

DC ALT.NET ~ February 2010



A "Lingua Franca" to Ensure You Get the Right System

George Dinwiddie

Software Development Coach & Consultant

IDIA Computing, LLC

<http://idiacomputing.com/>



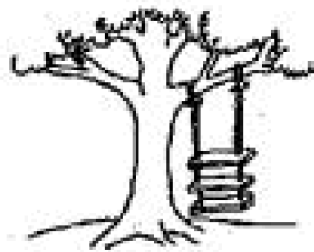
It Begins with an Idea



Photo by Peter Jackson



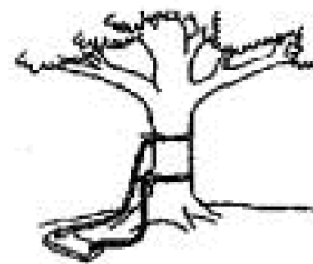
The Product



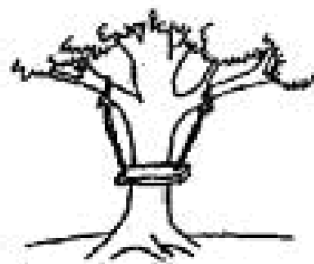
**As proposed
by the project
sponsor.**



**As specified
in the project
request.**



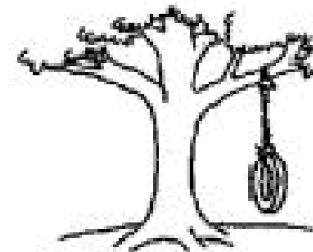
**As designed
by the senior
architect.**



**As produced
by the
engineers.**



**As installed at
the user's
site.**



**What the
customer
really wanted.**



Discussion



Photograph by Lars Plougmann



We want to encourage avid readers to order more books. So we're going to offer every 4th book ordered in a month for free.

Discussion

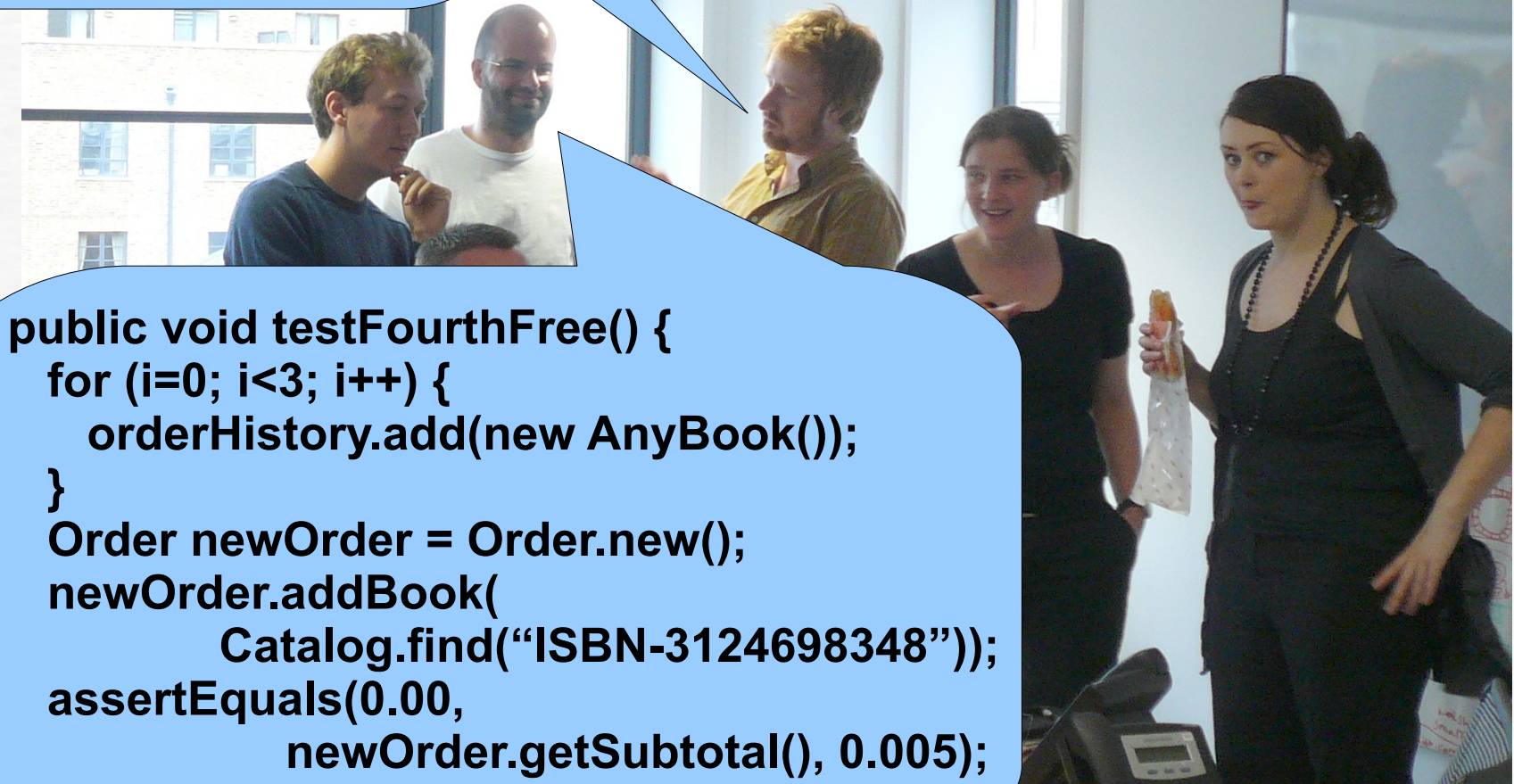


Photograph by Lars Plougmann



We want to encourage avid readers to order more books. So we're going to offer every 4th book ordered in a month for free.

