agile faqs

Acceptance Test Driven Development

Naresh Jain
naresh@agilefaqs.com
http://blogs.agilefaqs.com

agile faqs

Warmup Scenarios

agile faqs

Warmup Scenarios

Pick one scenario and in relation to your scenario, what are the specific observable results that will tell you that the activity has been successfully completed?

agile faqs

Warmup Scenarios

Pick one scenario and in relation to your scenario, what are the specific observable results that will tell you that the activity has been successfully completed?

☑ Going out for Movie (THX sound and Digital projection)

agile faqs

Warmup Scenarios

Pick one scenario and in relation to your scenario, what are the specific observable results that will tell you that the activity has been successfully completed?

- ☑ Going out for Movie (THX sound and Digital projection)
- ☑ Going out for meal (one veg.)

agile faqs

Warmup Scenarios

Pick one scenario and in relation to your scenario, what are the specific observable results that will tell you that the activity has been successfully completed?

- ☑ Going out for Movie (THX sound and Digital projection)
- ☑ Going out for meal (one veg.)
- ☑ Going shopping ($50)

agile faqs

Warmup Scenarios

Pick one scenario and in relation to your scenario, what are the specific observable results that will tell you that the activity has been successfully completed?

- ☑ Going out for Movie (THX sound and Digital projection)
- ☑ Going out for meal (one veg.)
- ☑ Going shopping ($50)

[10 Minutes]

agile faqs

Warmup Scenarios

Pick one scenario and in relation to your scenario, what are the specific observable results that will tell you that the activity has been successfully completed?

- ☑ Going out for Movie (THX sound and Digital projection)
- ☑ Going out for meal (one veg.)
- ☑ Going shopping ($50)

[10 Minutes]

- ☑ Present back to the group your findings. [3 minutes per group]

What is a Story?

Story is a smallest piece of functionality that add business value

Story Title - Actor Action Context

As a .. <user who requires this feature>

I want .. <do something>

So that... <user goal/business justification>

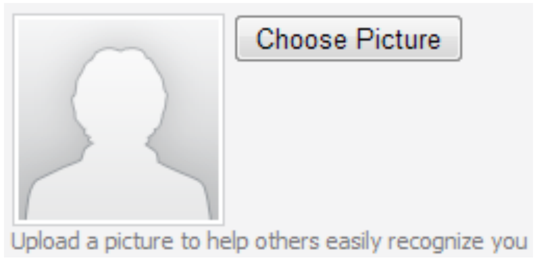
Ron Jeffries' 3 Cs - Card, Conversation and Confirmation

Story Example

- Title: Keen Reader subscribes to a blog
- As a keen reader of your blog
- I want to subscribe to your blog
- So that I can stay up-to-date with the new posts

Another Story Example

- ☑ Title: Social Networking Enthusiast uploads profile picture
- ☑ As a Social Networking Enthusiast
- ☑ I want to upload my profile picture
- ☑ So my friends can see how I look and recognize me



Upload a picture to help others easily recognize you

What makes a good Story?

What makes a good Story?

- ✓ Stories should follow the **INVEST** principle:

What makes a good Story?

- ☑ Stories should follow the **INVEST** principle:
- ☑ Independent

What makes a good Story?

- ☑ Stories should follow the **INVEST** principle:
- ☑ Independent
- ☑ Negotiable

What makes a good Story?

- ☑ Stories should follow the **INVEST** principle:
- ☑ Independent
- ☑ Negotiable
- ☑ Valuable

What makes a good Story?

- Stories should follow the **INVEST** principle:
- Independent
- Negotiable
- Valuable
- Estimate-able

What makes a good Story?

- ✓ Stories should follow the **INVEST** principle:
- ✓ Independent
- ✓ Negotiable
- ✓ Valuable
- ✓ Estimate-able
- ✓ Small

What makes a good Story?

- ✓ Stories should follow the **INVEST** principle:
- ✓ Independent
- ✓ Negotiable
- ✓ Valuable
- ✓ Estimate-able
- ✓ Small
- ✓ Testable

Stories are fundamental unit of activity

Stories are fundamental unit of activity

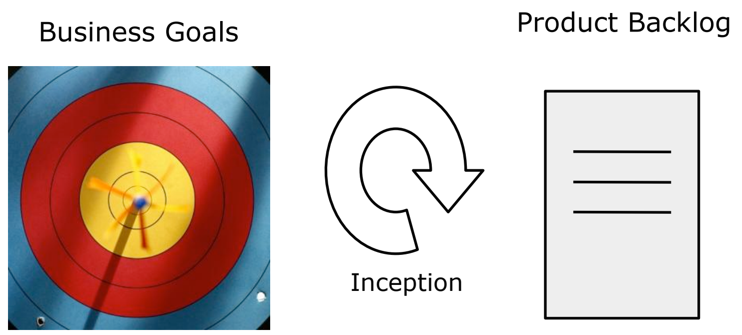
Business Goals

Stories are fundamental unit of activity

Business Goals

Inception

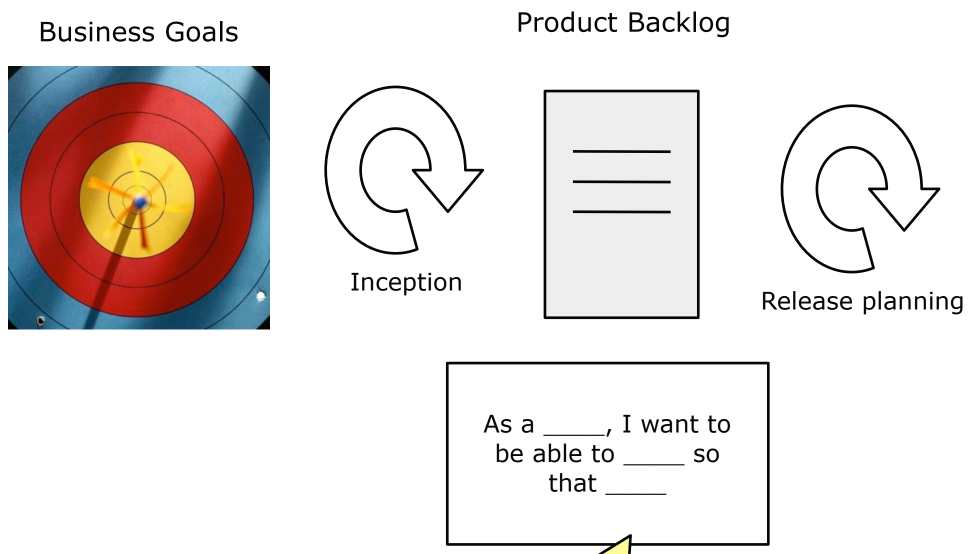
Stories are fundamental unit of activity



As a _____, I want to be able to _____ so that _____

Might have an initial estimate (perhaps for both analysis and development), and an expression of technical and business confidence that this is real and achievable

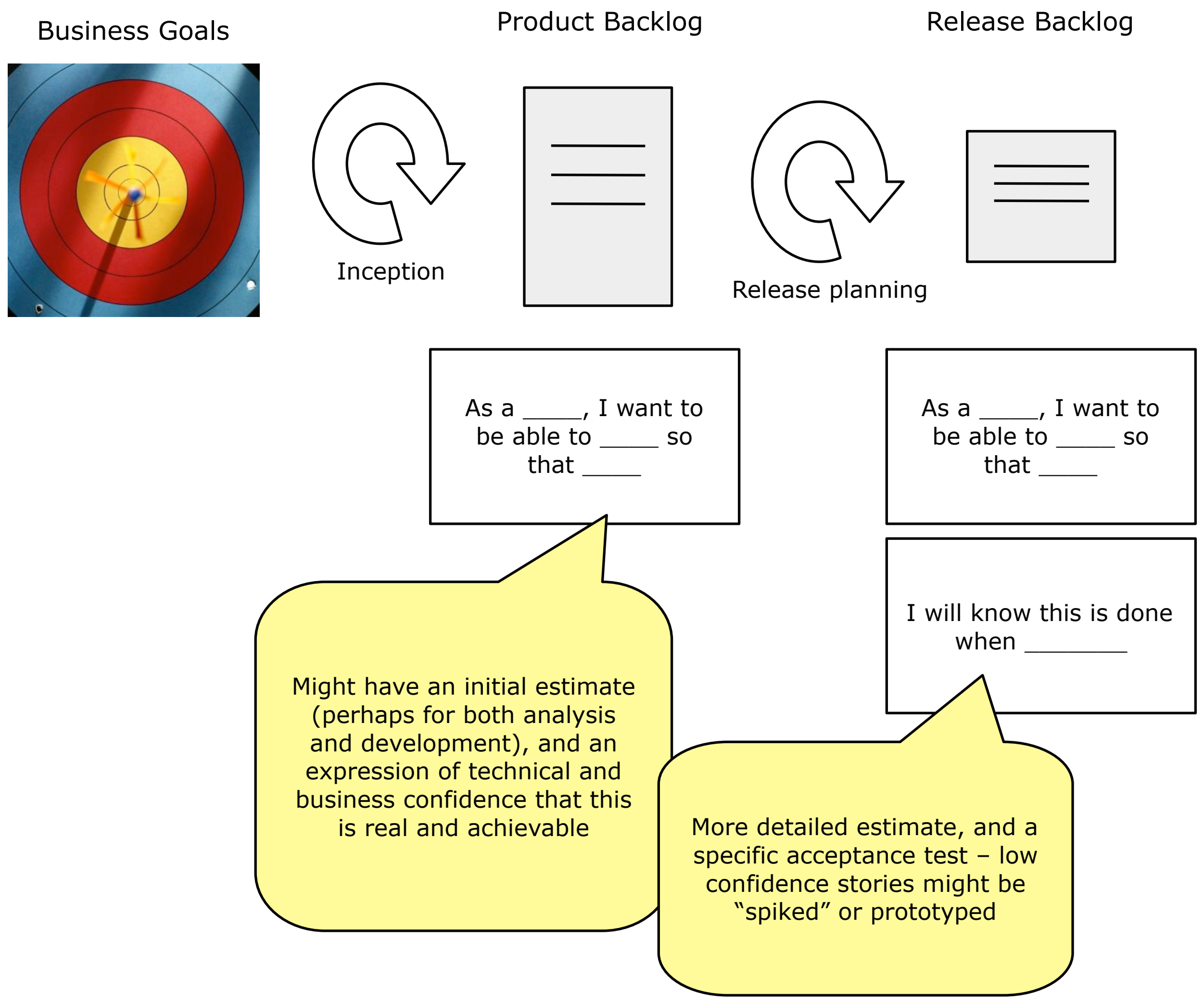
Stories are fundamental unit of activity



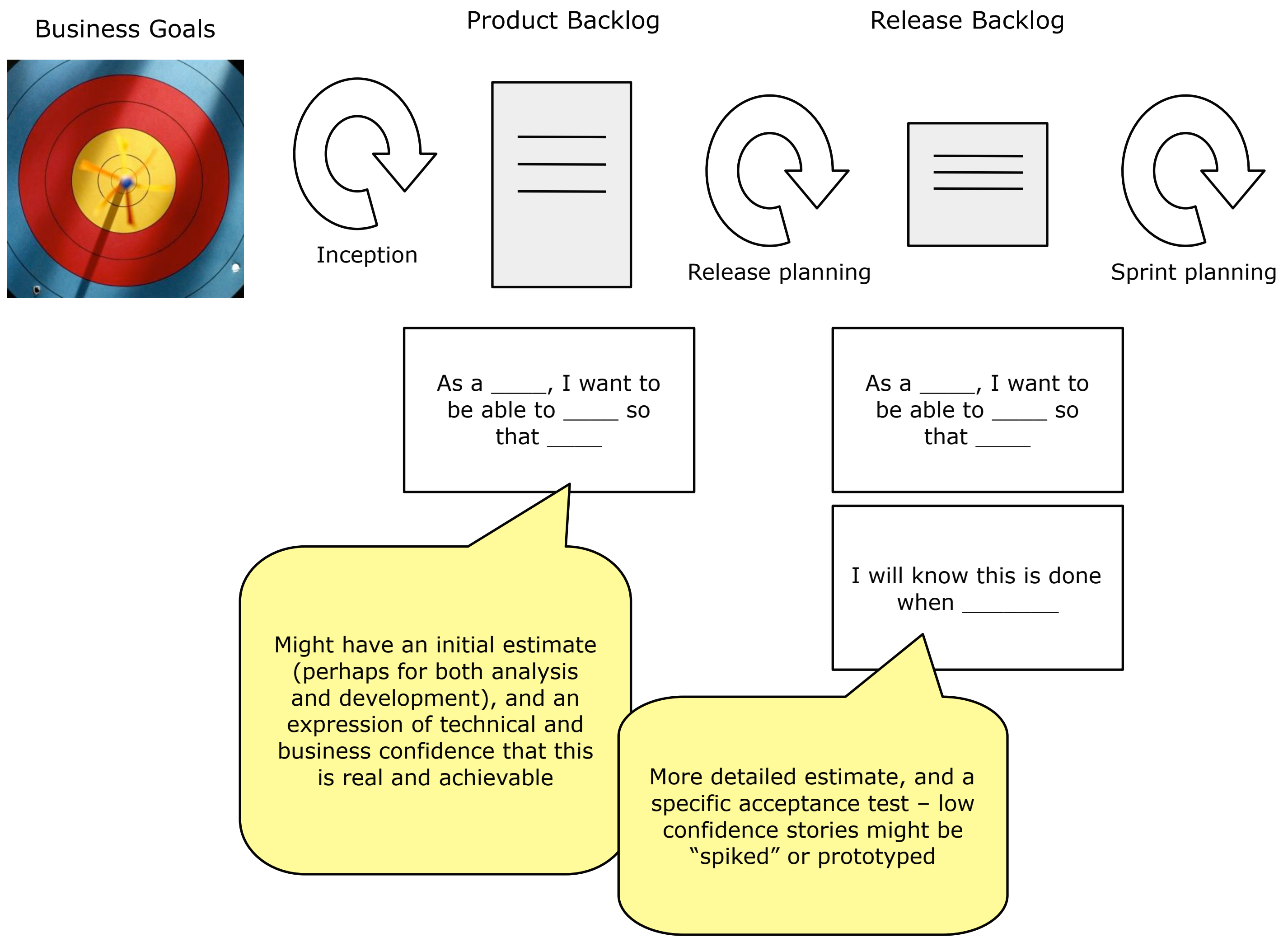
As a _____, I want to be able to _____ so that _____

Might have an initial estimate (perhaps for both analysis and development), and an expression of technical and business confidence that this is real and achievable

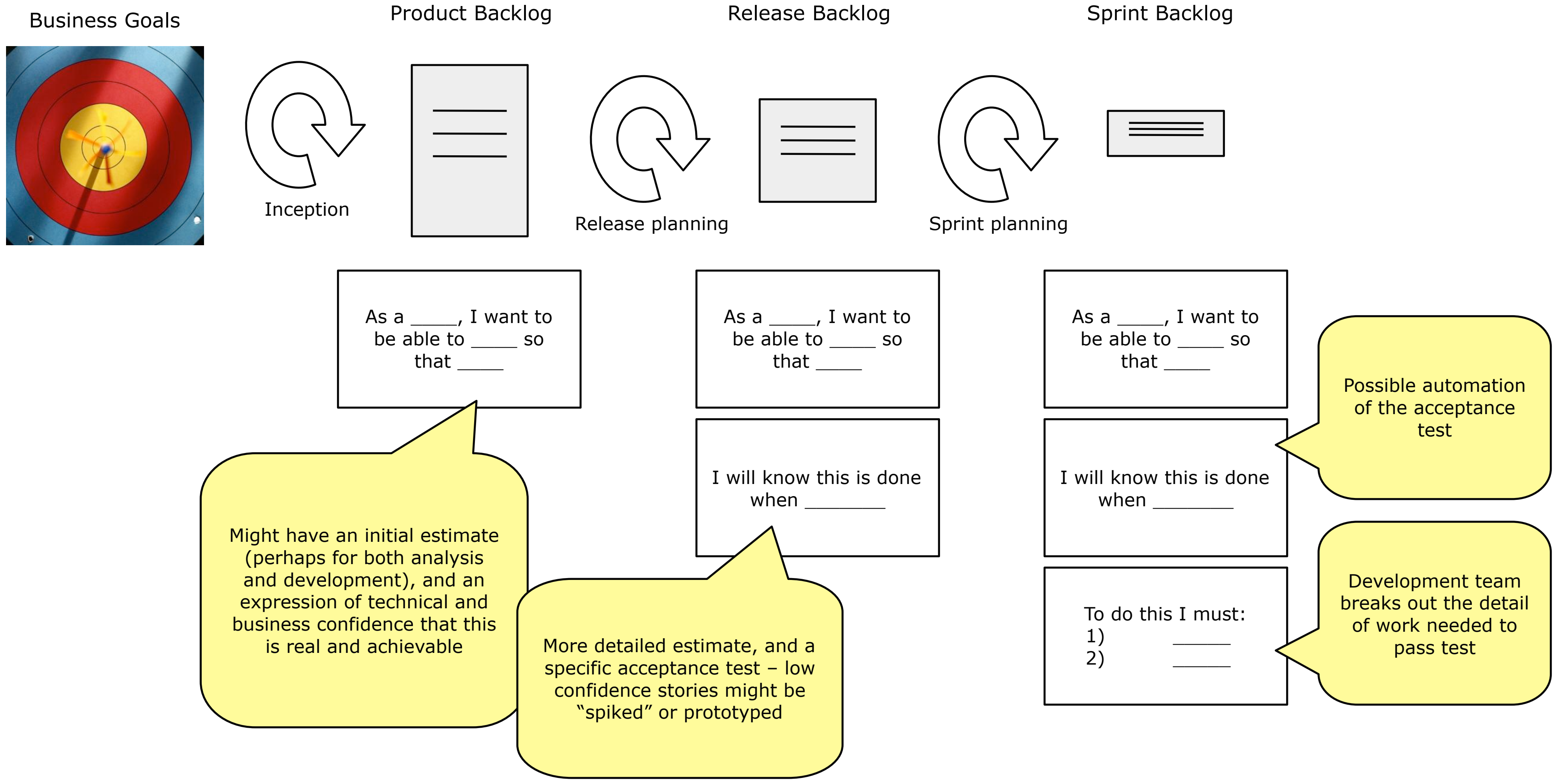
Stories are fundamental unit of activity


Business Goals
Inception
Product Backlog
Release planning
Release Backlog
As a _____, I want to be able to _____ so that _____
As a _____, I want to be able to _____ so that _____
I will know this is done when ________
Might have an initial estimate (perhaps for both analysis and development), and an expression of technical and business confidence that this is real and achievable
More detailed estimate, and a specific acceptance test – low confidence stories might be "spiked" or prototyped

Stories are fundamental unit of activity


Business Goals
Inception
Product Backlog
Release planning
Release Backlog
Sprint planning
As a _____, I want to be able to _____ so that _____
As a _____, I want to be able to _____ so that _____
I will know this is done when ________
Might have an initial estimate (perhaps for both analysis and development), and an expression of technical and business confidence that this is real and achievable
More detailed estimate, and a specific acceptance test – low confidence stories might be "spiked" or prototyped

Stories are fundamental unit of activity



Business Goals
Product Backlog
Release Backlog
Sprint Backlog
Inception
Release planning
Sprint planning
As a _____, I want to be able to _____ so that _____
As a _____, I want to be able to _____ so that _____
I will know this is done when ________
As a _____, I want to be able to _____ so that _____
I will know this is done when ________
To do this I must: 1) ______ 2) ______
Possible automation of the acceptance test
Development team breaks out the detail of work needed to pass test
Might have an initial estimate (perhaps for both analysis and development), and an expression of technical and business confidence that this is real and achievable
More detailed estimate, and a specific acceptance test – low confidence stories might be "spiked" or prototyped

Acceptance Criteria

- ☑ Is a set of conditions that the Story must meet for it to be accepted as complete
- ☑ Is typically provided by the customer or product owner.

Is not a replacement for conversation.

