providing rapid response to small changes
- ◆ Result – Highly cohesive and loosely coupled designs

20

*Intro to Unit Testing, Version 1.0 (10/03/05)*

©2005 CDS - Fred Hutch Cancer Research Center

*. All Rights Reserved Worldwide. Unauthorized duplication is strictly prohibited.*

## Demo – Stack Implementation

- ◆ see demo

21

## Other xUnit Frameworks

- ◆ original was SUnit for SmallTalk
- ◆ Ports exists for virtually every language
  - JUnit, CppUnit, DUnit, VbUnit, HtmlUnit, ..
  - List at: <http://www.xprogramming.com>
- ◆ Has become a defacto-standard for testing architecture
  - VS 2005 will have an integrated version

22

*Intro to Unit Testing, Version 1.0 (10/03/05)*

©2005 CDS - Fred Hutch Cancer Research Center

. All Rights Reserved Worldwide. Unauthorized duplication is strictly prohibited.

## Summary

- ◆ Test Techniques
- ◆ Practices
- ◆ Process
- ◆ Writing Unit Tests
- ◆ Tools & Resources

23

## Thank You

- ◆ Brian Henderson,
- ◆ Collaborative Data Services
  - bhenders@fhcrc.org
  - x3256
- ◆ Slides and utilities are available
  - You download from  
<http://cdsweb.fhcrc.org/events.aspx>

24

*Intro to Unit Testing, Version 1.0 (10/03/05)*

©2005 CDS - Fred Hutch Cancer Research Center

. All Rights Reserved Worldwide. Unauthorized duplication is strictly prohibited.

## References & Resources

- ◆ Books – Testing in General
  - ◆ How to Break Software  
James A Whittaker
  - ◆ Lessons Learned in Software Testing  
Cem Kaner, James Bach, Bret Petichord
  - ◆ Testing Computer Software, 2ed  
Cem Kaner Jack Falk, Hung Quoc Nguyen
  - ◆ Testing Object-Oriented Systems  
Robert V. Binder
  - ◆ The Practice of Programming  
Kernighan, et. al.

25

## References & Resources (+)

- ◆ Books – Unit Testing Specific
  - ◆ Test-Driven Development –  
A Practical Guide  
David Astels
  - ◆ Test-Driven Development -  
By Example  
Kent Beck
  - ◆ Pragmatic Unit Testing  
with C# / with Java  
Andrew Hunt, David Thomas
  - ◆ Agile Software Development  
Patterns & Practices  
Robert Martin
  - ◆ JUnit (O'Reilly Pocket Guide)  
Kent Beck
  - ◆ NUnit (O'Reilly Pocket Guide)  
Bill Hamilton

26

*Intro to Unit Testing, Version 1.0 (10/03/05)*

©2005 CDS - Fred Hutch Cancer Research Center

. All Rights Reserved Worldwide. Unauthorized duplication is strictly prohibited.

## References (cont.)

The Coad Letter's: Test-driven Development  
Dave Astels  
[dave@adaptionsoft.com](mailto:dave@adaptionsoft.com)

Ron Jeffries.  
XProgramming.com an extreme programming  
resource.  
[www.xprogramming.com](http://www.xprogramming.com)

NUnit - Download  
[www.nunit.org](http://www.nunit.org)

JUnit –  
<http://junit.sourceforge.net/>

Cookbook –  
<http://junit.sourceforge.net/doc/cookbook/cookbook.htm> 27