Discussion



```
public void testFourthFree() {  
    for (i=0; i<3; i++) {  
        orderHistory.add(new AnyBook());  
    }  
    Order newOrder = Order.new();  
    newOrder.addBook(  
        Catalog.find("ISBN-3124698348"));  
    assertEquals(0.00,  
        newOrder.getSubtotal(), 0.005);  
}
```

Photograph by Lars Plougmann



Discussion

We want to encourage avid readers to order more books. So we're going to offer every 4th book ordered in a month for free.

Login as "bart_bookworm"
Go to "/catalog/search"
Enter "ISBN-3124698348"
Click "Search"
Click "Add to Cart"
Click "View Cart"
Verify "Subtotal" is "\$0.00"

```
public void testFourthFree() {  
    for (i=0; i<3; i++) {  
        orderHistory.add(new AnyBook());  
    }  
    Order newOrder = Order.new();  
    newOrder.addBook(  
        Catalog.find("ISBN-3124698348"));  
    assertEquals(0.00,  
        newOrder.getSubtotal(), 0.005);  
}
```

Photograph by Lars Plougmann



**Given a user has ordered 3
books previously in the month
When the user puts a new
book in the shopping cart
Then the subtotal should be
zero.**

cussion



Photograph by Lars Plougmann



No, that's not quite it. The *cheapest* of the 4 books is free, not necessarily the *last* one.

Discussion



Photograph by Lars Plougmann



Testimonial



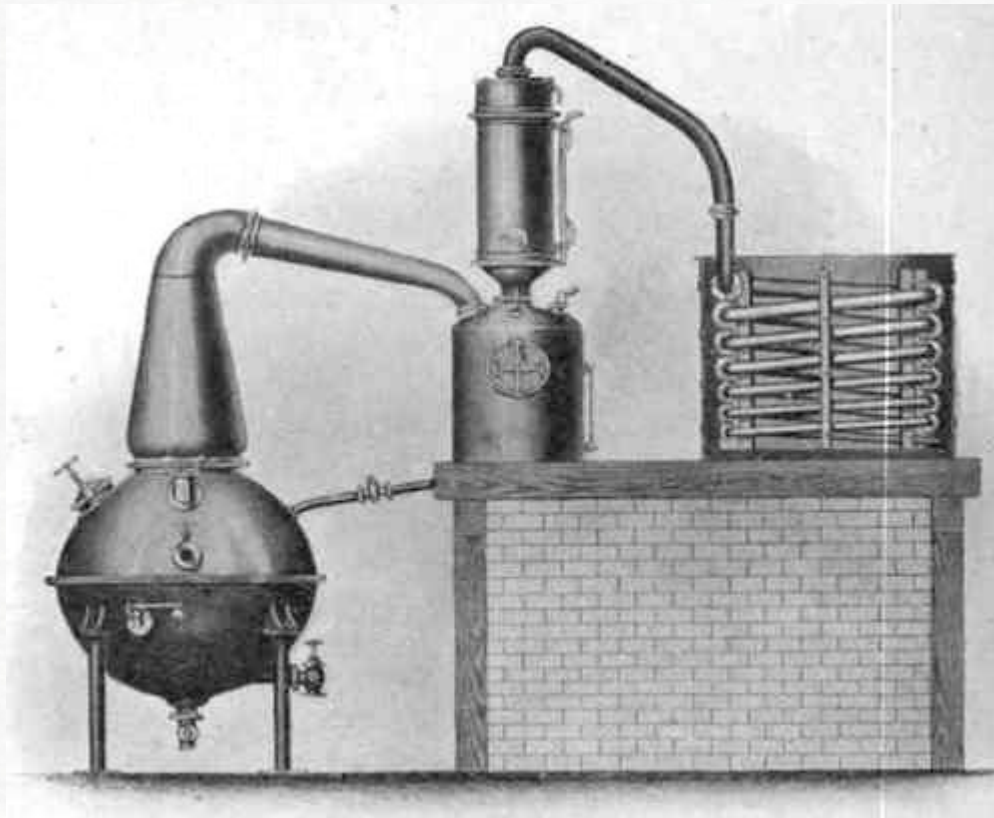
“Last night, I introduced my 76-year-old mother to her first Cucumber feature. As the domain expert, she was able to discuss the issue with me without her eyes glazing over even once. I think that's because we were able to discuss the issue in her language, not my technobabble. Even though I started writing the story myself, she quickly was able to add to it.”