Is the results of the conversation

Acceptance Criteria are **NOT** tests

Writing Acceptance Criteria

Acceptance Criteria should contain:

- ☑ ACTOR
- ☑ VERB – DESCRIBING A BEHAVIOR
- ☑ OBSERVABLE RESULT

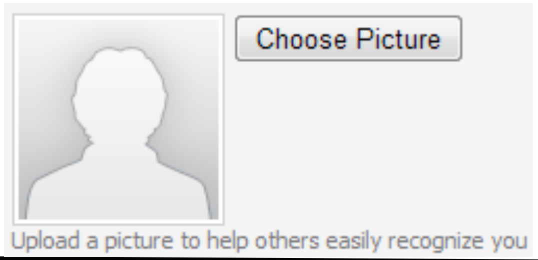
To accommodate pre-conditions Acceptance Criteria can be expressed as

Given [Precondition]

When [Actor + Action]

Then [Observable Result]

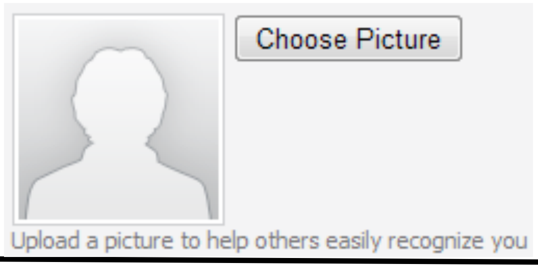
Example



Upload a picture to help others easily recognize you

- ☑ Social Networking Enthusiast uploads profile picture

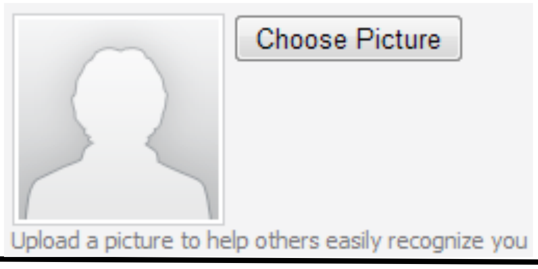
Example



- ✓ Social Networking Enthusiast uploads profile picture

- ✓ Given the user has a valid facebook account and a digital picture on her computer,
- ✓ When she uploads a picture in facebook,
- ✓ Then her the picture should be visible to all her friends in her network.

Example

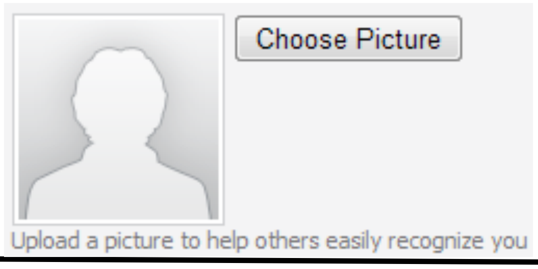


- Social Networking Enthusiast uploads profile picture

- Given the user has a valid facebook account and a digital picture on her computer,
- When she uploads a picture in facebook,
- Then her the picture should be visible to all her friends in her network.

- Given an user is trying to find a friend on facebook,
- When the user searches for a person using their name,
- Then their profile picture should be displayed along with other details.

Example



- ☑ Social Networking Enthusiast uploads profile picture

- ☑ Given the user has a valid facebook account and a digital picture on her computer,
- ☑ When she uploads a picture in facebook,
- ☑ Then her the picture should be visible to all her friends in her network.

- ☑ Given an user is trying to find a friend on facebook,
- ☑ When the user searches for a person using their name,
- ☑ Then their profile picture should be displayed along with other details.

- ☑ As owner of facebook,
- ☑ I want users to upload authentic, personal profile picture,
- ☑ So facebook's reputation remains intact and facebook stays out of legal hassles.

Acceptance Tests

agile faqs

Acceptance Criteria & Tests: Definition

Acceptance Tests

Acceptance Criteria

agile faqs

Acceptance Criteria & Tests: Definition

Acceptance Tests

Acceptance Criteria

+

Acceptance Criteria & Tests: Definition

Acceptance Tests

Acceptance Criteria

+ Examples (data + scenarios)

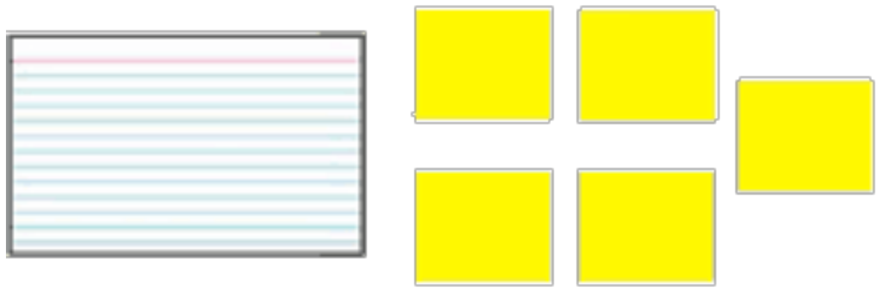
agile faqs

Acceptance Criteria & Tests: Definition

$$\text{Acceptance Tests} = \frac{\text{Acceptance Criteria} + \text{Examples (data + scenarios)}}{\text{Acceptance Tests}}$$

Tasks

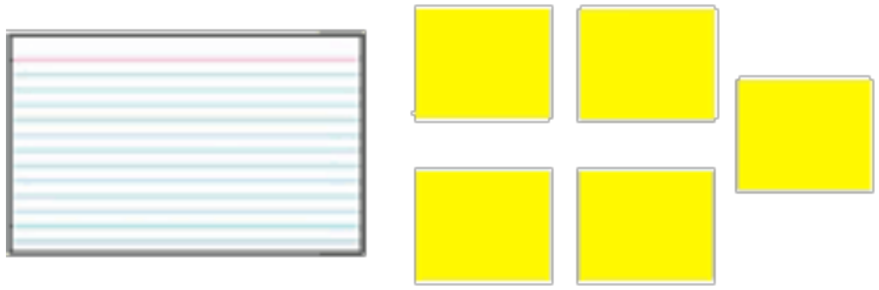
Team members further break down each story into tasks that need to be completed to meet the acceptance criteria for the story.



Tasks

Team members further break down each story into tasks that need to be completed to meet the acceptance criteria for the story.

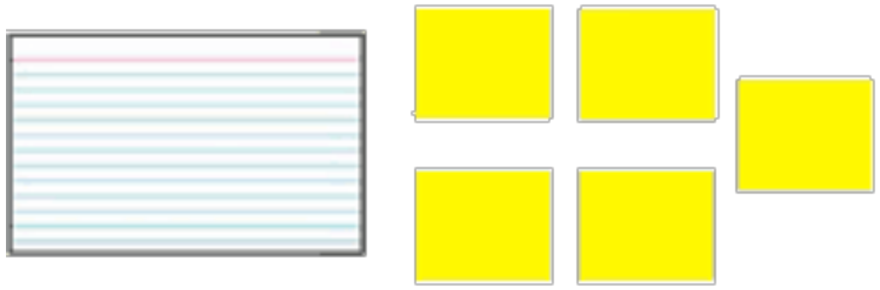
- ☑ To accomplish this story:
  - ☑ we start off with a simple upload and image display



Tasks

Team members further break down each story into tasks that need to be completed to meet the acceptance criteria for the story.

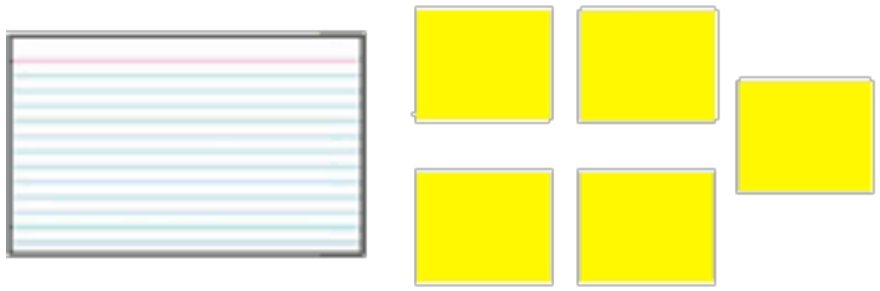
- ☑ To accomplish this story:
  - ☑ we start off with a simple upload and image display
  - ☑ restrict user to only upload certain image types (gif, jpg and png)



Tasks

Team members further break down each story into tasks that need to be completed to meet the acceptance criteria for the story.

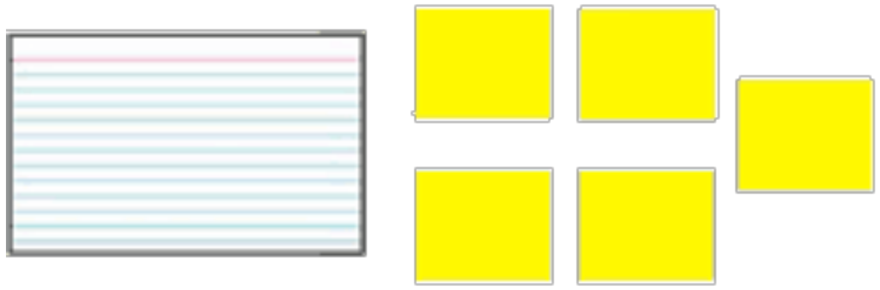
- To accomplish this story:
  - we start off with a simple upload and image display
  - restrict user to only upload certain image types (gif, jpg and png)
  - figure out where to store the image. (performant and fault-tolarent)



Tasks

Team members further break down each story into tasks that need to be completed to meet the acceptance criteria for the story.

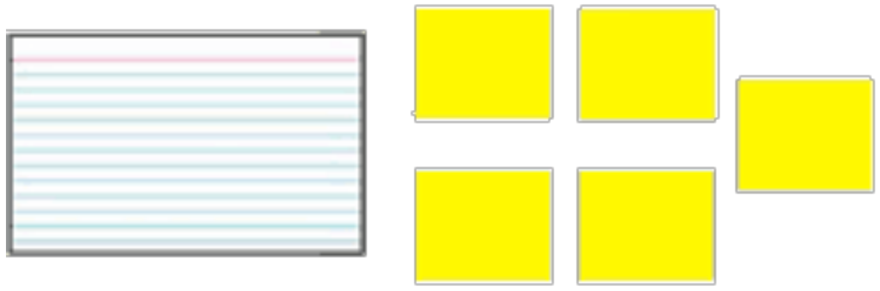
- To accomplish this story:
  - we start off with a simple upload and image display
  - restrict user to only upload certain image types (gif, jpg and png)
  - figure out where to store the image. (performant and fault-tolarent)
  - scale down (size, resolution, etc.) of the image



Tasks

Team members further break down each story into tasks that need to be completed to meet the acceptance criteria for the story.

- To accomplish this story:
  - we start off with a simple upload and image display
  - restrict user to only upload certain image types (gif, jpg and png)
  - figure out where to store the image. (performant and fault-tolarent)
  - scale down (size, resolution, etc.) of the image
  - and so on...



agile faqs

Demo

Roman Numerals to Decimal Conversion Example

agile faqs

Demo

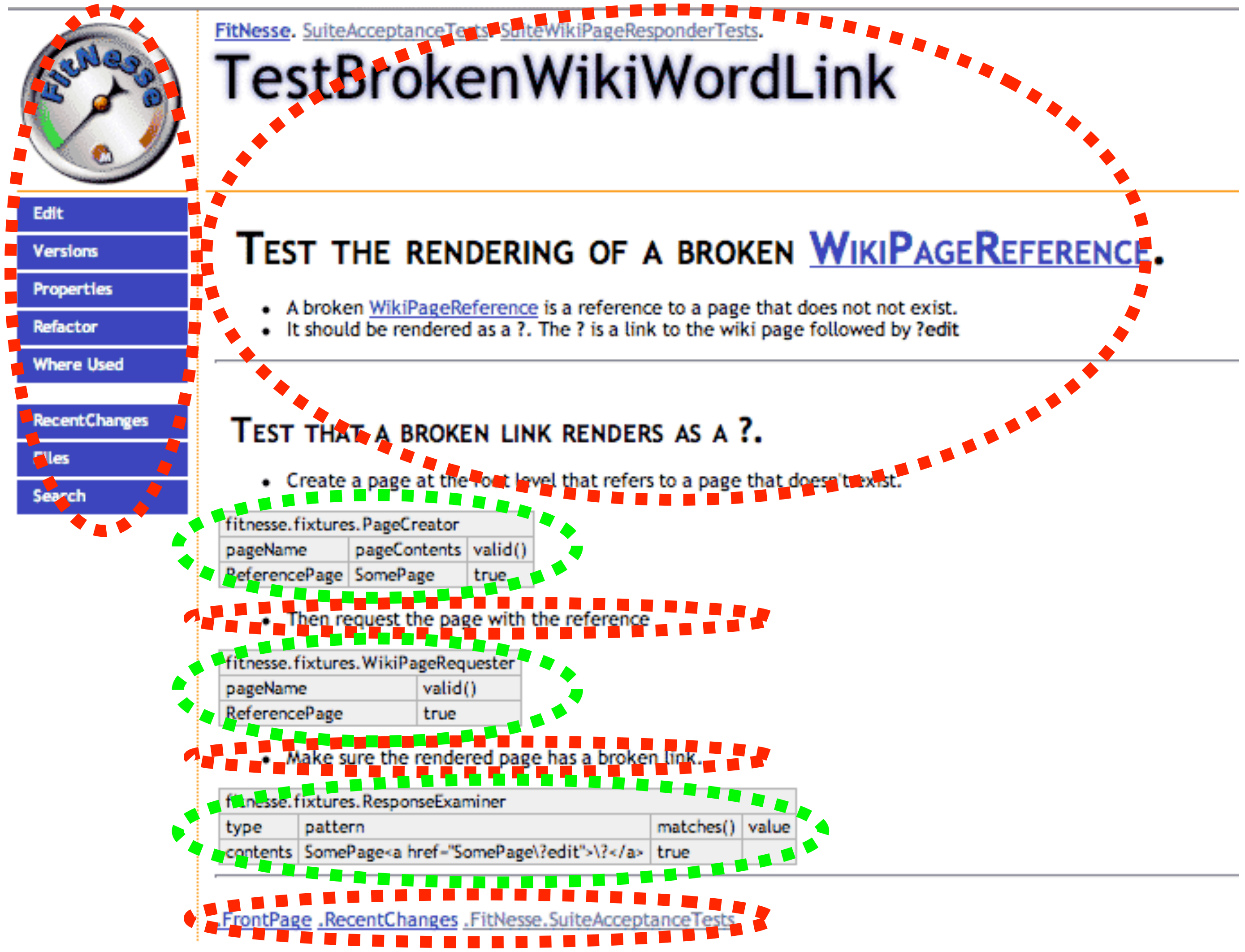
Real World Domain Forwarding Server

agile faqs

Thinking in Tables

Thinking in Tables

Only Tables Execute



FitNesse. SuiteAcceptanceTests. SuiteWikiPageResponderTests.

TestBrokenWikiWordLink

TEST THE RENDERING OF A BROKEN WIKIPAGEREFERENCE.

- A broken WikiPageReference is a reference to a page that does not not exist.
- It should be rendered as a ?. The ? is a link to the wiki page followed by ?edit

TEST THAT A BROKEN LINK RENDERS AS A ?.

- Create a page at the root level that refers to a page that doesn't exist.

fitnesse.fixtures.PageCreator		
pageName	pageContents	valid()
ReferencePage	SomePage	true

- Then request the page with the reference

fitnesse.fixtures.WikiPageRequester	
pageName	valid()
ReferencePage	true

- Make sure the rendered page has a broken link.

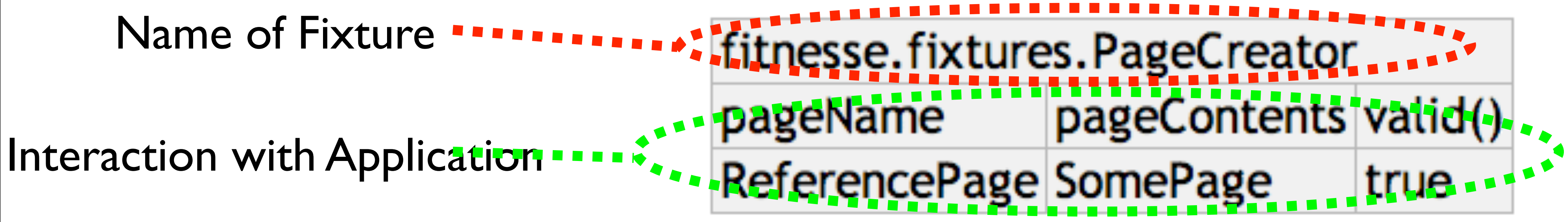
fitnesse.fixtures.ResponseExaminer			
type	pattern	matches()	value
contents	SomePage<a href="SomePage\?edit">\?</a>	true	

.FrontPage .RecentChanges .FitNesse.SuiteAcceptanceTests

Ignored

Executed

Foundational Table Structure

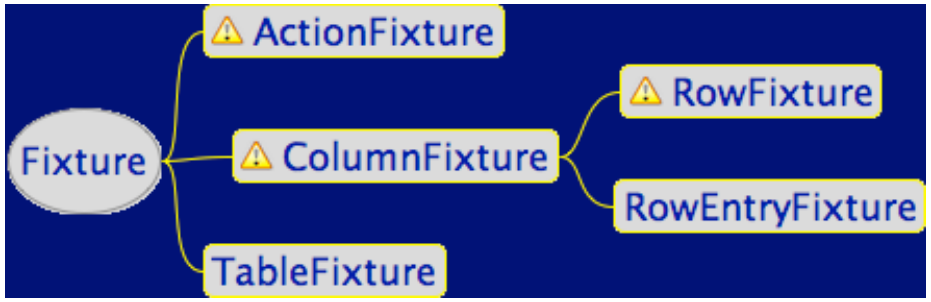


fitnesse.fixtures.PageCreator		
pageName	pageContents	valid()
ReferencePage	SomePage	true

☑ Table structure depends on type of Fixture

3 Foundation Fixtures

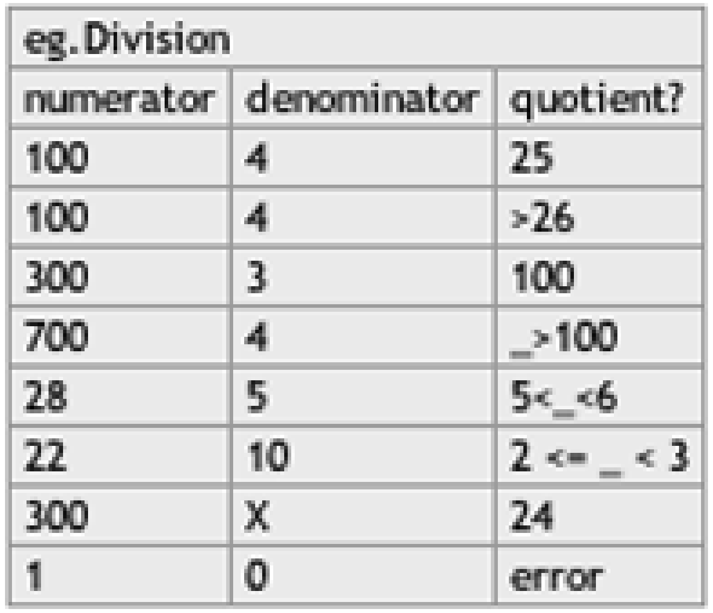
- ✓ Column Fixture
- ✓ Row Fixture
- ✓ Action Fixture



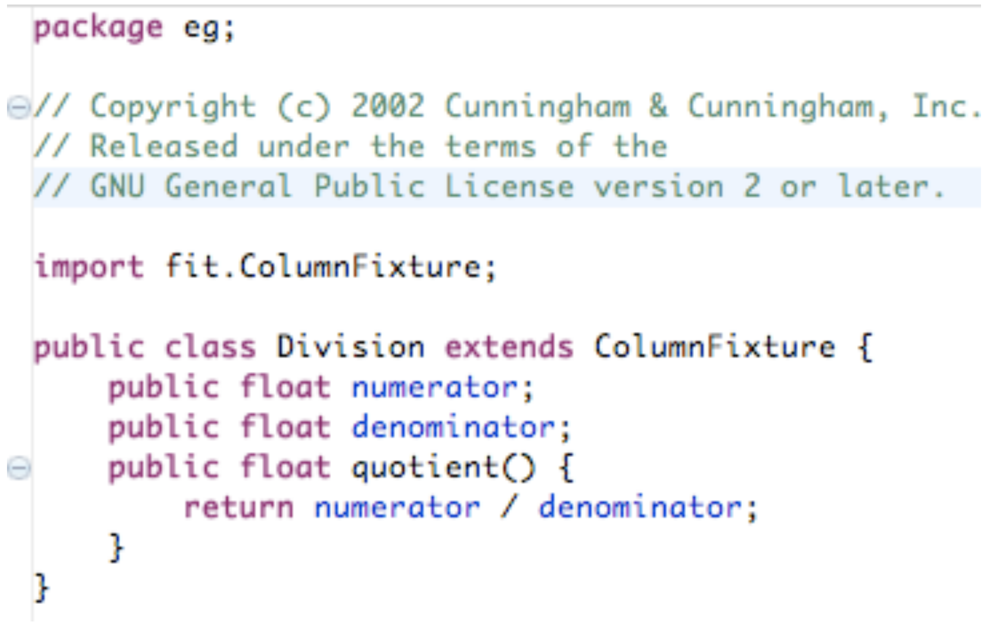
agile faqs

Thinking in Tables

Column Fixture

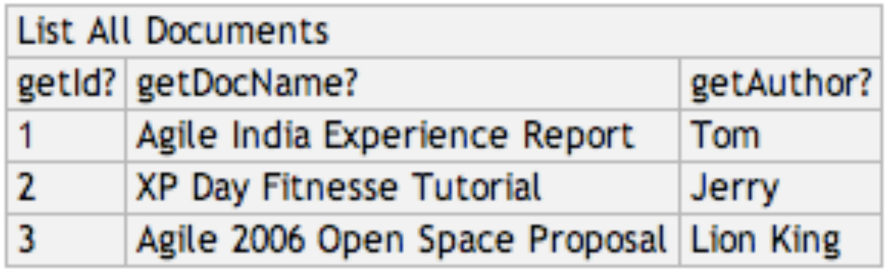


eg.Division		
numerator	denominator	quotient?
100	4	25
100	4	>26
300	3	100
700	4	_>100
28	5	5<_<6
22	10	2 <= _ < 3
300	X	24
1	0	error