– Cire

http://groups.google.com/group/cukes/browse_thread/thread/709eab29a4de5b8b#msg_407c961bea961bdc



Distillation



Given – When – Then



Given some initial context (the givens),

When an event occurs,

Then ensure some outcomes.

From Dan North & Chris Matts
See <http://dannorth.net/introducing-bdd>



Incidental Details & Implicit Details

Given I enter "bart_bookworm" into "name" field
and I enter "SuPeRsEkRiT" into "password" field
and I press "Login" button
and I search for "Star Wars" within Books
and the price is \$3.95
When I click "Add to Cart"
and I click "View Cart"
Then I should see "Subtotal: \$0.00"



Distill the Essence



Given I've previously ordered, this month:

title	price
"Hitchhiker's Guide"	\$11.95
"Neutron Star"	\$4.50
"Across Realtime"	\$6.50

When I select "Star Wars" for \$3.95

Then my shopping cart subtotal should be \$0.00



Development



Photo by George Dirwiddie



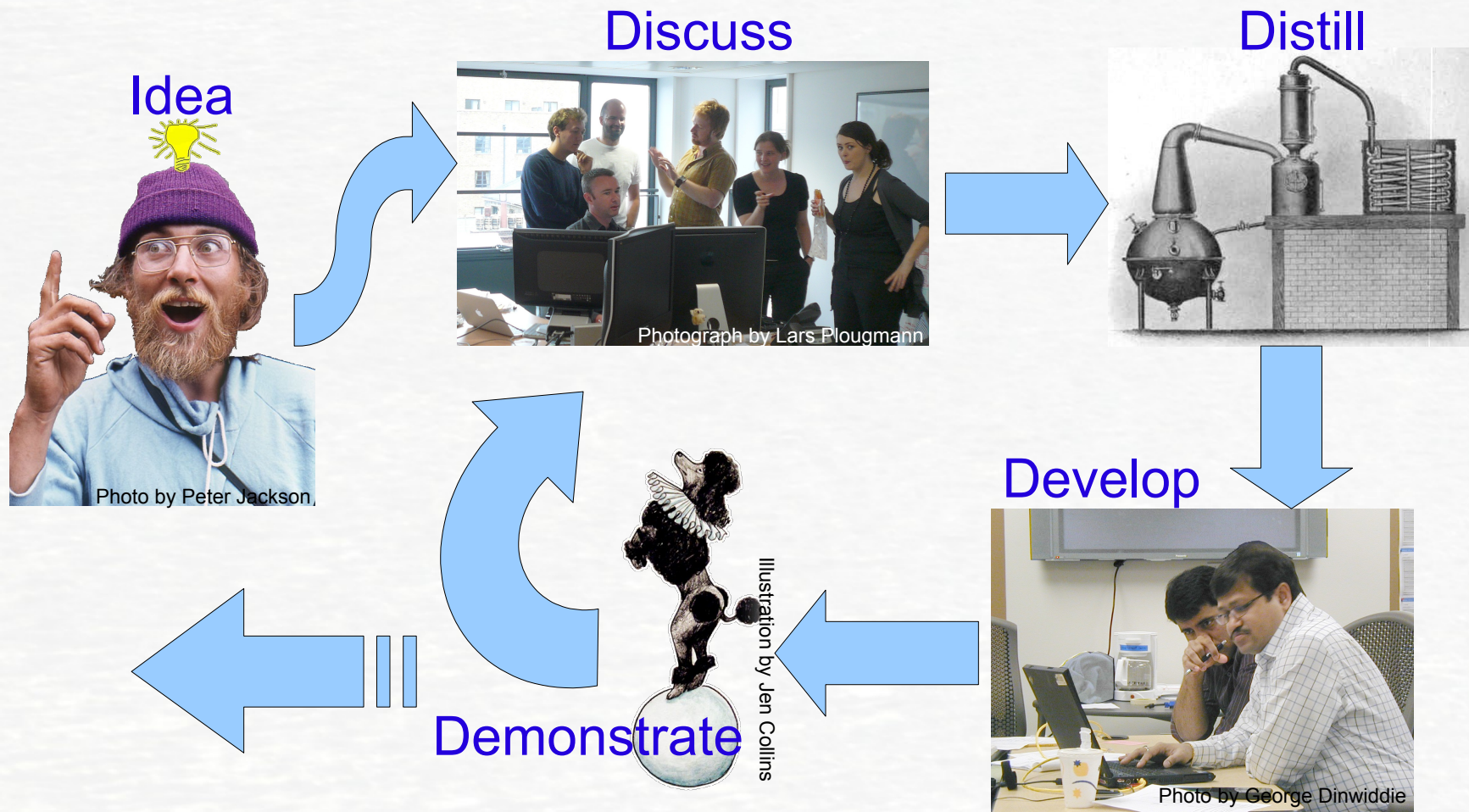
Demonstration



Illustration by Jen Collins



The Acceptance-Test-Driven or Story-Test-Driven Cycle



Why do we want automated acceptance tests?