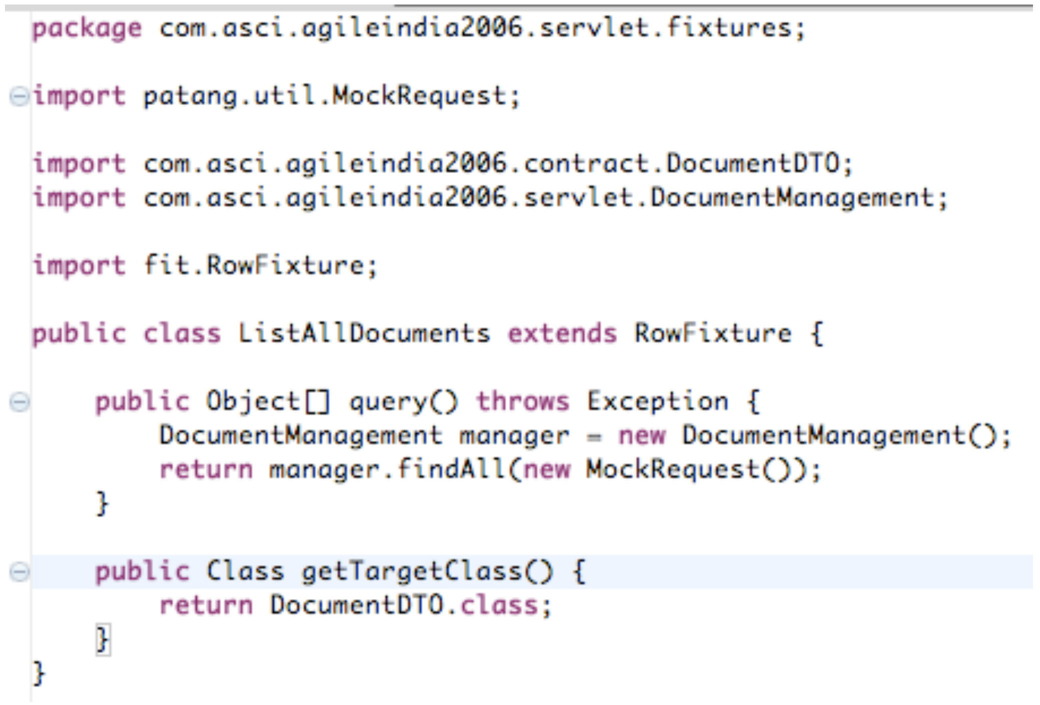
```
package eg;

// Copyright (c) 2002 Cunningham & Cunningham, Inc.
// Released under the terms of the
// GNU General Public License version 2 or later.

import fit.ColumnFixture;

public class Division extends ColumnFixture {
    public float numerator;
    public float denominator;
    public float quotient() {
        return numerator / denominator;
    }
}
```

Row Fixture

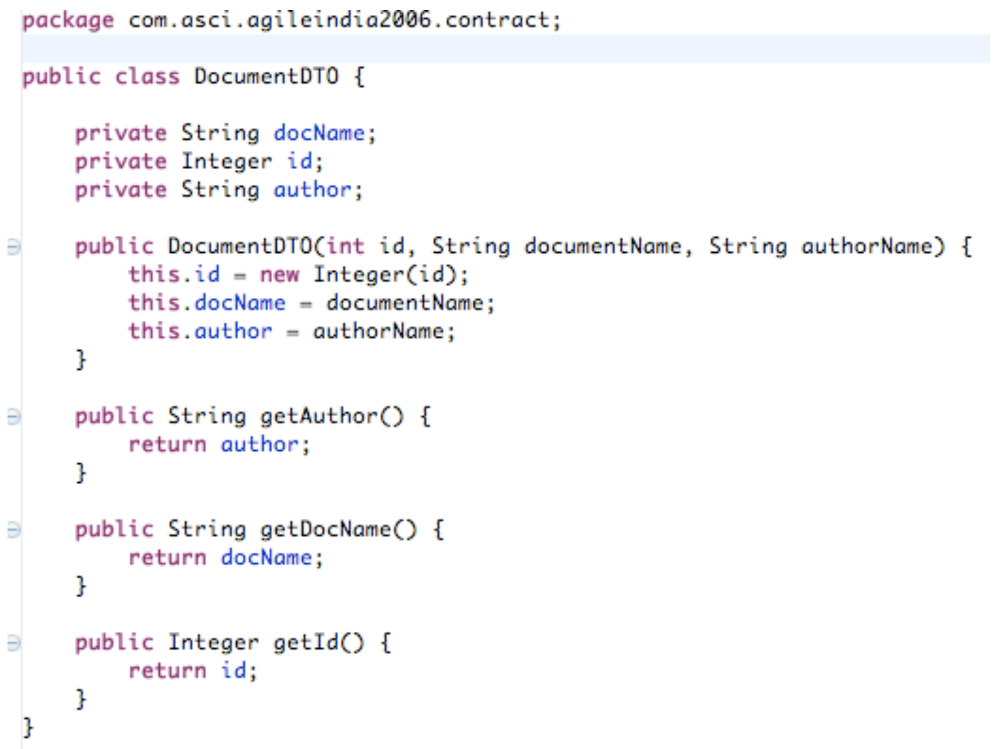


List All Documents		
getId?	getDocName?	getAuthor?
1	Agile India Experience Report	Tom
2	XP Day Fitnesse Tutorial	Jerry
3	Agile 2006 Open Space Proposal	Lion King

Analogous to comparing against rows in a database table



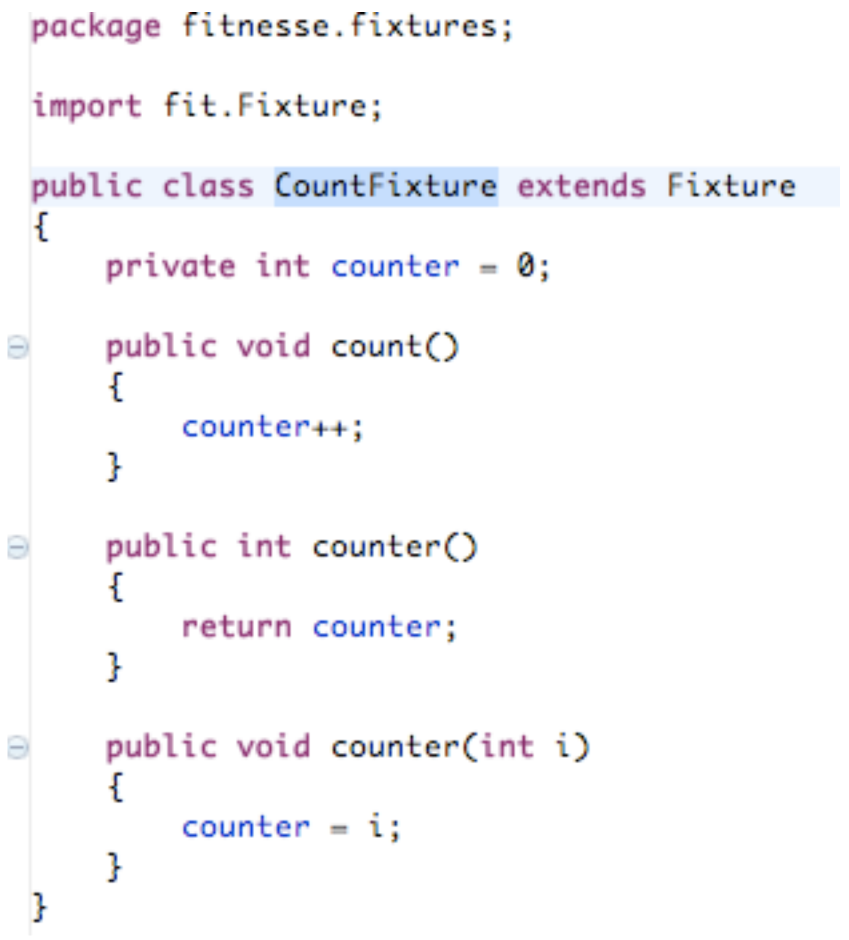
```java
package com.asci.agileindia2006.servlet.fixtures;

import patang.util.MockRequest;

import com.asci.agileindia2006.contract.DocumentDTO;
import com.asci.agileindia2006.servlet.DocumentManagement;

import fit.RowFixture;

public class ListAllDocuments extends RowFixture {

    public Object[] query() throws Exception {
        DocumentManagement manager = new DocumentManagement();
        return manager.findAll(new MockRequest());
    }

    public Class getTargetClass() {
        return DocumentDTO.class;
    }
}
```



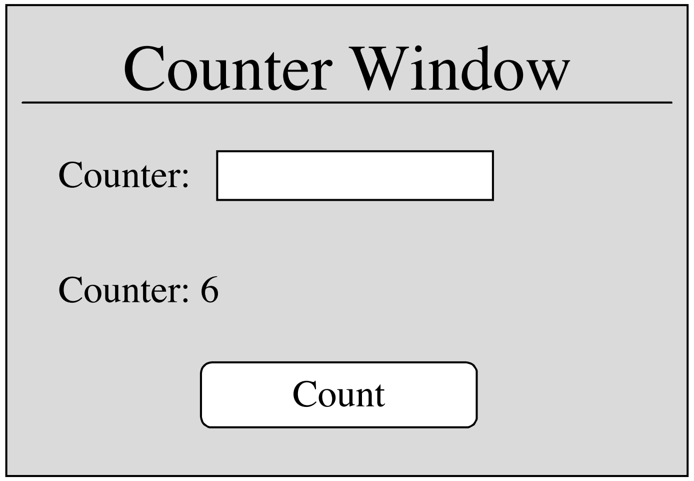
```java
package com.asci.agileindia2006.contract;

public class DocumentDTO {

    private String docName;
    private Integer id;
    private String author;

    public DocumentDTO(int id, String documentName, String authorName) {
        this.id = new Integer(id);
        this.docName = documentName;
        this.author = authorName;
    }

    public String getAuthor() {
        return author;
    }

    public String getDocName() {
        return docName;
    }

    public Integer getId() {
        return id;
    }
}
```

Action Fixture

☑ Think GUI window



```
package fitnesse.fixtures;

import fit.Fixture;

public class CountFixture extends Fixture
{
    private int counter = 0;

    public void count()
    {
        counter++;
    }

    public int counter()
    {
        return counter;
    }

    public void counter(int i)
    {
        counter = i;
    }
}
```



Counter Window

Counter:

Counter: 6

fit.ActionFixture		
start	fitnesse.fixtures.CountFixture	
check	counter	0
press	count	
check	counter	1
press	count	
check	counter	2
enter	counter	5
press	count	
check	counter	6

agile faqs

FitLibrary

- ☑ Extension to FIT
- ☑ Written by Rick Mugridge
- ☑ Adds some handy Fixtures

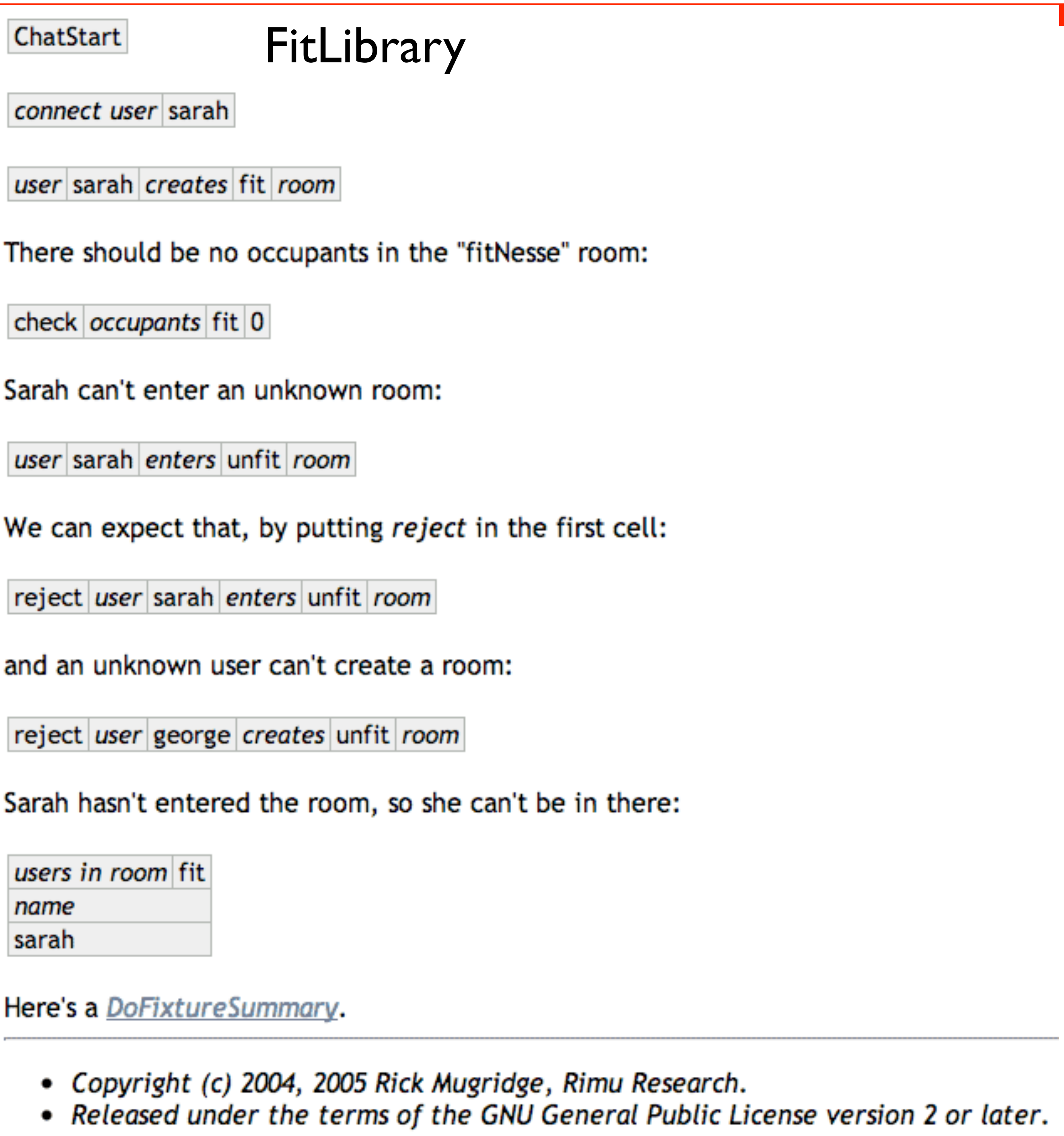
FitLibrary Fixtures

- ☑ ArrayFixture for ordered lists
- ☑ SetFixture for unordered lists
- ☑ SetUpFixture
- ☑ Supports
  - ☑ Graphics
  - ☑ Tree structures
  - ☑ Nested Tables

agile faqs

DoFixture

- Broken tables
- Highly readable
- Flexibility



ChatStart

FitLibrary

connect user	sarah

user	sarah	creates	fit	room

There should be no occupants in the "fitNesse" room:

check	occupants	fit	0

Sarah can't enter an unknown room:

user	sarah	enters	unfit	room

We can expect that, by putting *reject* in the first cell:

reject	user	sarah	enters	unfit	room

and an unknown user can't create a room:

reject	user	george	creates	unfit	room

Sarah hasn't entered the room, so she can't be in there:

users in room	fit
name	
sarah	

Here's a DoFixtureSummary.

agile faqs

Tools

FIT

- ☑ Framework for Integrated Tests
- ☑ Created by Ward Cunningham
- ☑ Open Source
- ☑ The most accepted solution for agile acceptance testing

agile faqs

Tools

FitNesse

- ✓ Environment build around FIT
- ✓ Makes everything easier
- ✓ Created by Object Mentor, Inc.
- ✓ Open Source

agile faqs

FIT

- Tests written in HTML
- Tests are executed on the command line
- Tables are executed
- Non-table markup is ignored
- Tables map to Fixtures
- Fixtures are code that is aware of the system
- Supplies foundational Fixtures
- Implementations ported to many languages

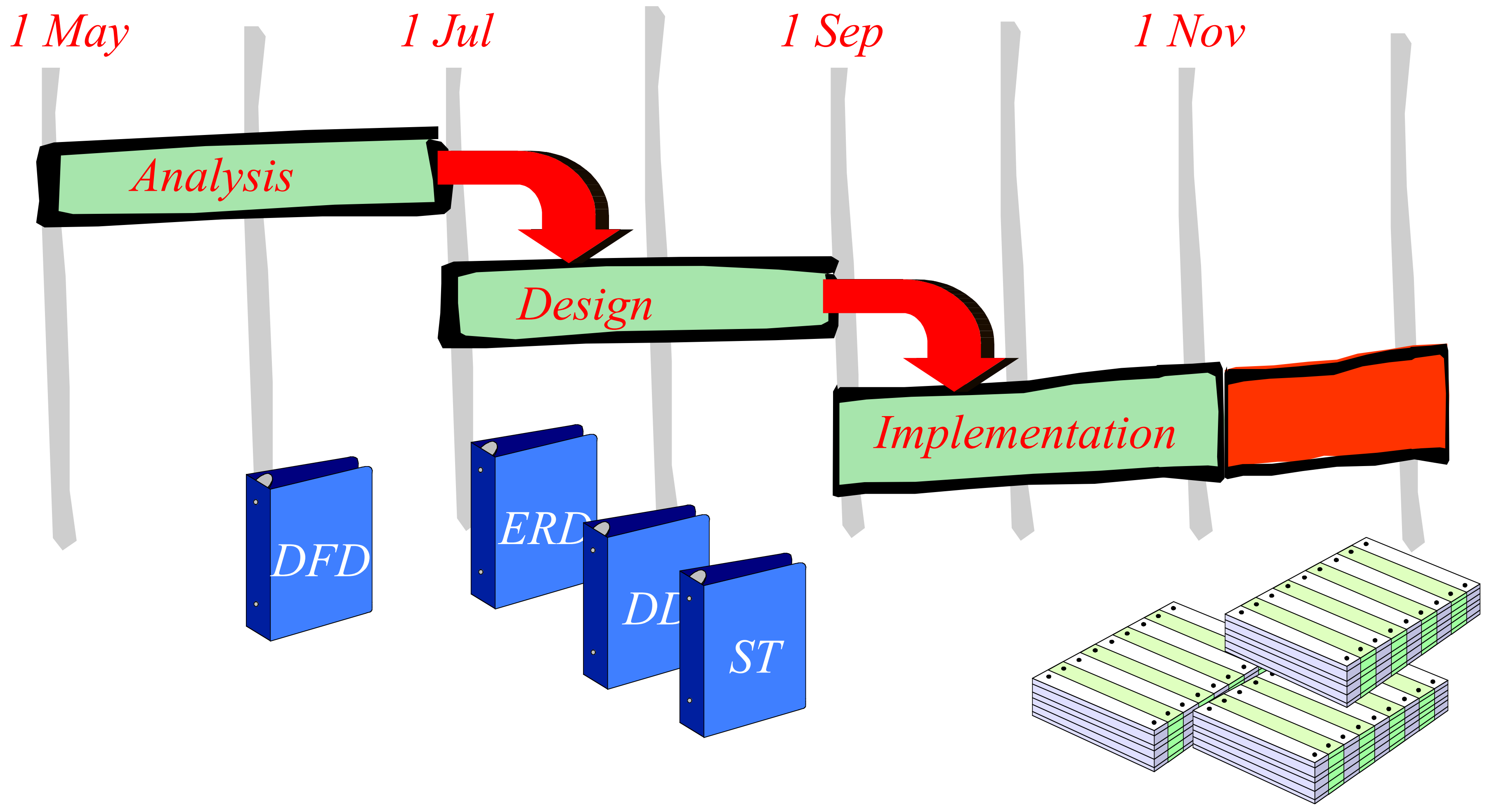
FitNesse

- Stand alone web server
- Is a wiki
- Tests written in wiki text
- Tests are executed from within the wiki
- Translates tests into HTML
- Uses FIT to execute tests
- Supports test suites
- Supports variables in tests
- Supports test refactoring
- Written in Java
- Supports FIT implementations in any language