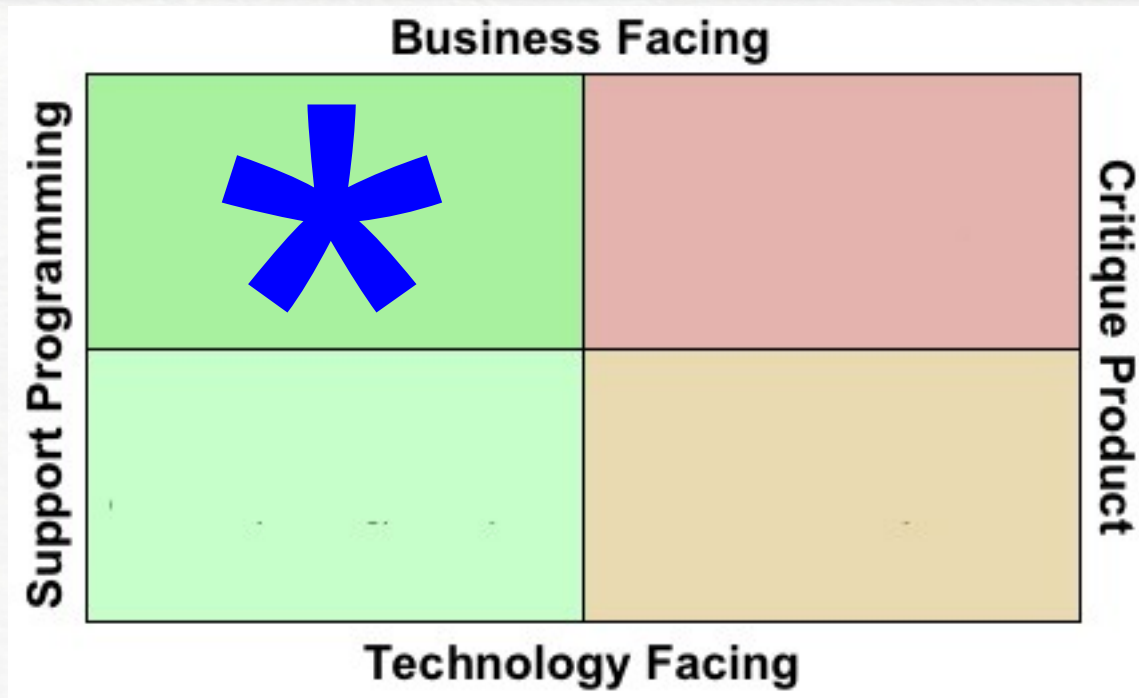
- To express functional expectations such that Developers, Testers, and Analysts all can read and understand and agree.
- To define “DONE” for the story.
- To avoid unpleasant surprises in older features.
- To *know* the functionality *really* works with out a separate “testing phase.”

and, most importantly,

- To **prevent** defects rather than find them.



Categorization of Tests



- “A business-facing test is one you could describe to a business expert in terms that would (or should) interest her”
- “By "support programming", I mean that the programmers use them as an integral part of the act of programming.”

Brian Marick, <http://www.exampler.com/old-blog/2003/08/21/#agile-testing-project-1>



Arrange – Act – Assert



```
[Test]
public void
RotateAngle_Given27Degress_Returns64Degrees()
{
    //Arrange or Given
    var someAngleClass = new Angle();

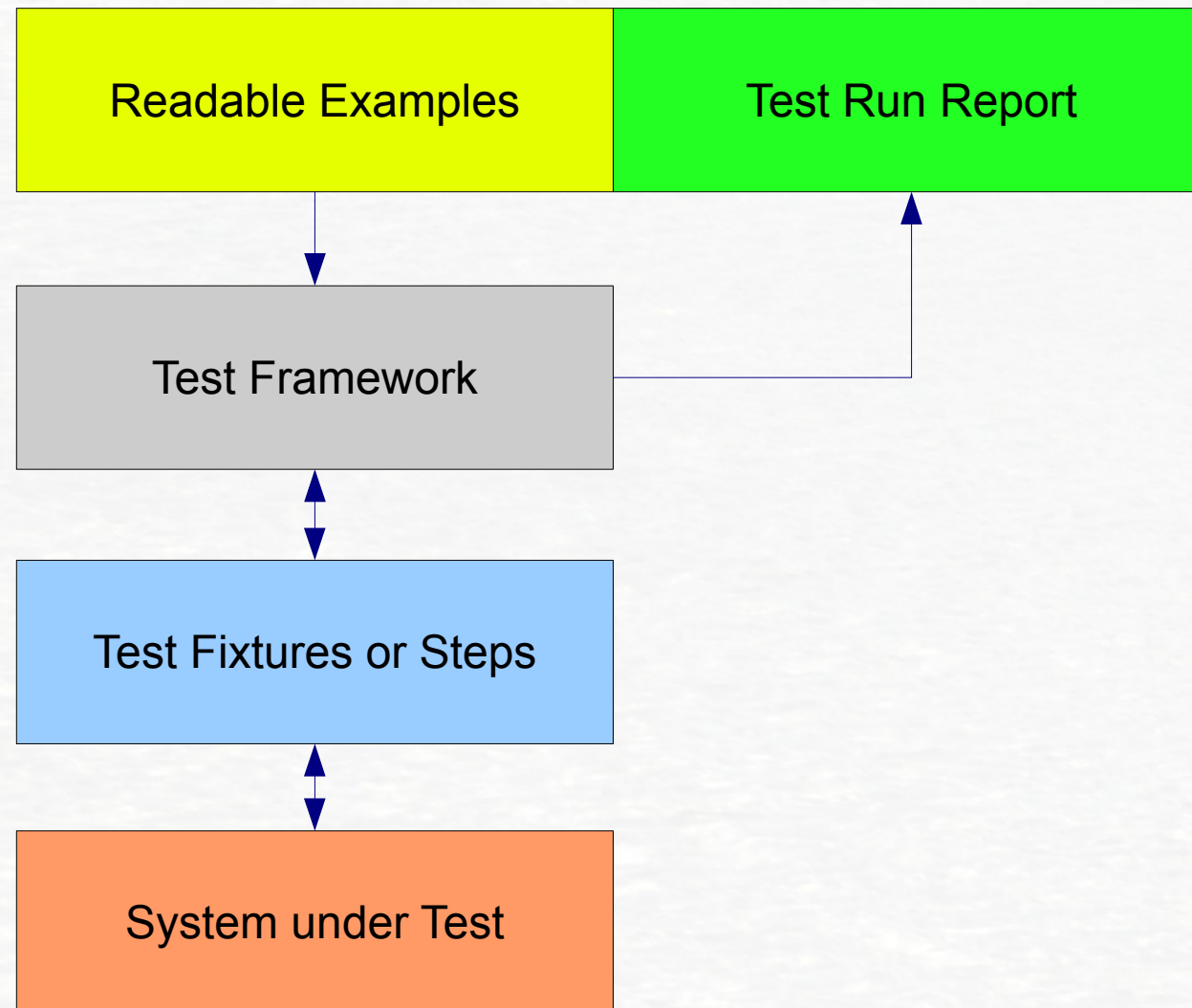
    //Act or When
    var result = someAngleClass.Rotate(27);

    //Assert or Then
    Assert.That(result, Is.EqualTo(64));
}
```