agile faqs

Acceptance Criteria and Tests: A Critical Piece of Agile

Traditional Approach



Key Questions

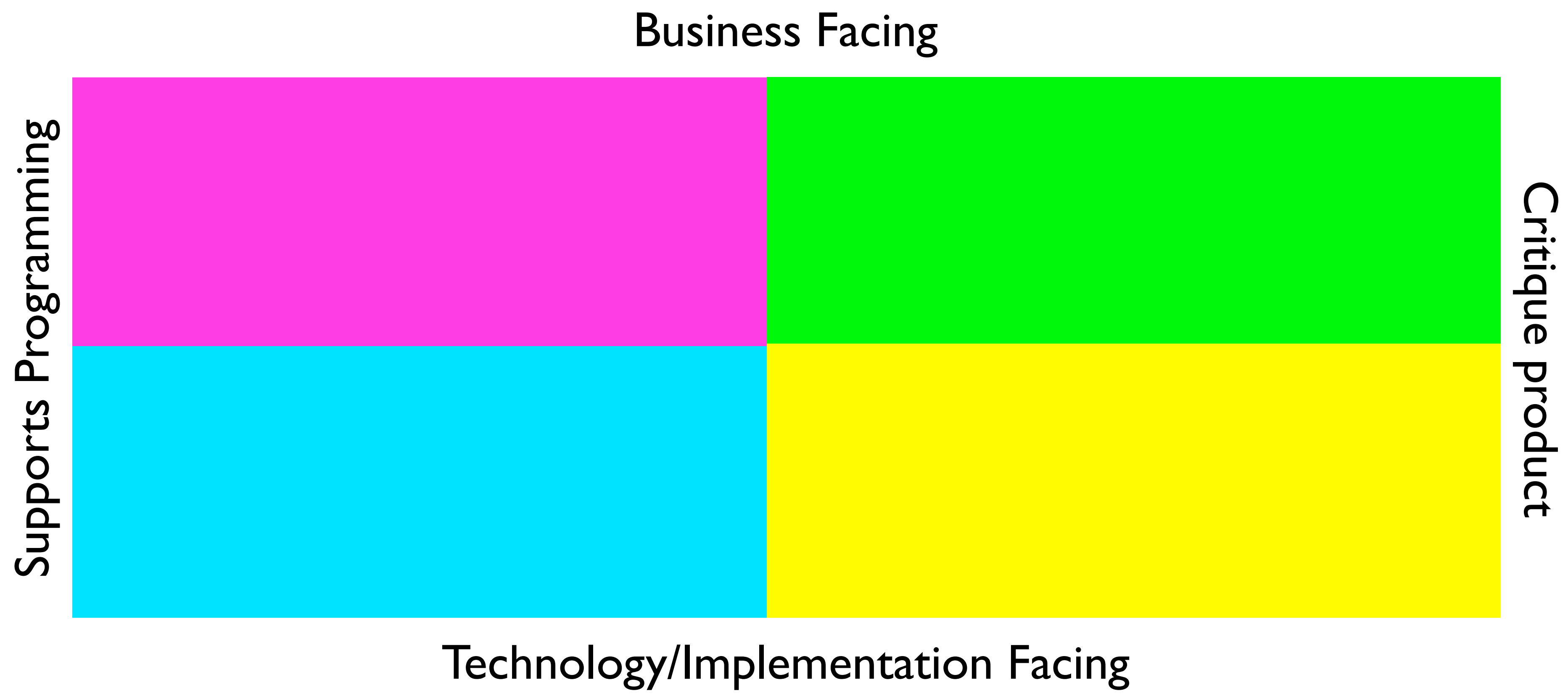
Business Facing

Are we building the right product?

Are we building the product right?

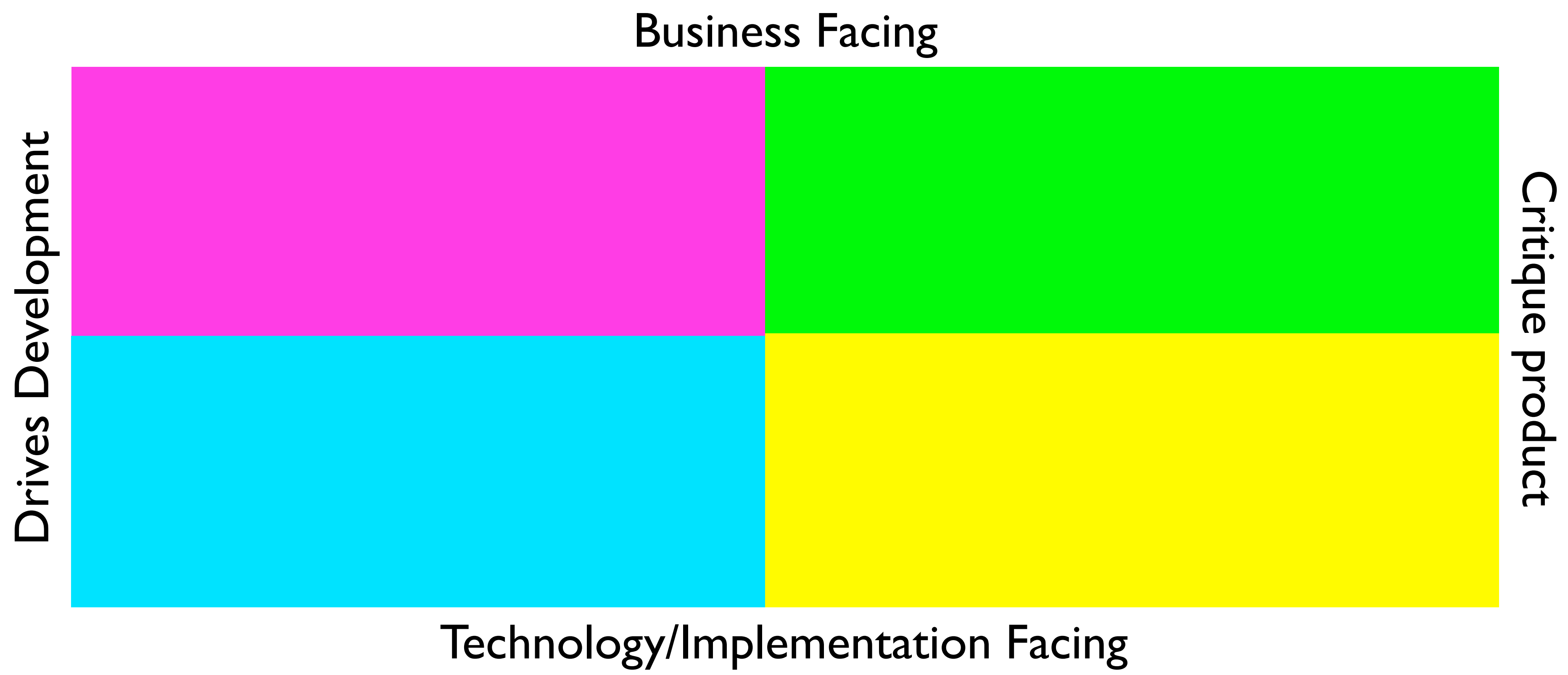
Technology/Implementation Facing

Brian Marick's Test Categorization



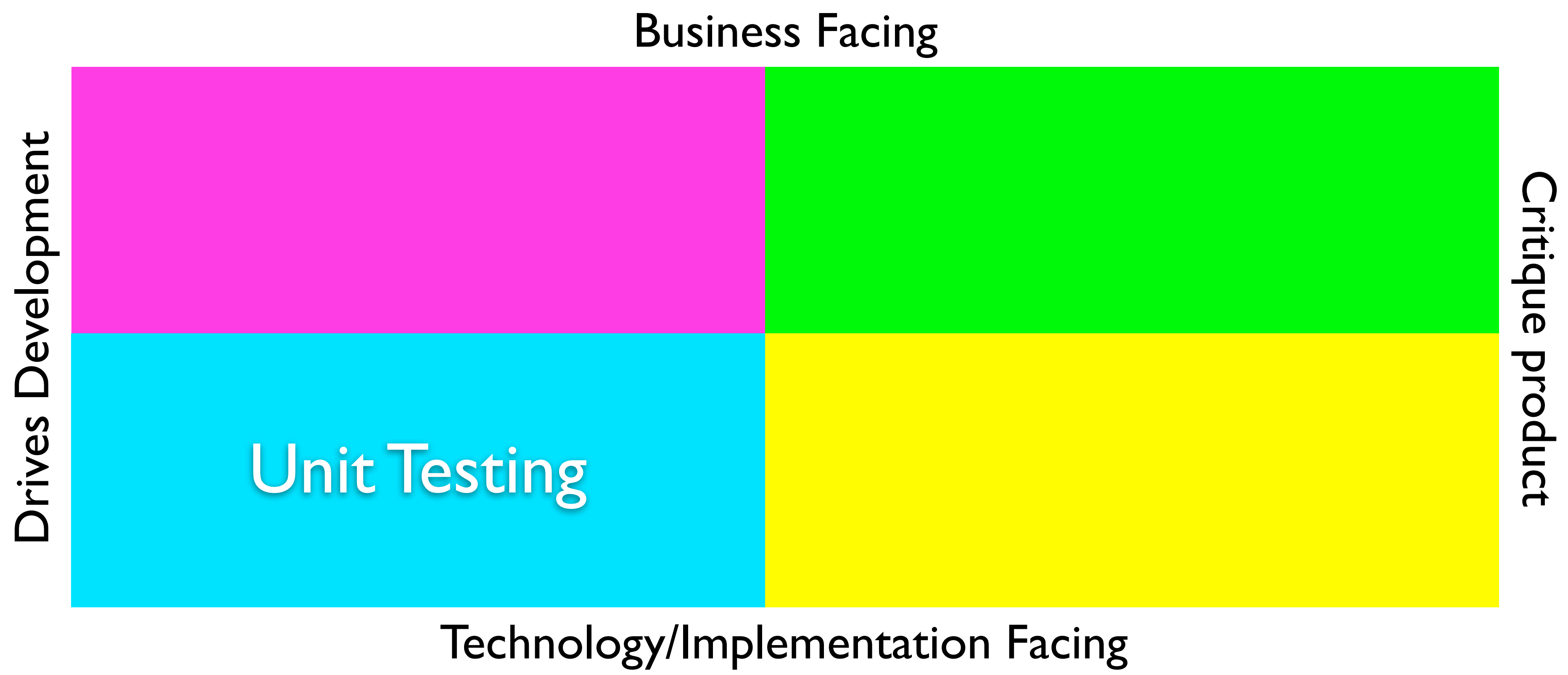
agile faqs

It Helps to Think of Tests this way...



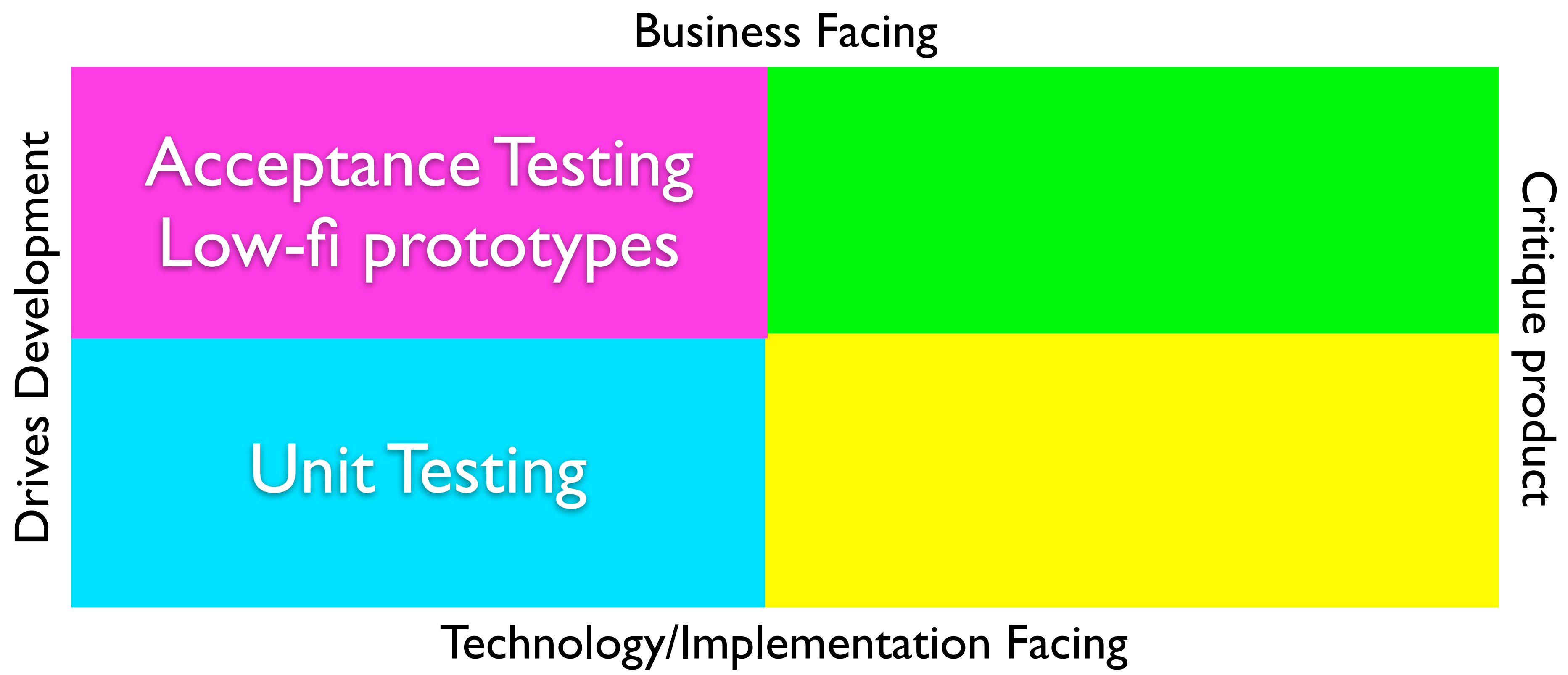
agile faqs

It Helps to Think of Tests this way...



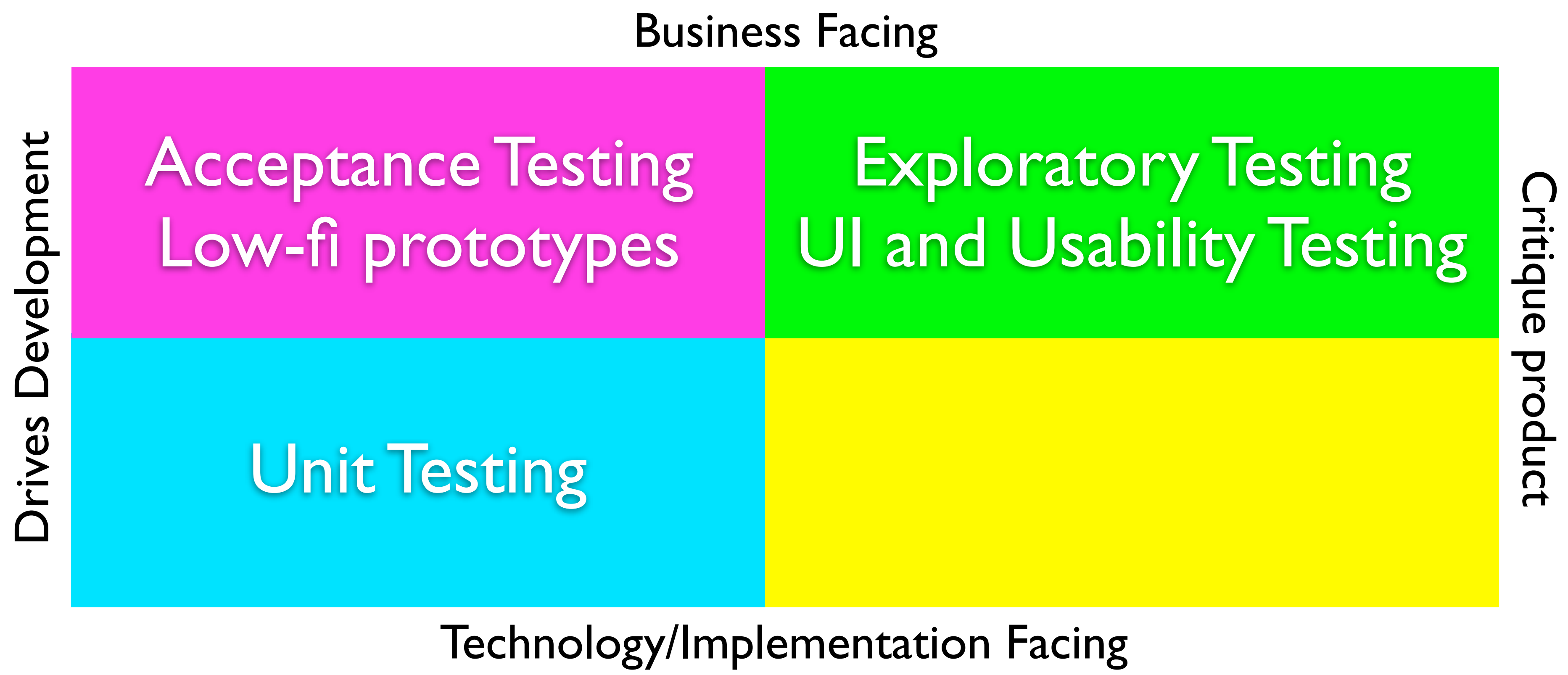
agile faqs

It Helps to Think of Tests this way...



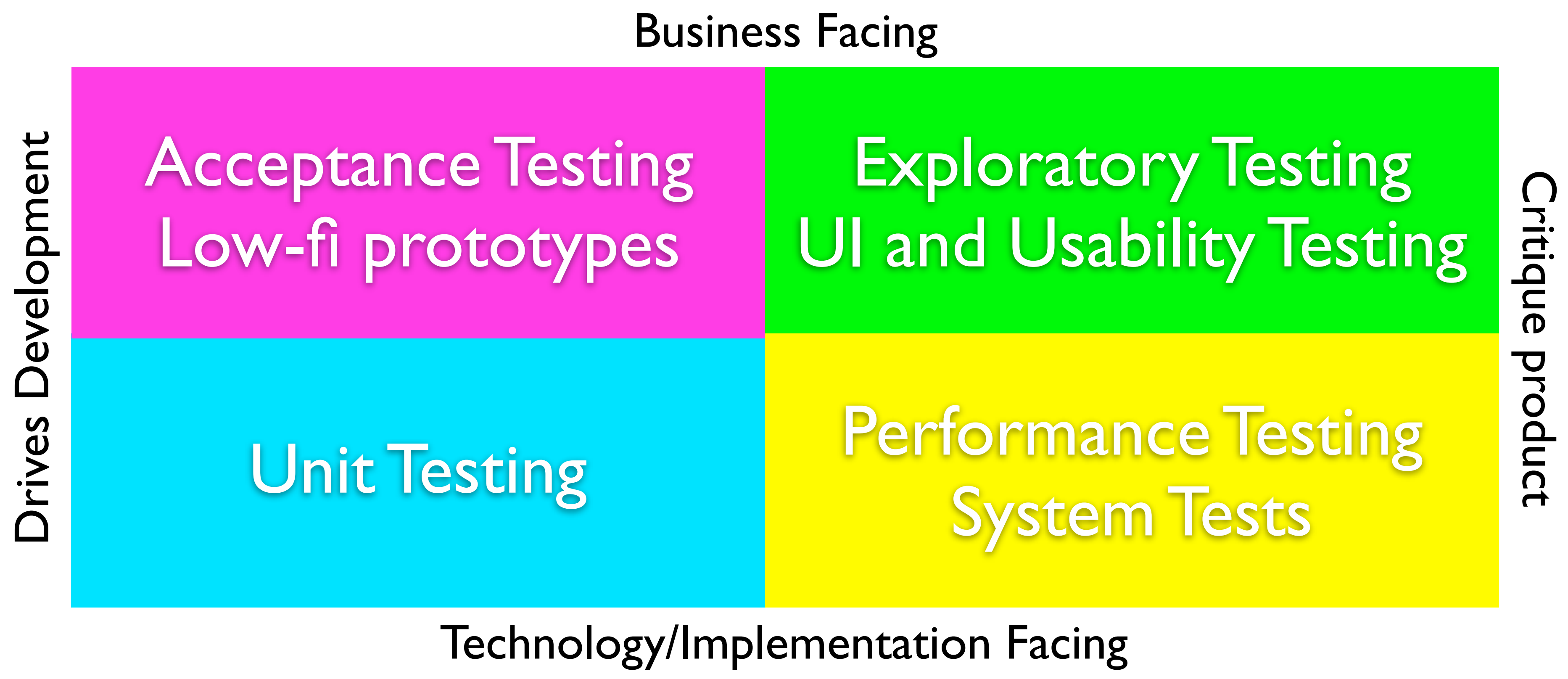
agile faqs

It Helps to Think of Tests this way...