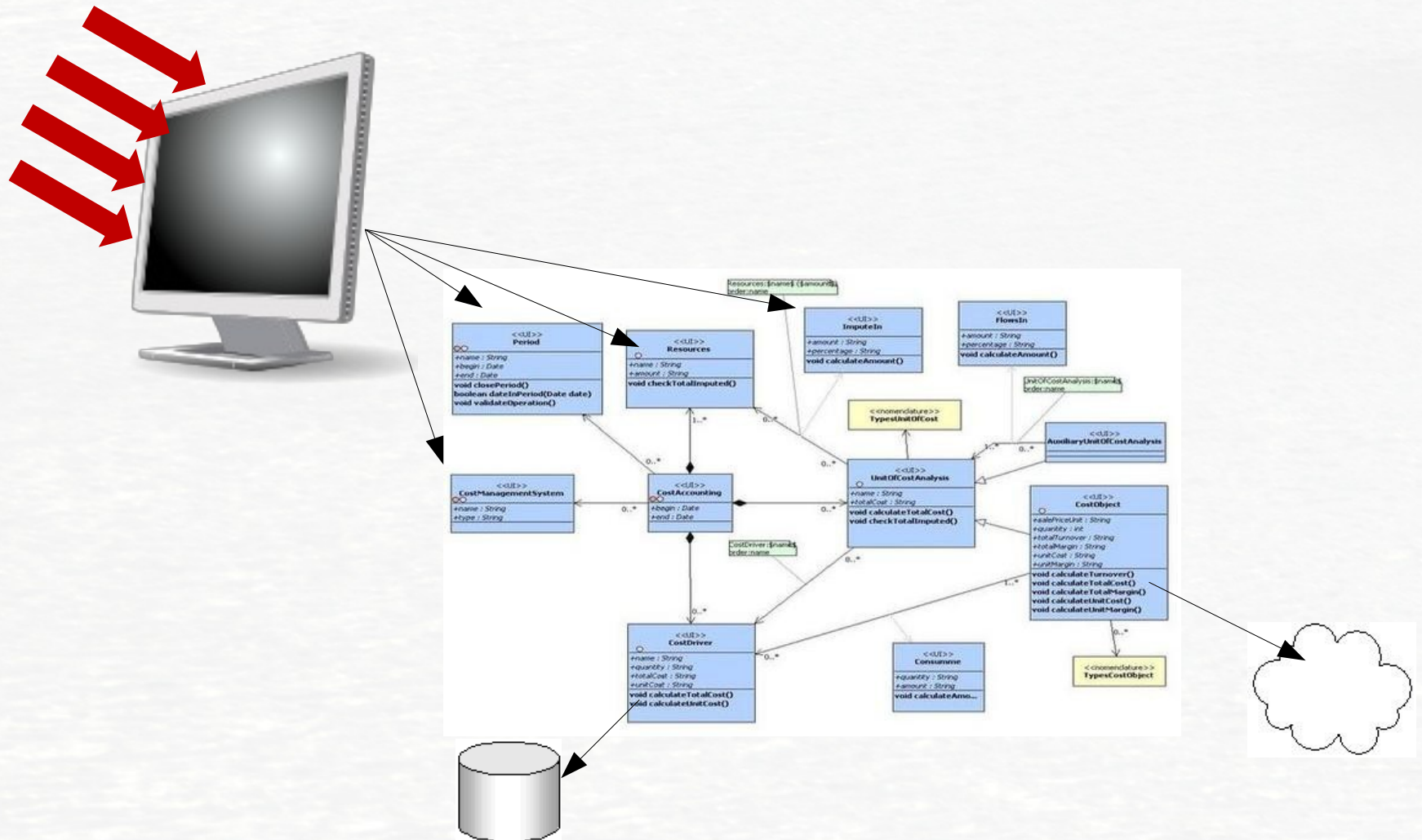


<http://stackoverflow.com/questions/1643933/given-when-then-example-with-nunit>

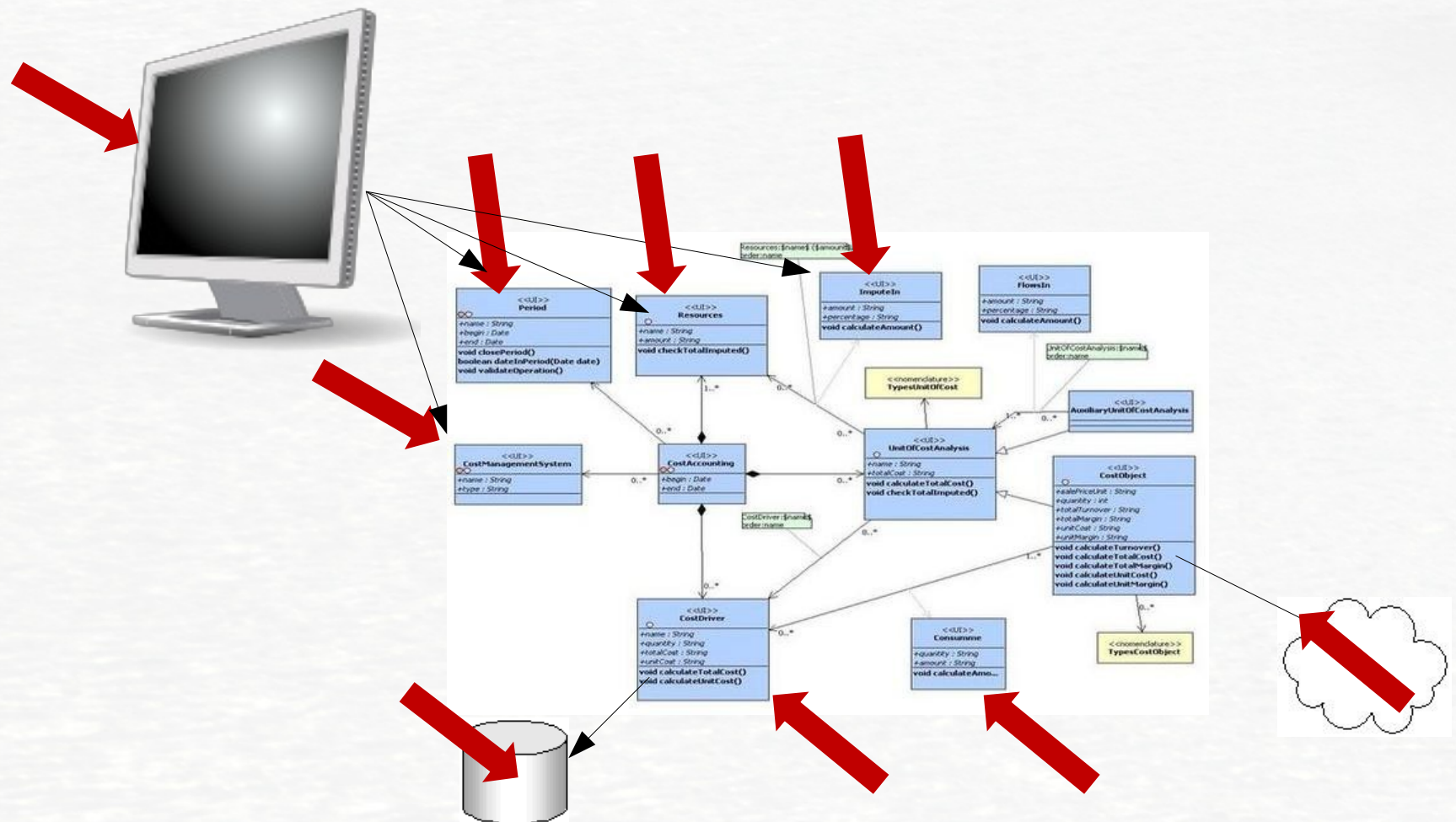
Typical Acceptance Test Architecture



UI-Centric Testing



Fixtures can access anywhere



Cucumber example



Feature: Fourth book is free

As Bart Bookworm, I get a free BookClub book for every three BookClub selections I buy in a month. Cheapest book wins.

Scenario: 4 books during a single shopping experience

Given I am Bart Bookworm

And I select “Hitchhiker’s Guide” for \$11.95 from the BookClub

And I select “Star Wars” for \$3.95 from the BookClub

And I select “Neutron Star” for \$4.50 from the BookClub

And I select “Across Realtime” for \$6.50 from the BookClub

When I view my shopping cart

Then the discount for “Star Wars” is \$3.95

But the discount for “Neutron Star” is \$0.00

See <http://cukes.info/>



Cucumber tables in a scenario



Feature: Fourth book is free

As Bart Bookworm, I get a free BookClub book for every three BookClub selections I buy in a month. Cheapest book wins.

Scenario: 4 books during a single shopping experience

Given I am Bart Bookworm

And I select the following books from the BookClub:

title	price
"Hitchhiker's Guide"	\$11.95
"Star Wars"	\$3.95
"Neutron Star"	\$4.50
"Across Realtime"	\$6.50

When I view my shopping cart

Then the discount for "Star Wars" is \$3.95

But the discount for "Neutron Star" is \$0.00



Cucumber tables for multiple examples



Feature: Shipping Calculation

As Bart Bookworm, I can see the shipping cost as I shop.

Scenario Outline: Shipping charge is updated based on book weight

Given I am Bart Bookworm

And I select <book> which weighs <pounds> pounds <ounces> ounces

When I view my shopping cart

Then the shipping charge is <postage>

Examples:

book	pounds	ounces	postage	
"Poky Little Puppy"	0	2	\$3.95	
"Encyclopedia Britannica"	62	5	\$28.95	
"War and Peace"	1	8	\$4.95	



Cucumber



- Written in Ruby
- Supports any JVM language (Java, Groovy, Scala, etc.) by running under Jruby
- Step definitions in these languages using Cuke4Duke
- Supports any .Net language using Cuke4Nuke

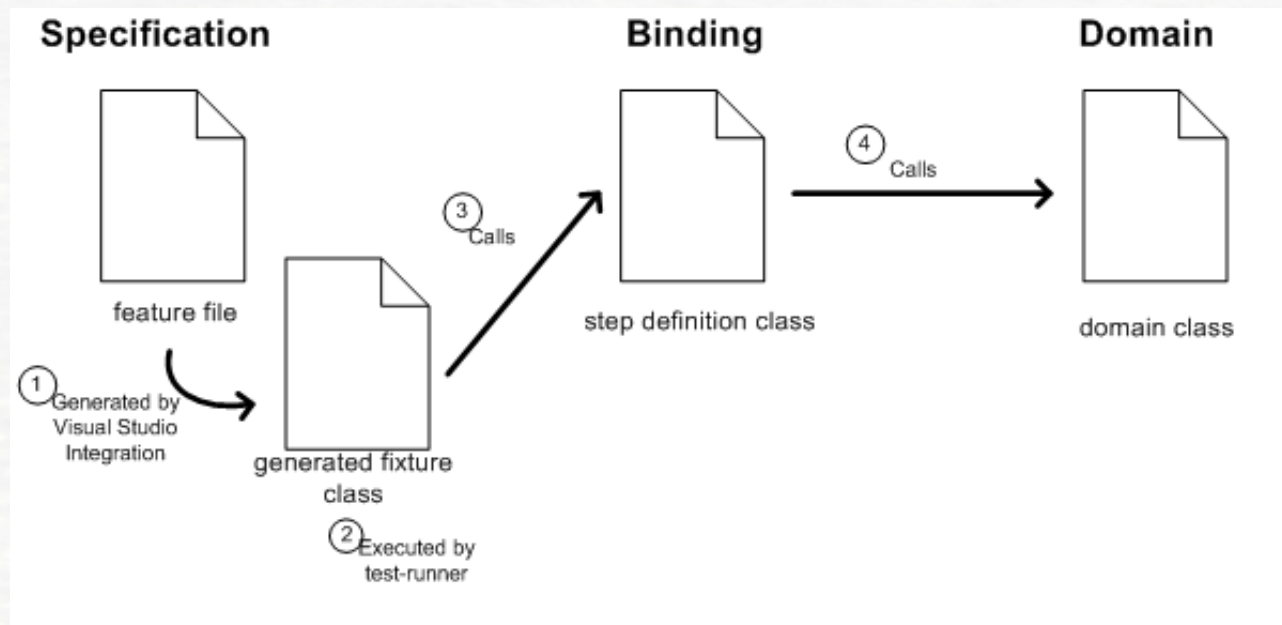


SpecFlow



- Supports gherkin language (same as Cucumber)
- Step definitions in any .Net language

<http://specflow.org/>



JBehave example



Trader is alerted of status

Scenario:

In order to ensure a quick response

As a trader

I want to monitor stock prices

Given a stock of symbol STK1 and a threshold of 15.0

When the stock is traded at price 5.0

Then the alert status is OFF

When the stock is sold at price 11.0

Then the alert status is OFF

When the stock is sold at price 16.0

Then the alert status is ON

<http://jbehave.org/>



JBehave tables for multiple examples



Scenario:

In order to ensure a quick response

As a trader

I want to monitor stock prices

Given a stock of <symbol> and a <threshold>

When the stock is traded with <price>

Then the trader is alerted with <status>

Examples:

symbol	threshold	price	status
--------	-----------	-------	--------

STK1	15.0	5.0	OFF
------	------	-----	-----

STK1	15.0	11.0	OFF
------	------	------	-----

STK1	15.0	16.0	ON
------	------	------	----



easyb



- Written in Groovy
- Runs under JVM
- A bit more “code like”
- Two modes of expression
 - Stories (similar to Cucumber)
 - Specifications (similar to rspec)