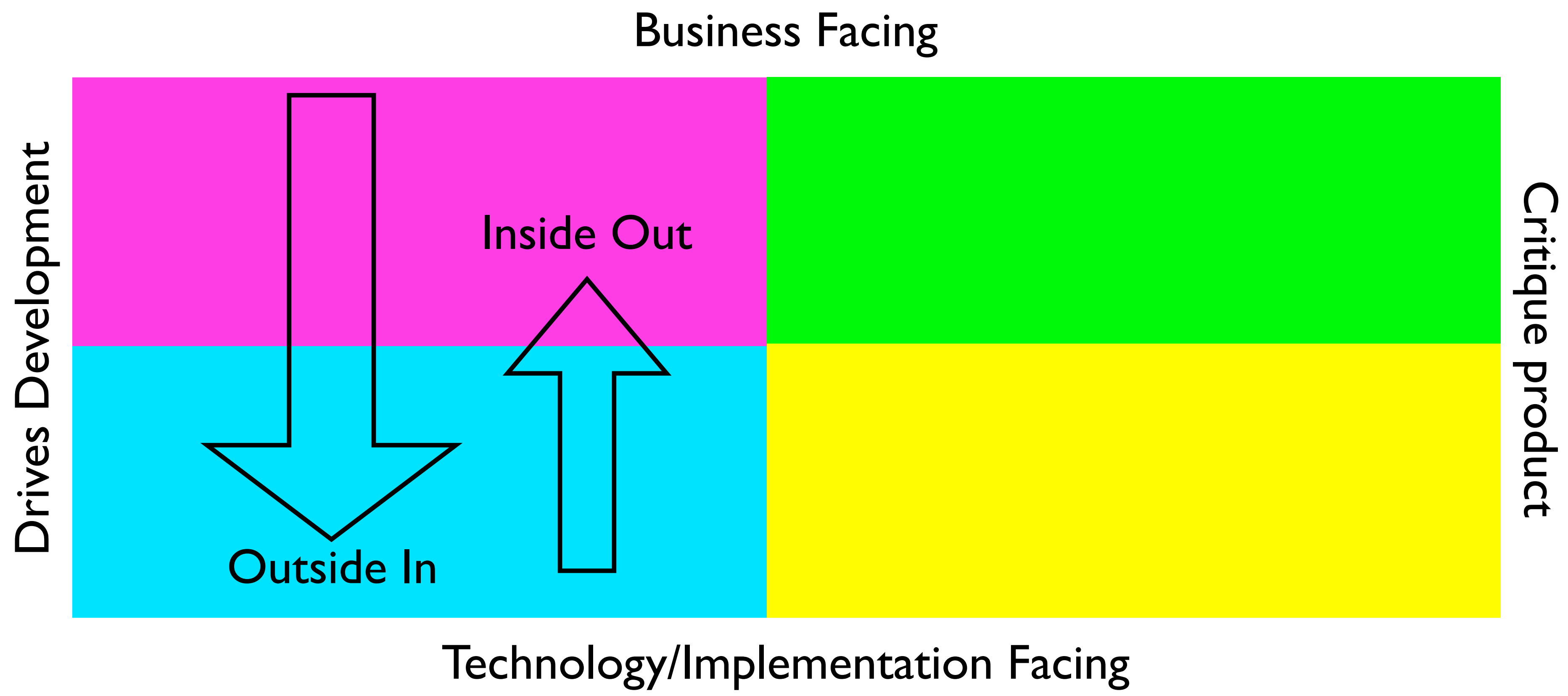
agile faqs

It Helps to Think of Tests this way...