<http://www.easyb.org/>



easyb story example



```
package org.easyb.bdd.money

description "This story is about currency management"

narrative "this string is required for now", {
  as a "person who uses money"
  i want "to be able to add them together"
  so that "that I can become rich"
}

scenario "two moneys of the same currency are added", {
  given "one money object is added to another", {
    total = new Money(12, "CHF").add(new Money(14, "CHF"))
  }
  then "the total amount should be the sum of the two", {
    total.amount().shouldBe 26
  }
}
```



easyb specification example

```
package org.easyb.bdd.specification.queue
import org.easyb.bdd.Queue
```

```
description "This is how a Queue must work"
before "initialize the queue for each spec", {
  queue = new Queue()
}
```

```
it "should dequeue item just enqueued", {
  queue.enqueue(2)
  queue.dequeue().shouldBe(2)
}
```

```
it "should dequeue items in same order enqueued", {
  [1..5].each {val ->
    queue.enqueue(val)
  }
  [1..5].each {val ->
    queue.dequeue().shouldBe(val)
  }
}
```



GivWenZen example



```
|script|  
| given | a ToDo item is due tomorrow |  
| when | the date changes to tomorrow |  
| then | a notification exists indicating the ToDo is due |
```

A plugin for FitNesse
<http://code.google.com/p/givwenzen/>



GivWenZen output





Test

Edit

Properties

Refactor

Where Used

Search

Files

Versions

Recent Changes

User Guide

Test History

PaymentHandler, RefillSuite

CompleteRefillTest

Assertions: 5 right, 0 wrong, 0 ignored, 0 exceptions

► Set Up: [PaymentHandler.Setup \(edit\)](#)

[bdd.steps.CompletePurchase](#)

script	
given	a purchase is in INITIALIZED state
and	payment succeeds
and	delivery succeeds
when	the client completes the purchase
then	a purchase receipt should be shown

script		
given	a purchase is in INITIALIZED state	
and	payment fails	
when	the client completes the purchase	
then	purchase should fail with error IPL_PAYMENT_FAILED	
show	then	what the error is
		IPL_PAYMENT_FAILED

See <http://ketiljensen.wordpress.com/2009/12/13/bdd-with-scenario-tables-in-fitness-slim/>



Morelia Viridis (Python)



Feature: Addition

As a math idiot, I'd like a program that can add numbers for me.

Scenario: Add two numbers

Given I have entered 50 into the calculator

And I have entered 70 into the calculator

When I press add

Then the result should be 120 on the screen

<http://c2.com/cgi/wiki?MoreliaViridis>



Morelia Viridis Tables



Scenario: orders above \$100.00 to the continental US get free ground shipping
When we send an order totaling \$<total>, with a 12345 SKU, to our warehouse
And the order will ship to <destination>
Then the ground shipping cost is \$<cost>
And <rapid> delivery might be available

total	destination	cost	rapid
98.00	Rhode Island	8.25	yes
101.00	Rhode Island	0.00	yes
99.00	Kansas	8.25	yes
101.00	Kansas	0.00	yes
99.00	Hawaii	8.25	yes
101.00	Hawaii	8.25	yes
101.00	Alaska	8.25	yes
99.00	Ontario, Canada	40.00	no
99.00	Brisbane, Australia	55.00	no
99.00	London, United Kingdom	55.00	no
99.00	Kuantan, Malaysia	55.00	no
101.00	Tierra del Fuego	55.00	no



StoryQ example



```
using StoryQ;
using Microsoft.VisualStudio.TestTools.UnitTesting;
namespace StoryQ.Demo
{
    [TestClass]
    public class DemoTest
    {
        [TestMethod]
        public void PassingExample()
        {
            new Story("Data Safety")
                .InOrderTo("Keep my data safe")
                .AsA("User")
                .IWant("All credit card numbers to be encrypted")
                .WithScenario("submitting shopping cart")
                .Given(IHaveTypedMyCreditCardNumberIntoTheCheckoutPage)
                .When(IClickThe_Button, "Buy")
                .And(TheBrowserPostsMyCreditCardNumberOverTheInternet)
                .Then(TheForm_BePostedOverHttps, true)

            .ExecuteWithSimpleReport(MethodBase.GetCurrentMethod());
        }
    }
}
```



StoryQ output



```
DemoTest.PassingExample : Passed
Story is Data Safety
  In order to Keep my data safe
  As a User
  I want All credit card numbers to be encrypted
```

```
  With scenario submitting shopping cart
```

```
    Given I have typed my credit card number into the checkout page => Passed
    When I click the Buy button                                     => Passed
      And the browser posts my credit card number over the internet => Passed
    Then the form should be posted over https                     => Passed
```

See <http://storyq.codeplex.com/>



StorEvil



- Supports subset of gherkin
- It supports .NET languages such as C#.
- It has only been tested on Windows (so far).
- It only supports English (so far).
- Regular-Expression matching is supported
- Matching based on function names is preferred

<http://anotherdave.wordpress.com/2010/02/23/storevil-intro/>