agile faqs

Avatars of TDD



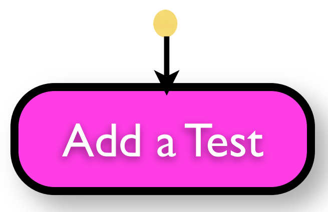
agile faqs

Test Driven Development

TDD Rhythm - Test, Code, Refactor

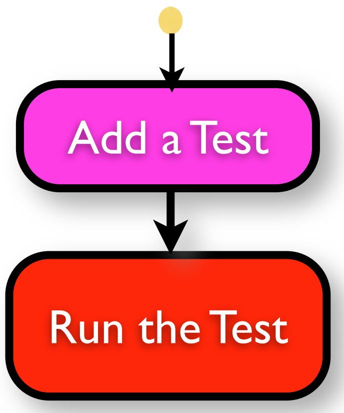
agile faqs

Test Driven Development



TDD Rhythm - Test, Code, Refactor

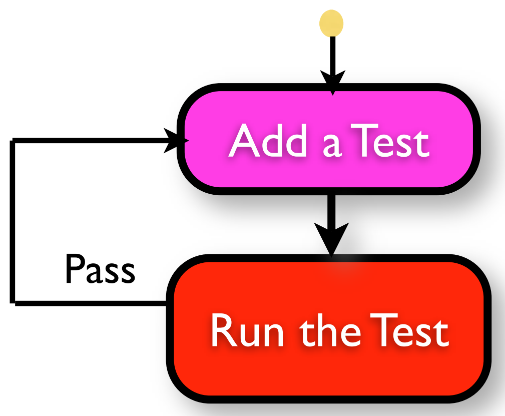
Test Driven Development



TDD Rhythm - Test, Code, Refactor

agile faqs

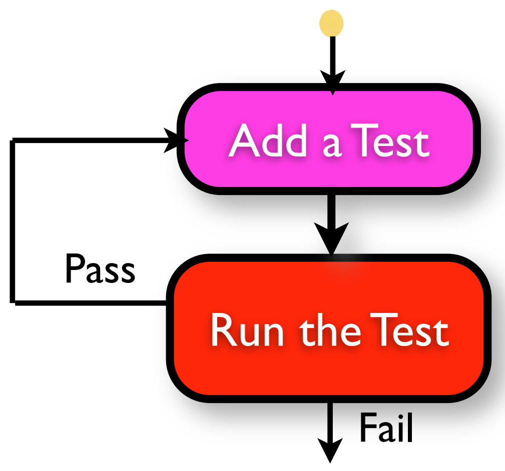
Test Driven Development



TDD Rhythm - Test, Code, Refactor

agile faqs

Test Driven Development

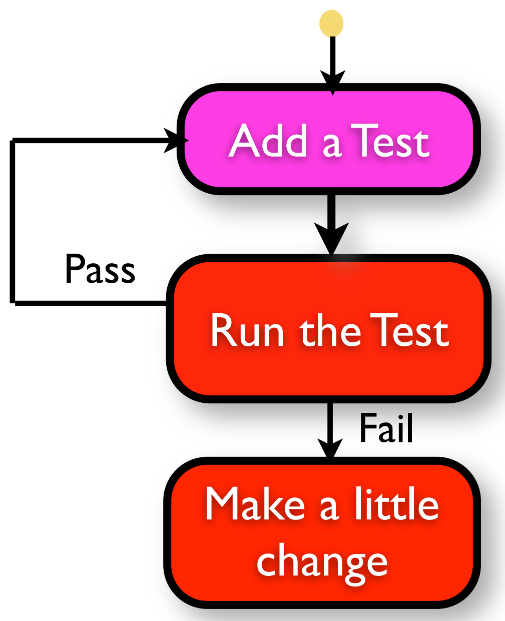


TDD Rhythm - Test, Code, Refactor

agile faqs

Test Driven Development

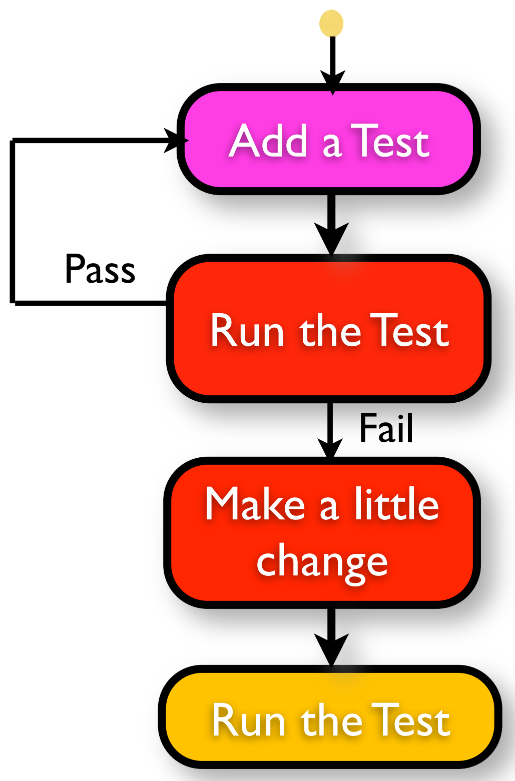
TDD Rhythm - Test, Code, Refactor



agile faqs

Test Driven Development

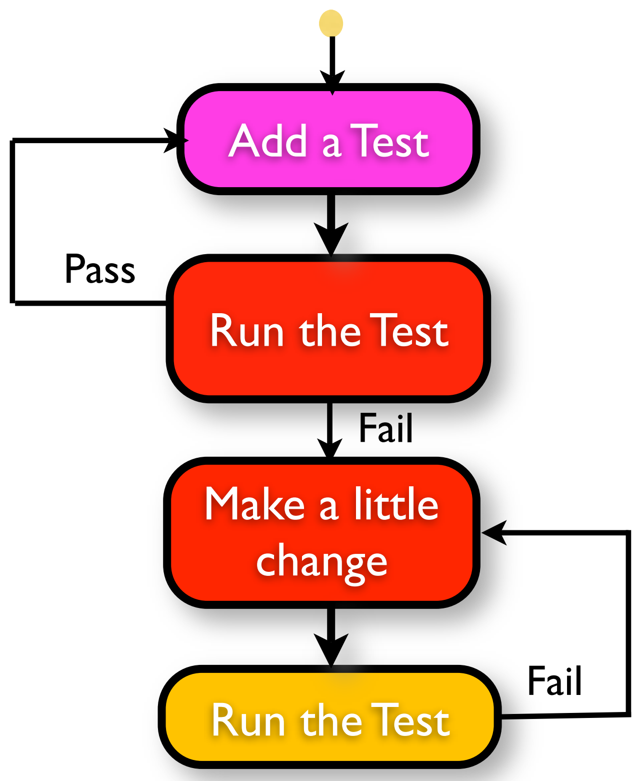
TDD Rhythm - Test, Code, Refactor



agile faqs

Test Driven Development

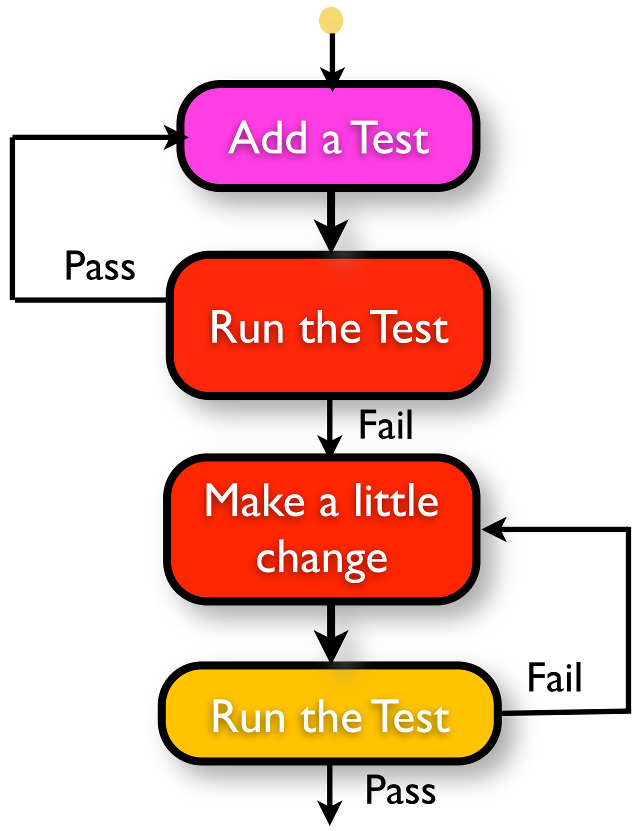
TDD Rhythm - Test, Code, Refactor



agile faqs

Test Driven Development

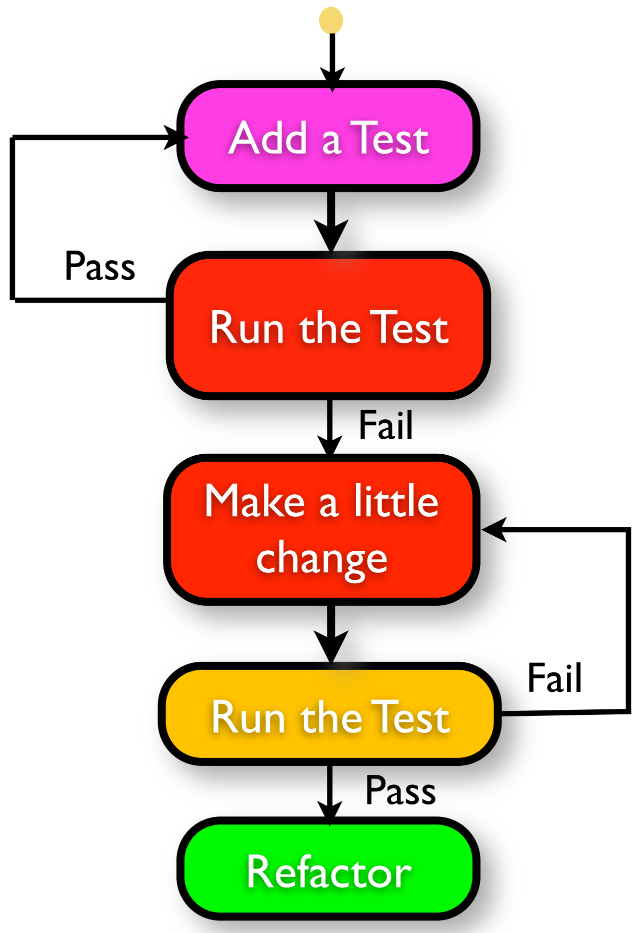
TDD Rhythm - Test, Code, Refactor



agile faqs

Test Driven Development

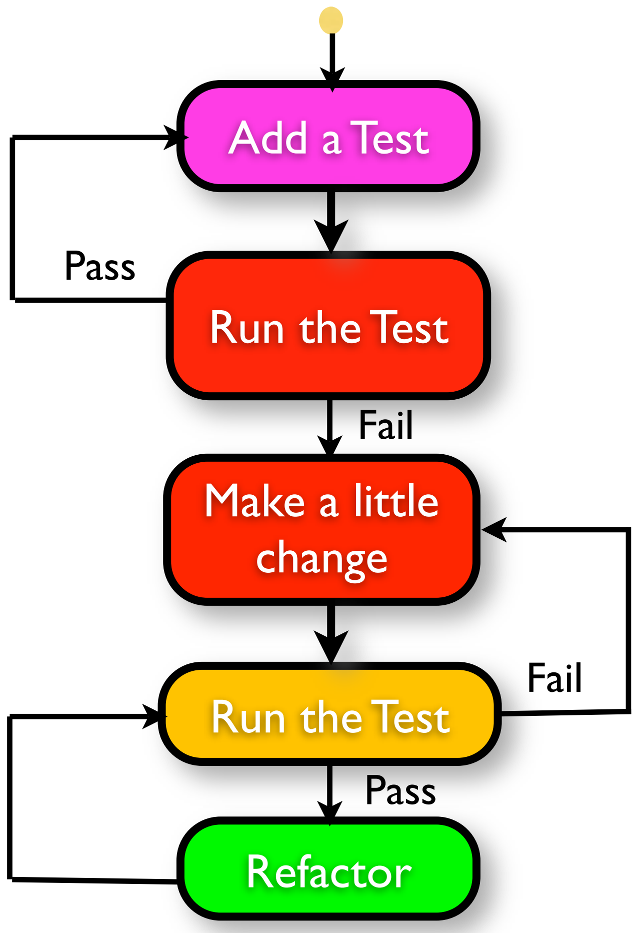
TDD Rhythm - Test, Code, Refactor



agile faqs

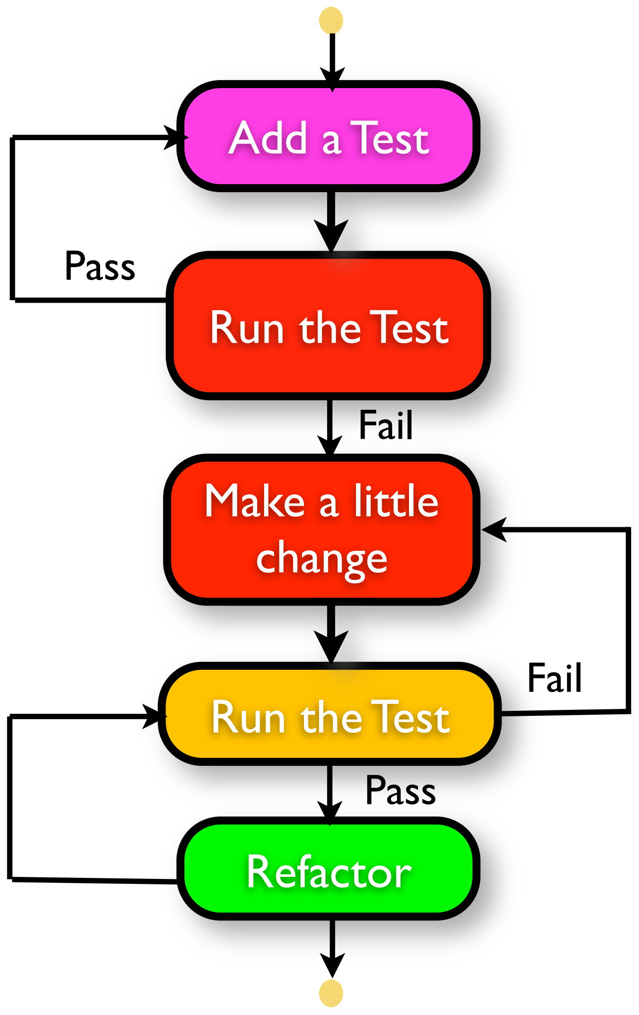
Test Driven Development

TDD Rhythm - Test, Code, Refactor



Test Driven Development

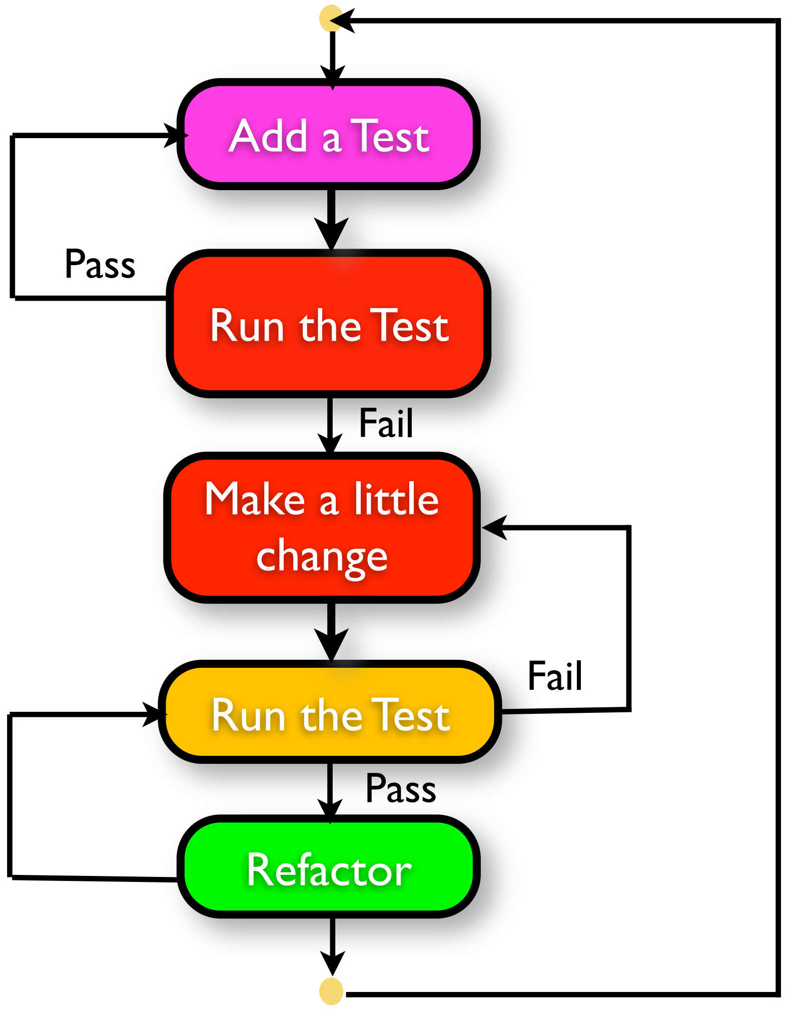
TDD Rhythm - Test, Code, Refactor



agile faqs

Test Driven Development

TDD Rhythm - Test, Code, Refactor



agile faqs

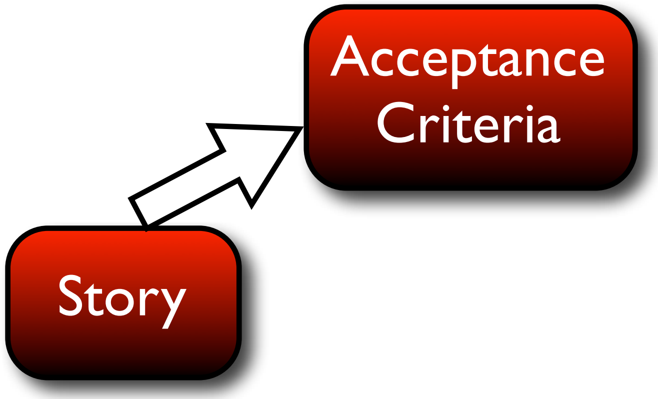
Acceptance Test Driven Development

Acceptance Test Driven Development

Story

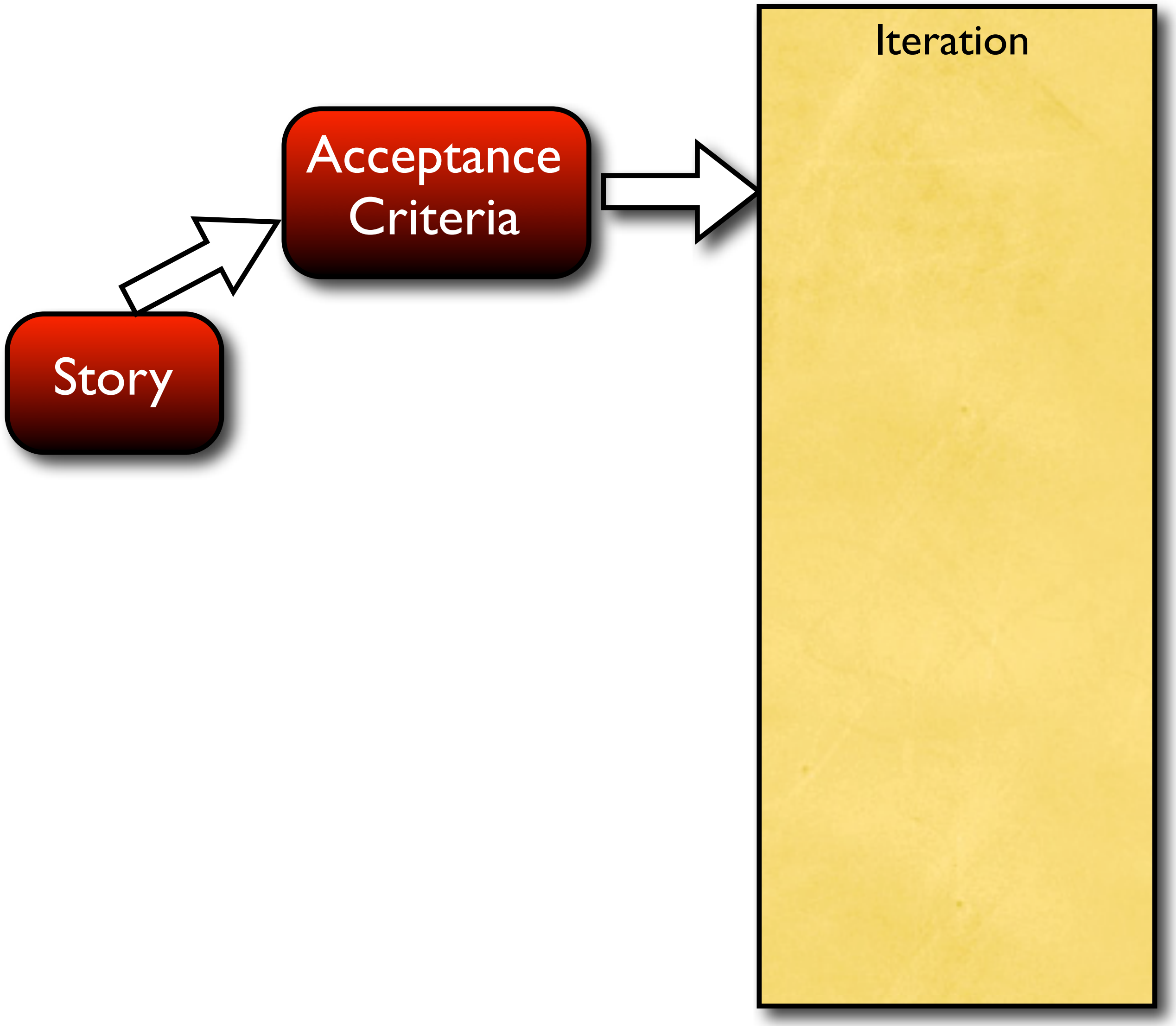
agile faqs

Acceptance Test Driven Development



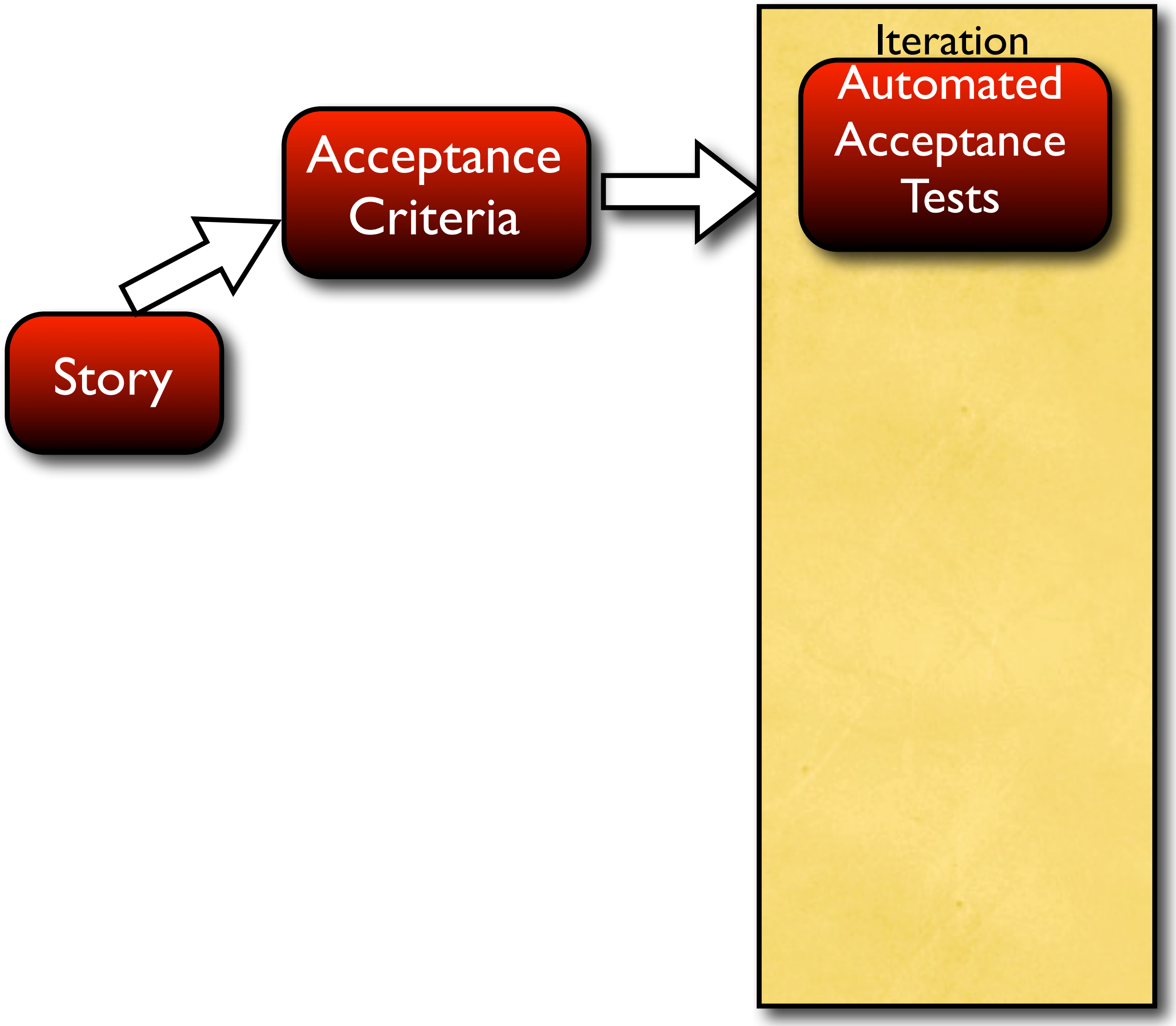
agile faqs

Acceptance Test Driven Development



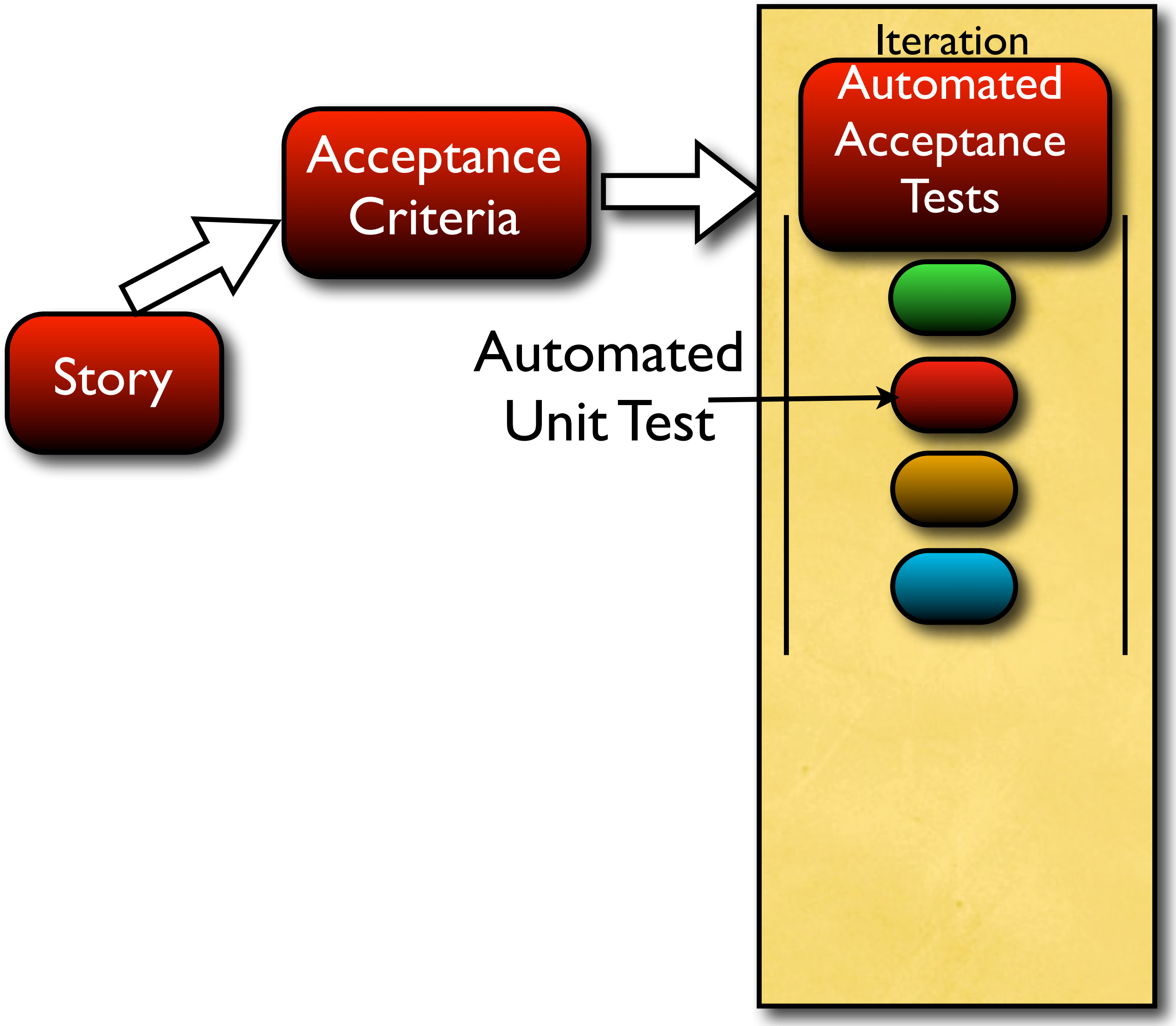
agile faqs

Acceptance Test Driven Development



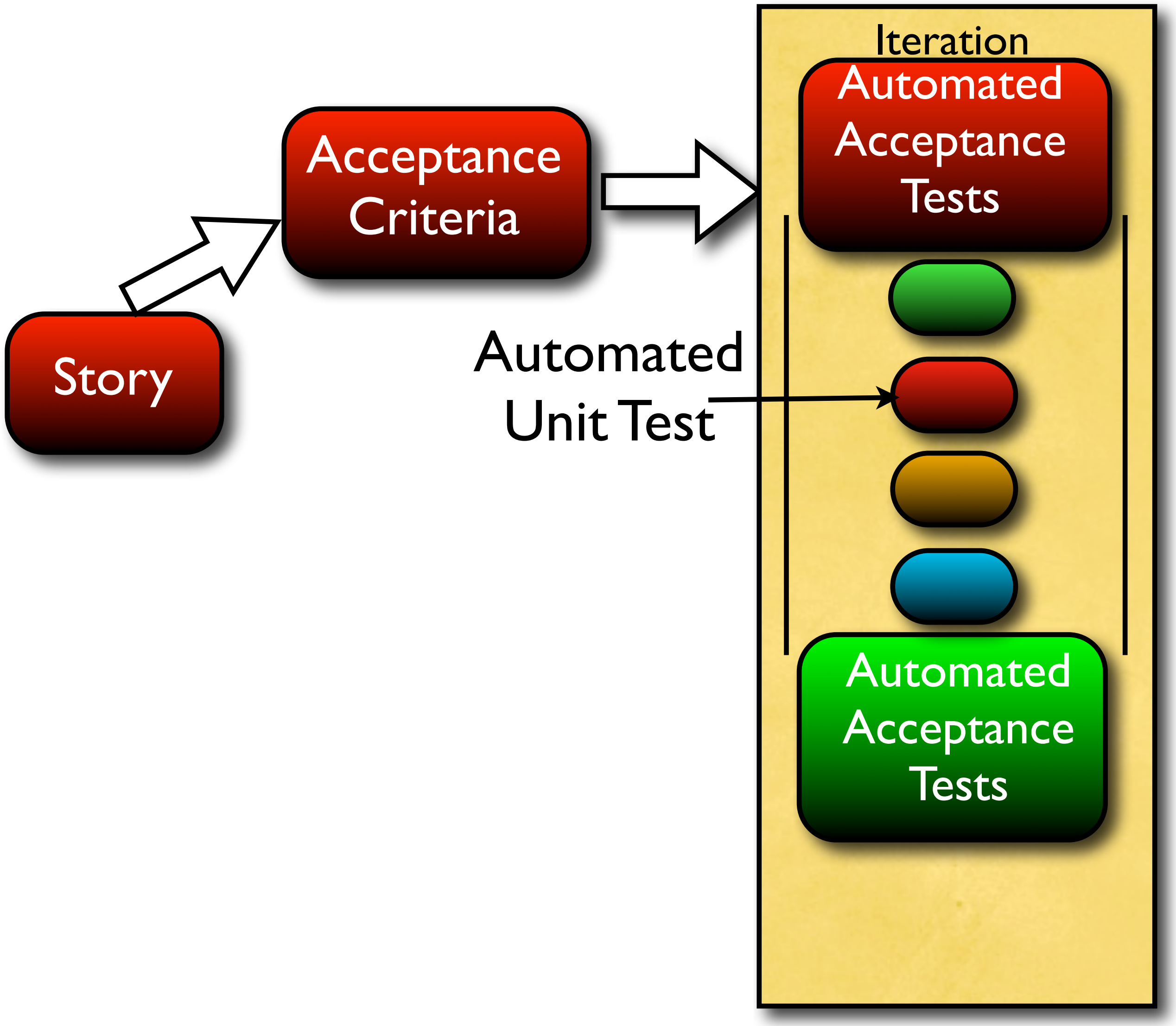
agile faqs

Acceptance Test Driven Development



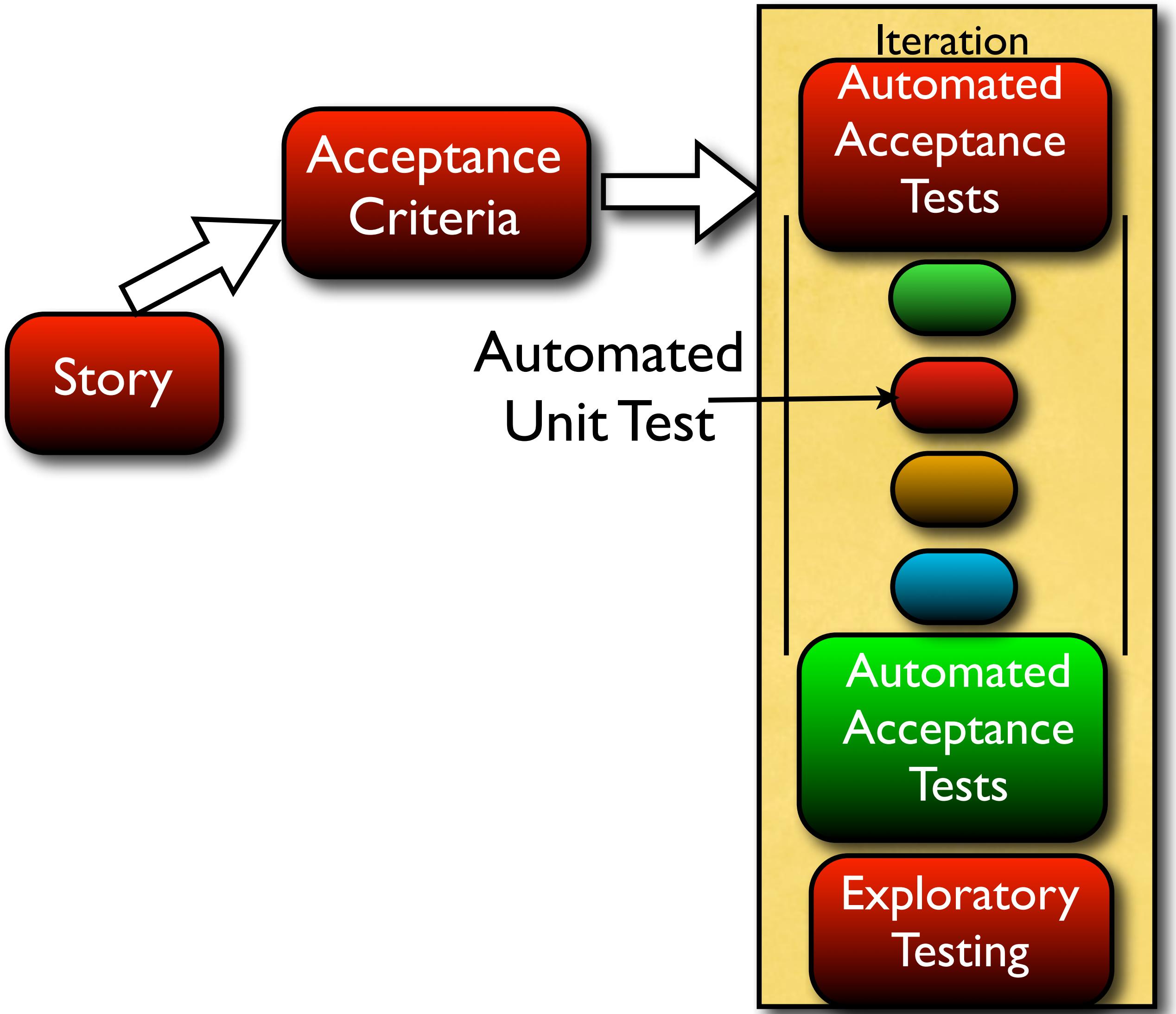
agile faqs

Acceptance Test Driven Development



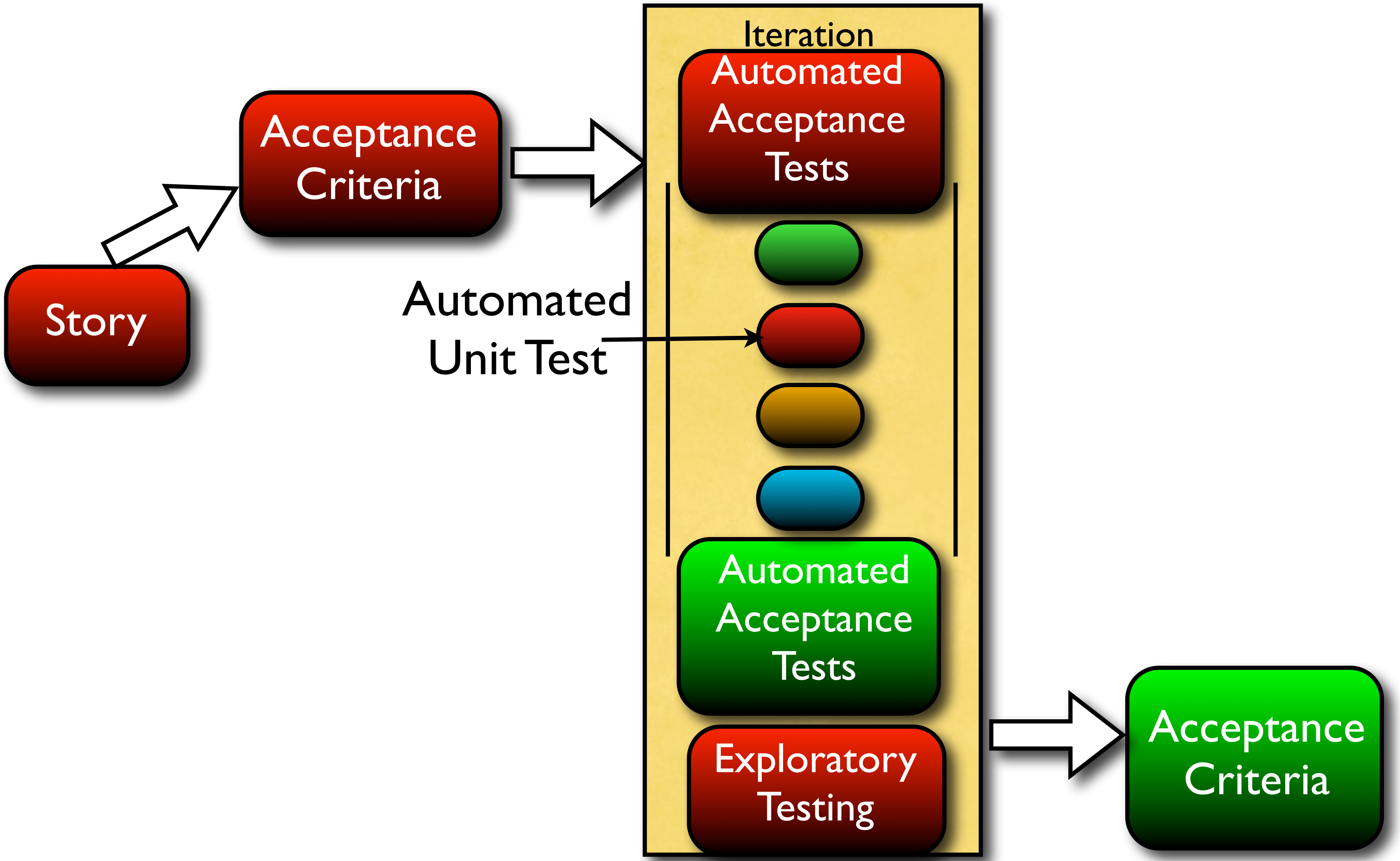
agile faqs

Acceptance Test Driven Development



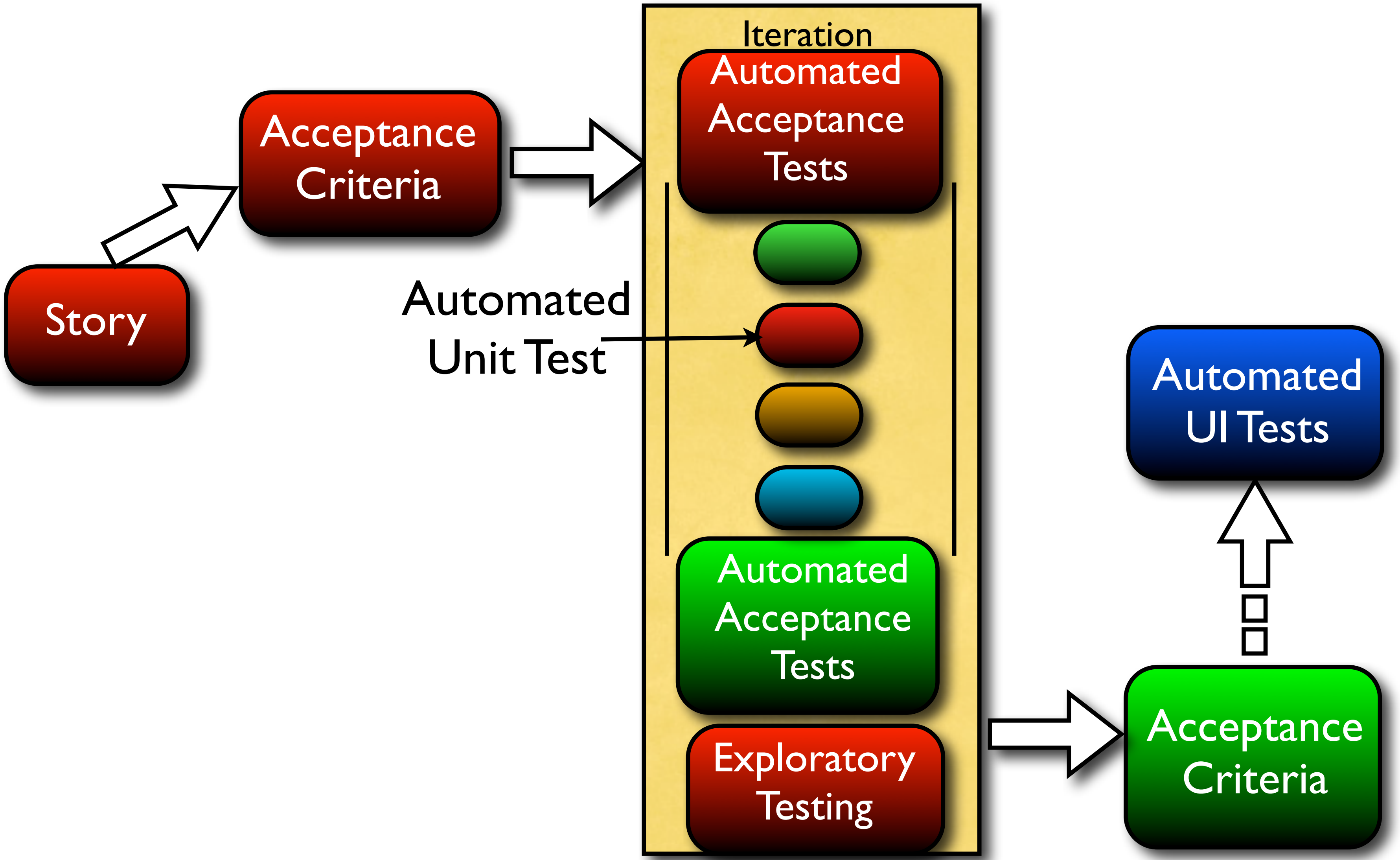
agile faqs

Acceptance Test Driven Development



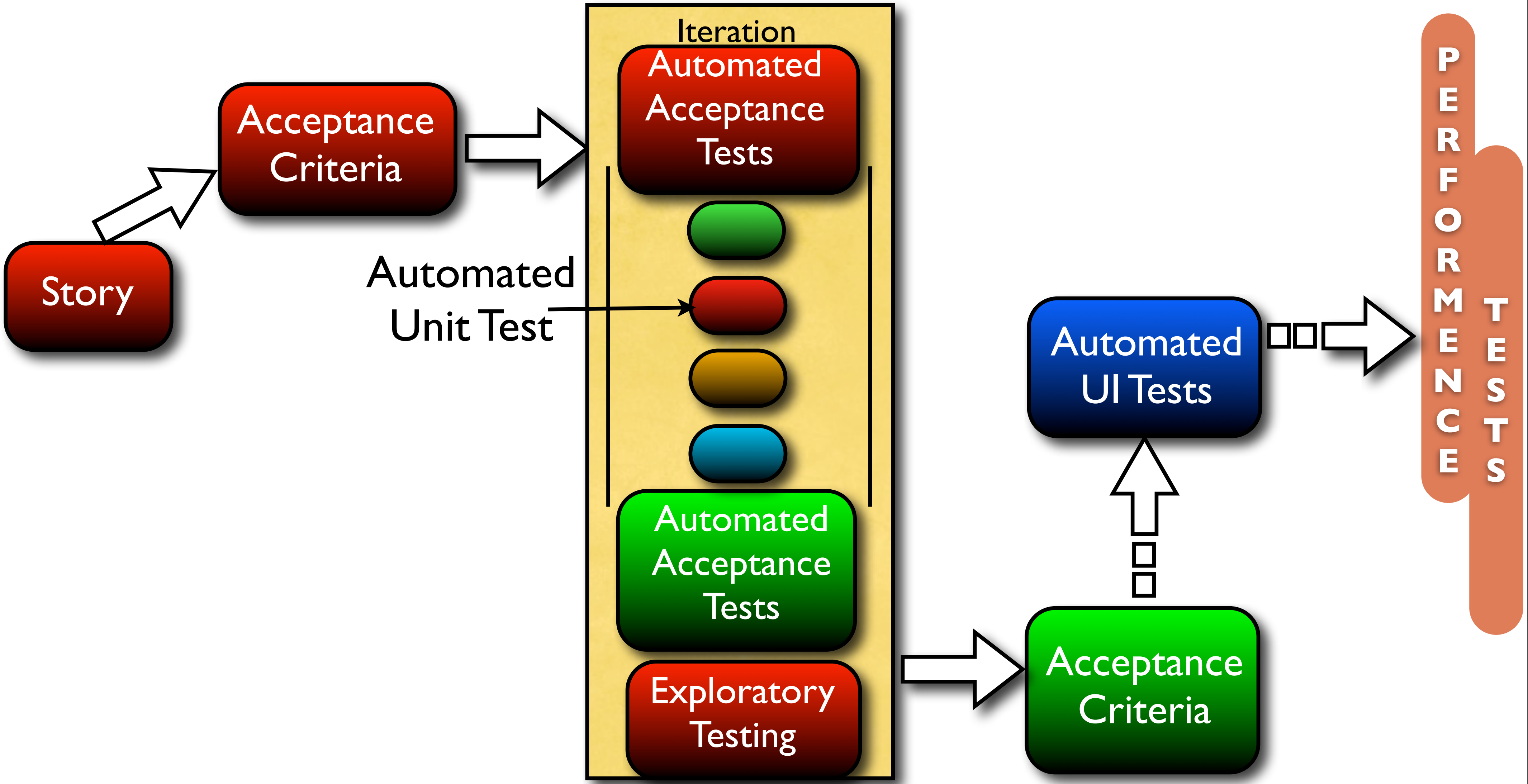
agile faqs

Acceptance Test Driven Development



agile faqs

Acceptance Test Driven Development

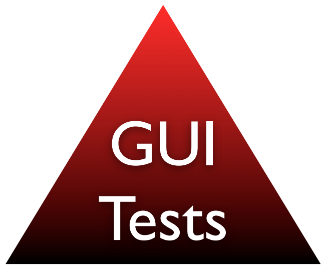


agile faqs

Mike Cohn's Testing Pyramid

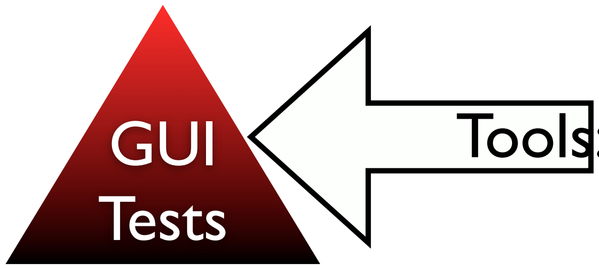
agile faqs

Mike Cohn's Testing Pyramid



agile faqs

Mike Cohn’s Testing Pyramid

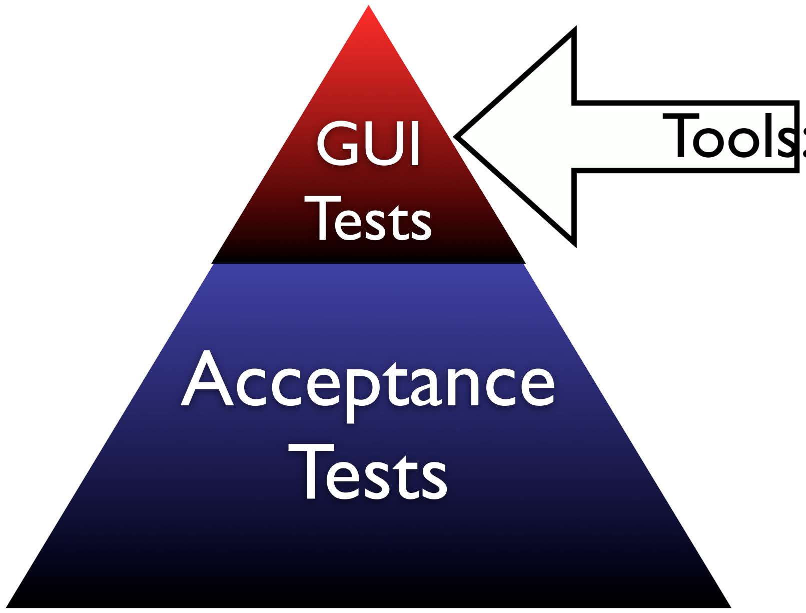


Small in Number

Tools: Selenium, Sahi, Watir, Abbot, Frankenstein

agile faqs

Mike Cohn's Testing Pyramid

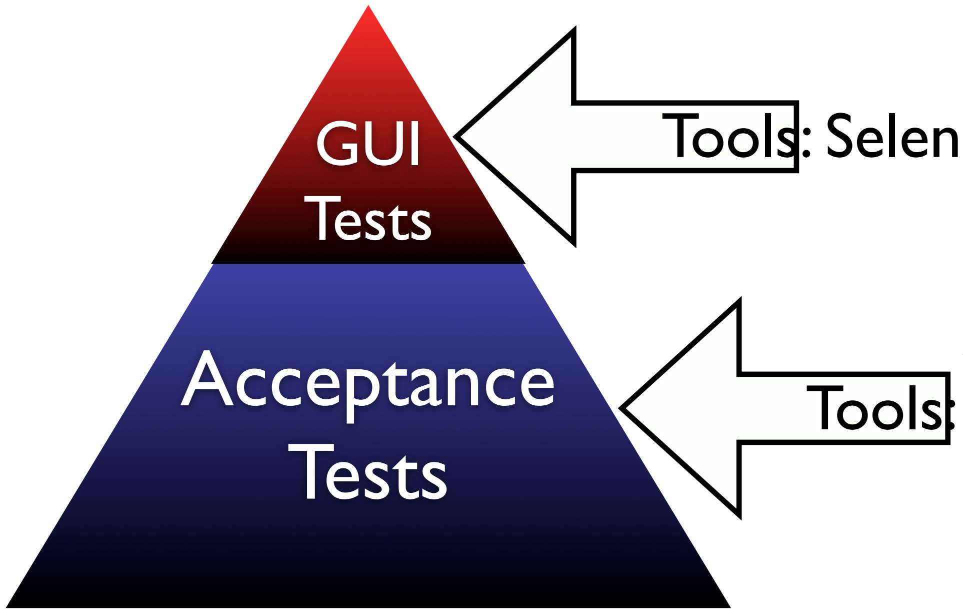


Small in Number

Tools: Selenium, Sahi, Watir, Abbot, Frankenstein

agile faqs

Mike Cohn's Testing Pyramid



Small in Number

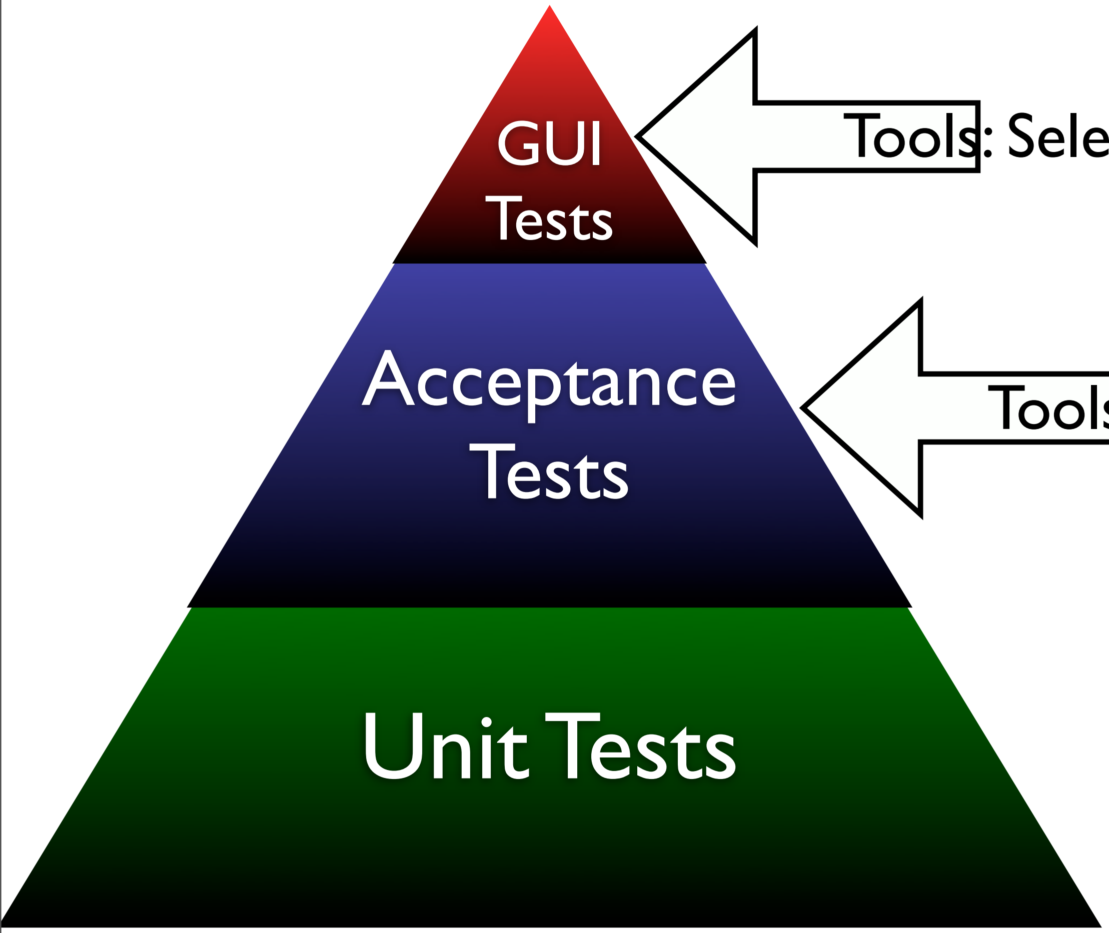
Tools: Selenium, Sahi, Watir, Abbot, Frankenstein

At least one per story

Tools: Fit, FitNesse, RSpec, JBehave

agile faqs

Mike Cohn's Testing Pyramid

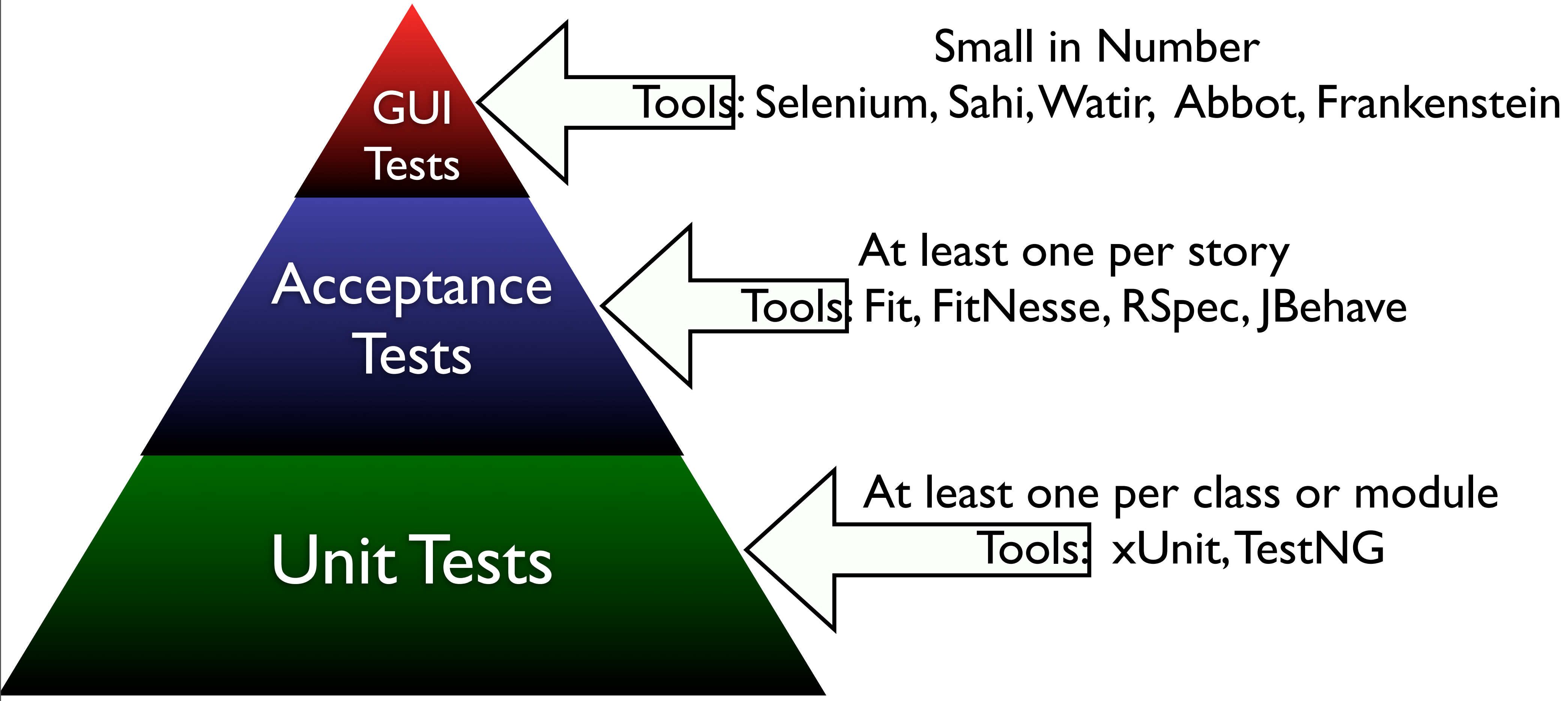


Small in Number
Tools: Selenium, Sahi, Watir, Abbot, Frankenstein

At least one per story
Tools: Fit, FitNesse, RSpec, JBehave

agile faqs

Mike Cohn's Testing Pyramid


GUI Tests
Acceptance Tests
Unit Tests
Small in Number
Tools: Selenium, Sahi, Watir, Abbot, Frankenstein
At least one per story
Tools: Fit, FitNesse, RSpec, JBehave
At least one per class or module
Tools: xUnit, TestNG

Criteria for DONE

- ☑ Every story must have at least one Acceptance Test
- ☑ A story is not DONE until it passes it's Acceptance Tests

Manual Acceptance Tests

Manual Acceptance Tests

Manual

Acceptance Tests

Manual Acceptance Tests

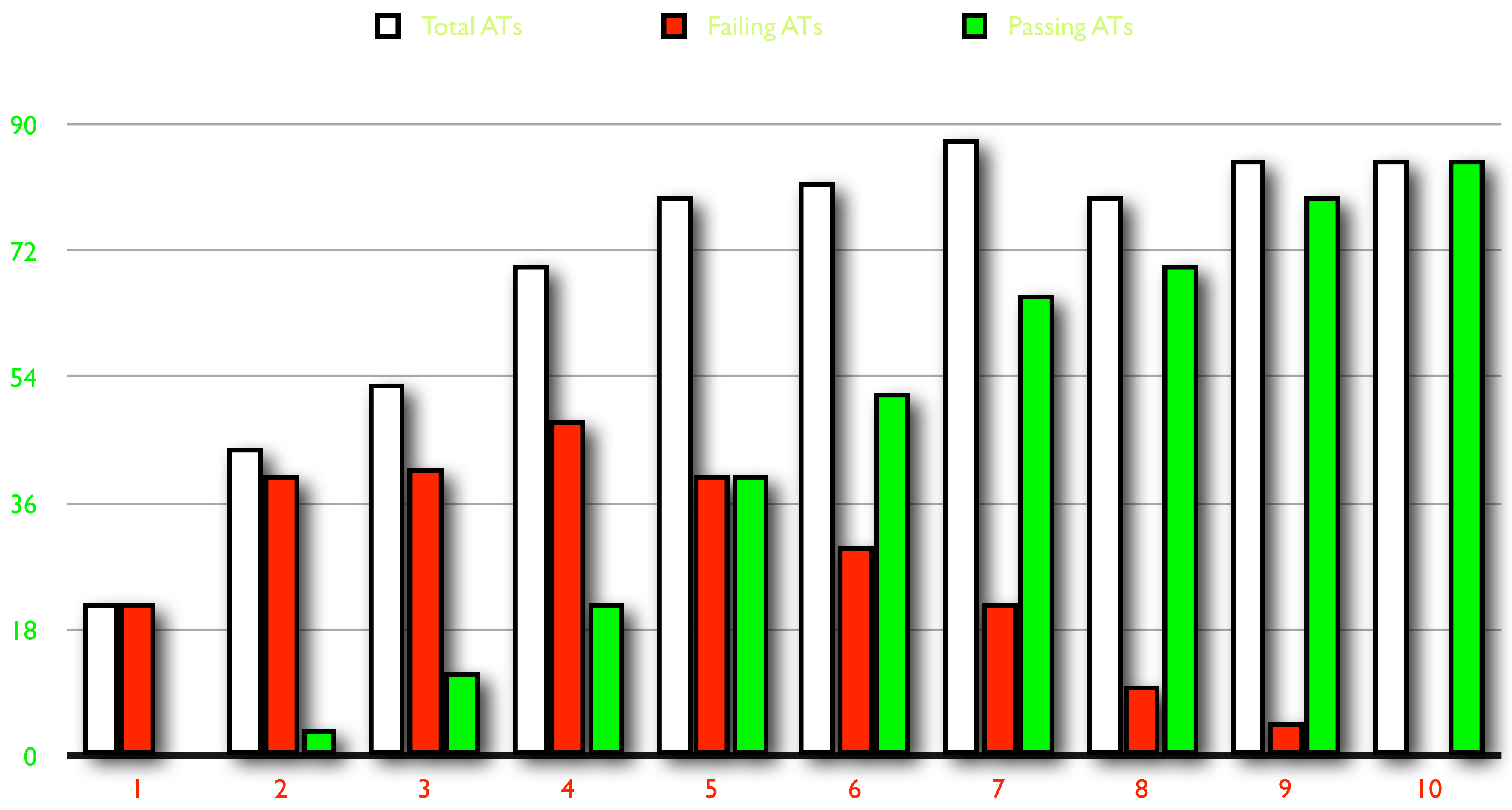
agile faqs

Why Acceptance Tests?

- ☑ Criteria for Completion
- ☑ Great Collaboration tool
- ☑ Source of Feedback
- ☑ Real data to measure progress

agile faqs

Data From Acceptance Tests



agile faqs

Acceptance Tests Are Automated

The Button

- ☑ How often would you press it?
- ☑ When would you press it?
- ☑ Who would press it?
  - ☑ Testers, Developers, Managers, Customers, Spectators, etc.

agile faqs

agile faqs

Criteria for DONE

agile faqs

Criteria for DONE

+

agile faqs

Criteria for DONE

+ Automated

agile faqs

Criteria for DONE

+ Automated

Executable Specification

Executable Specification

- ☑ A new paradigm for testing
- ☑ Puts quality first
- ☑ Removes ambiguity from requirements

agile faqs

Who Writes Acceptance Tests?

agile faqs

Who Writes Acceptance Tests?

The Customer

- ☑ The Customer Role
  - ☑ Stake holder
  - ☑ Business Analyst
  - ☑ Quality Assurance
  - ☑ Product Owner
  - ☑ Developer

Tests Get Technical

- ✓ The "Customer" may need technical help to write tests
- ✓ Developers and QAs are technical
- ✓ Pair test authoring

Business Rules Get Fuzzy

- ☑ Sometimes developers need help understanding tests
- ☑ Customers know business rules
- ☑ Pair test implementation

agile faqs

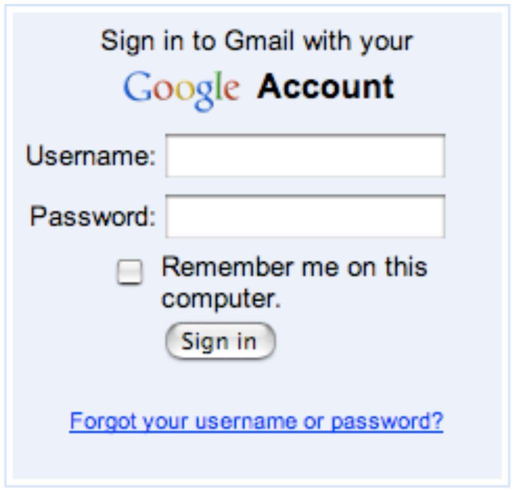
Exercise #1

agile faqs

Exercise #1

The Login Test

- ☑ Write a test plan, in plain text, for the business rules of logging in.
- ☑ Web application
- ☑ User credentials are stored in relational database
- ☑ Successful login redirects to "Welcome" page


Sign in to Gmail with your
Google Account
Username:
Password:
Remember me on this computer.
Sign in
Forgot your username or password?

agile faqs

Writing Good Acceptance Tests

Login Test Possibilities

1. Direct browser to URL for login page

Login Test Possibilities

~~1. Direct browser to URL for login page~~

Login Test Possibilities

1. ~~Direct browser to URL for login page~~

1. Enter the username ‘wallace’

Login Test Possibilities

1. ~~Direct browser to URL for login page~~
1. ~~Enter the username ‘wallace’~~

Login Test Possibilities

1. ~~Direct browser to URL for login page~~

1. ~~Enter the username 'wallace'~~

Build a Testable Environment First

Login Test Possibilities

1. Add some users to the system

Login Test Possibilities

1. ~~Add some users to the system~~

Login Test Possibilities

1. ~~Add some users to the system~~

3. Enter a value into the username field

Login Test Possibilities

1. ~~Add some users to the system~~

3. ~~Enter a value into the username field~~

Login Test Possibilities

1. ~~Add some users to the system~~
3. ~~Enter a value into the username field~~

Be Specific

Tests are Examples

- ☑ Use concrete examples
- ☑ Specify concrete behavior
- ☑ No ambiguity allowed

Login Test Possibilities

1. Insert into User table values ('wallace', 'ilikecheeze')

Login Test Possibilities

~~1. Insert into User table values ('wallace', 'ilikecheeze')~~

Login Test Possibilities

1. ~~Insert into User table values ('wallace', 'ilikecheeze')~~
2. Open a browser to the URL http://localhost/myapp

Login Test Possibilities

1. ~~Insert into User table values ('wallace', 'ilikecheeze')~~
2. ~~Open a browser to the URL http://localhost/myapp~~

Login Test Possibilities

1. ~~Insert into User table values ('wallace', 'ilikecheeze')~~
2. ~~Open a browser to the URL http://localhost/myapp~~

Avoid Implementation Details

agile faqs

Good Acceptable Criteria and Tests

Good Acceptable Criteria and Tests

☑ S - SPECIFIC - Explicitly defined and definite

Good Acceptable Criteria and Tests

- ✓ S - SPECIFIC - Explicitly defined and definite
- ✓ M - MEASURABLE - Possible to observe and quantify

agile faqs

Good Acceptable Criteria and Tests

- ✓ S - SPECIFIC - Explicitly defined and definite
- ✓ M - MEASURABLE - Possible to observe and quantify
- ✓ A - ACHIEVABLE - Capable of existing or taking place

agile faqs

Good Acceptable Criteria and Tests

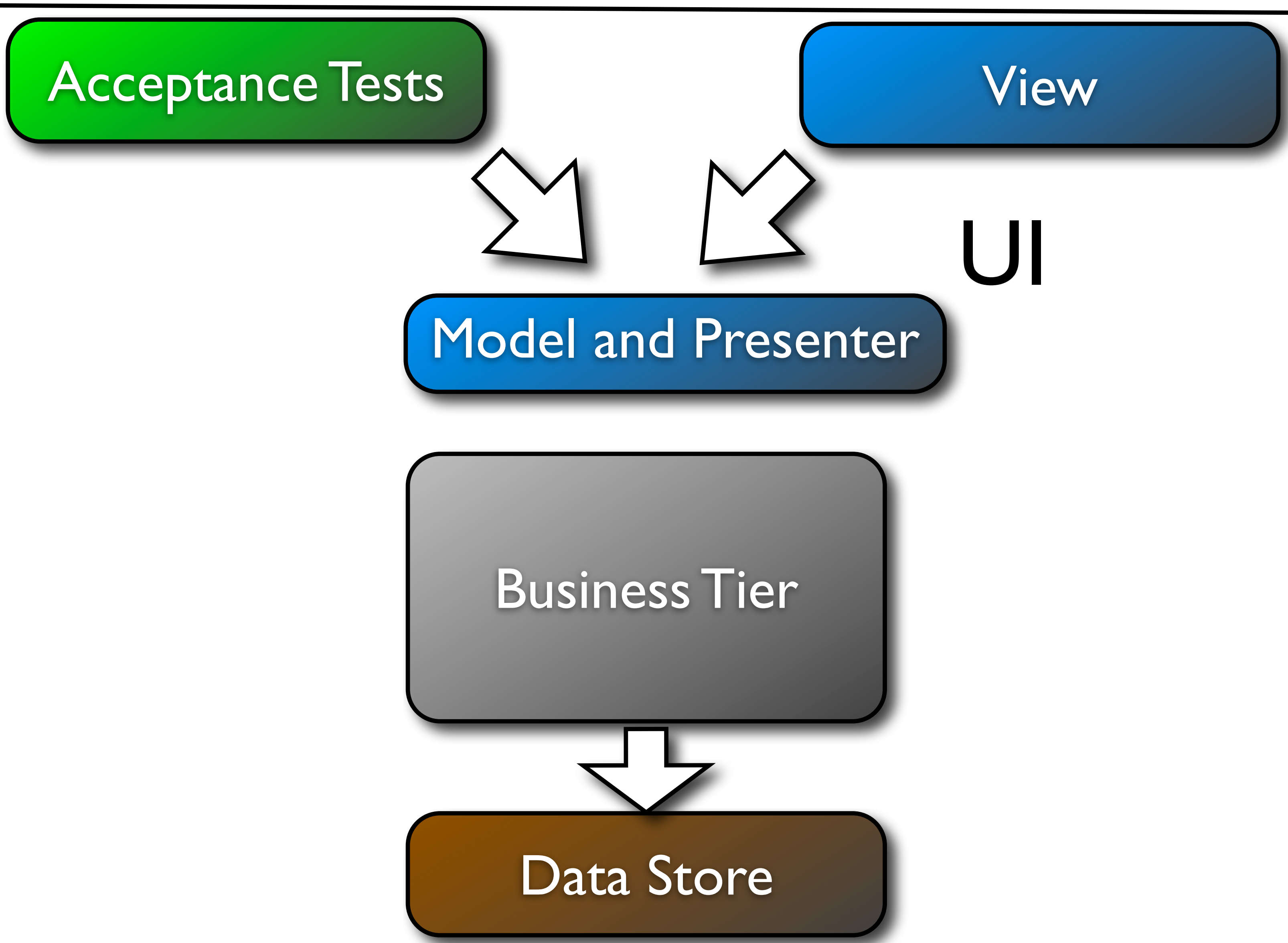
- ☑ S - SPECIFIC - Explicitly defined and definite
- ☑ M - MEASURABLE - Possible to observe and quantify
- ☑ A - ACHIEVABLE - Capable of existing or taking place
- ☑ R - RELEVANT - Having a connection with the story

agile faqs

Good Acceptable Criteria and Tests

- ✓ S - SPECIFIC - Explicitly defined and definite
- ✓ M - MEASURABLE - Possible to observe and quantify
- ✓ A - ACHIEVABLE - Capable of existing or taking place
- ✓ R - RELEVANT - Having a connection with the story
- ✓ T - TIME-BOUND – When will the outcome be observed

Avoid Implementation Details



Login Test: Possible Solution

- ☑ Add user to system: ('wallace', 'ilikecheeze')
- ☑ Process login with username 'wallace' and password 'blah'
- ☑ Check login failed
- ☑ Process login with username 'wallace' and password 'ilikecheeze'
- ☑ Check login succeeded

Tools

Commercial Tools

✓ WinRunner	TestPartner	EggPlant
✓ Silk	QTP	TestComplete
✓ RFT	Squish	WindowTester

Are not suitable for Acceptance Testing in an Agile environment

agile faqs

Open Source Options

☑ FIT	Sahi	Frankenstein
☑ FitNesse	Watir	Cucumber
☑ Selenium	Abbot	RSpec/JBehave

Among the few tools that support Test Driven Development

Wiki

What is it?

- ☑ A collaborative web site
- ☑ Editable by any
- ☑ Created by Ward Cunningham
- ☑ Every project should have one
- ☑ http://c2.com/wiki
- ☑ http://en.wikipedia.com

Creating Tests

- ☑ Use Wiki syntax to create a page with test tables
- ☑ Label the page as a Test Page
  - ☑ Use a page name of the form Test...
  - ☑ Turn on the Test property
- ☑ Make sure your Fixtures are in the classpath
  - ☑ Use !path widget
- ☑ Mechanics
  - ☑ !path values are concatenated
  - ☑ Java command to start FitServer is executed
  - ☑ Testable HTML is passed to FitServer
  - ☑ FitServer runs the tests
  - ☑ Results are passed back to FitNesse

Creating Suites

There are 2 ways to make Suites

- ☑ Set the Suite property
  - ☑ Create a page with the Suite property
  - ☑ Created test pages inside this page
  - ☑ When the suite is executed, all child test pages will be included in the suite execution
- ☑ Use the !see widget
  - ☑ !see <name of test page>
  - ☑ All "included" tests pages will be included in the suite execution

agile faqs

Hands-on Session

- ☑ Conference Proposal Submission Portal
- ☑ Some sample Stories
  - ☑ Should be able to submit new proposal
  - ☑ Should be able to list all submitted proposal
  - ☑ Submitting proposal with same title should display appropriate error message
  - ☑ Should be able to delete submitted proposal based on the title
  - ☑ Should be able to delete submitted proposal based on the title
  - ☑ Should be able to search proposals by title
  - ☑ Should be able to search proposals by ID
  - ☑ Should be able to find all proposal by an author's name

Break

agile faqs

Patterns

Patterns

Organizing Tests

- ☑ Allowing customers to add new tests without breaking the build

Version Control

☑ Keeping the acceptance test in version control with the code.

Cross-Functional Pairing

☑ Using FitNesse based acceptance tests for collaboration between cross-functional team members.

agile faqs

Patterns

ATDD

- ✓ Acceptance Test Driven Development

agile faqs

Patterns

CSTT

- ✓ Cleanup, Setup, Test, Teardown

Patterns

Independent Tests

- ☑ Tests shouldn't depend on each other.
- ☑ Tests leave the system in the same state it started in.

agile faqs

Patterns

Dynamic Stubbing

- ☑ Avoiding complications of external systems.

Non-Production Setup/Teardown

- ✓ Using non-production light weigh code for setup and teardown.
- ✓ Helps test only what you want to test.

Patterns

Suite Levels

- ☑ Creating different levels of suites depending on the depth/level of feedback desired.
- ☑ Smoke, Current Iteration/Sprint, Regression

agile faqs

Patterns

DRY

☑ Using !include to avoid repeating yourself.

Make it Real

☑ Write ATs as close as possible to the real environment.

Fixture Evolution

- ✓ Allow Fixture implementation to evolve over time.
- ✓ Treat fixtures as first class citizens.

At Least One Test/Story

- ☑ Every story should have at least one acceptance test
- ☑ Avoid long/multipurpose tests.

agile faqs

Anti-Patterns

Developer ATs

☑ Developers writing acceptance tests by themselves, for themselves.

Anti-Patterns

Unit Testing

- ☑ Don’t write ATs at the unit testing level
  - ☑ Unit tests are implementation specific
  - ☑ ATs are NOT implementation specific

QA Testing Tool

- ☑ Hard to write tests up front.
- ☑ Perhaps only on large projects.

agile faqs

Silver Bullet

- ☑ Trying to use FitNesse for all types of Acceptance Tests
  - ☑ UI testing
  - ☑ XML testing

Anti-Patterns

Test After

- ☑ Writing tests after the code is already written.

agile faqs

Anti-Patterns

Hidden Test Data

☑ Hiding test data in the fixtures.

Implementation Dependent ATs

- ☑ Making test pages (tables) dependent on implementation details and data structures.

Logging in Your Fixtures

- ☑ Putting log statements or print statements in the fixture code.
- ☑ Fixtures are probably too complicated.

agile faqs

Reference

- ✓ Portions of this presentation is adopted from Micah Martin's Introduction to Automated Acceptance Tests Presentation
- ✓ Kent Beck, Test Driven Development By Example.
- ✓ *"Agile Testing Directions" - Brian Marick*
- ✓ http://www.opensourcetesting.org/

agile faqs

The End

Questions?

naresh@agilefaqs.com
http://blogs.agilefaqs.com