Example Driven Architecture
Moving beyond the fragile test problem once and for all

Gerard Meszaros
Independent Consultant
CTO of FeedXL.Com
pnsqc2014@gerardm.com

My Background

- Software developer
- Development manager
- Project Manager
- Software architect
- OOA/OOD Mentor
- Requirements (Use Case) Mentor
- XP/TDD Mentor
- Agile PM Mentor
- Test Automation Consultant & Trainer
- Lean/Agile Coach/Consultant

These Slides: http://pnsqc2014.gerardm.com
http://pnsqc2014.gerardm.com
Copyright 2014 Gerard Meszaros
My Background

- Software developer
- Development manager
- Project Manager
- Software architect
- OOA/OOD Mentor
- Requirements (Use Case) Mentor
- XP/TDD Mentor
- Agile PM Mentor
- Test Automation Consultant & Trainer
- Lean/Agile Coach/Consultant

Agenda

- **Survey**
  - The State of Your Test Automation

- **Challenges of Test Automation**
  - Why is Test Automation So Hard

- **Rethinking the Development Process**
  - Example-Driven Development

- **Managing Scope vs Detail**
  - How to Define Effective Examples

- **Achieving a Test-Friendly Architecture**
  - Example-Driven Architecture
Functional Test Automation?

Please stand up if you are automating:
... tests for all user-visible functionality?
... tests for all business logic (but not the UI)?
... tests for key subset of business logic?

Please stand up if you are not automating any tests.

Unit Test Automation

Please sit down if you are:
... not automating any unit tests.

Please sit down if you are automating
... unit tests for core business logic only.
... unit tests for all business logic.
... unit tests for all code.
Effectiveness of Automation

Please stand up if your automated tests are:
...hard to prepare?
...difficult to understand?
...expensive to maintain?

Happiness With Test Automation?

Please sit down if you are:
• ...very unhappy with your test automation
• ...somewhat unhappy with your test automation.
• ...somewhat happy with your test automation.

So those of you still standing are very happy with your test automation??
Agenda

• Survey
  – The State of Your Test Automation

• Challenges of Test Automation
  – Why is Test Automation So Hard

• Rethinking the Development Process
  – Example-Driven Development

• Managing Scope vs Detail
  – How to Define Effective Examples

• Achieving a Test-Friendly Architecture
  – Example-Driven Architecture

So Why Is This So Hard To Do?

• Is it because we aren’t smart enough?
• Is it because the tools aren’t good enough?
• Is it because of factors beyond our control?

• Are we set up to fail?
Problems With Test Automation

- Tests are written separately from specification
- Tests are written separately from product code
- Functional Tests Interact with System via UI
- Tests are very detailed and repetitive
- Tests have to work around constraints of system and environment

Conway’s Law

- “The Architecture of System will resemble the Organization that Built it.”
Resulting Tests Are:

- **Forced to find sub-optimal ways to interact with the Product Code**
  - Via interfaces not designed for tests
  - Can only test the whole system, not the parts that contain the logic we want to test

- **Forced to find ways to control the state of the system**
  - Can’t set state directly; must use convoluted paths to set the state
Where Does This Leave Us?

- Tests are more complex than they should be
- Tests take longer to run
- Tests are fragile with frequent “false positive” test failures
- Tests require frequent, expensive maintenance

The Fragile Test Problem

- Kind of changes that break tests:
- Changes to the interface used to interact with system (e.g. UI)
- Changes to state of system (e.g. database contents)
- Changes to things system depends on:
  - Runtime container (user permissions, time/date)
  - Dependencies on other system (messaging, SQL, RMI)
Insanity: Doing the same thing over and over again and expecting different results.

Albert Einstein

---

**Agenda**

- **Survey**
  - The State of Your Test Automation

- **Challenges of Test Automation**
  - Why is Test Automation So Hard

- **Rethinking the Development Process**
  - Example-Driven Development

- **Managing Scope vs Detail**
  - How to Define Effective Examples

- **Achieving a Test-Friendly Architecture**
  - Example-Driven Architecture
This is not a technology problem!

- Our tools are very good
  - But they cannot overcome the obstacles
- Our people are very good
  - But they cannot overcome the obstacles

This is a System Design Problem

- We Have to Change the System we use to Build the Software
- To Remove the Obstacles
What Are the Obstacles?

• Business People:
  – … don’t know what developers need to be told
  – … don’t know what to ask for (what is possible)
  – … focus on success paths
• Developers:
  – … don’t know what would provide real value
  – … don’t understand testability requirements of system
• Testers and automaters:
  – … cannot communicate testability requirements
  – … are forced to use existing interfaces not designed for testing

What Are The Solutions

• T-B-D need to collaborate in preparing specifications before building the product.
  – T-B-D = Testers, Business & Development
• Specifications need to include concrete “potentially executable” examples at several levels, including
  – Workflow sequences
  – Transaction interactions
  – Business rules and algorithms
• Development needs to be responsible for automation of their execution
  – Directly from the business-readable specification
Example-Driven Development

- Business & Testers come up with examples; Developers help them formalize it.
- Developers write interpreters for examples; design product code to support them.
- Developers write product code guided by execution of the examples.

This is Like TDD but at a Very Large Scale
Example-Driven Development

• Runtime View:

- Potentially Executable Specifications
- Example Interpreter Code
- Product Code
- Specification Compliance Report

B,T,D → D,T

Example-Driven Development

• Feedback Loops:

- Potentially Executable Specification
- Example Interpreter Code
- Product Code
- Specification Compliance Report

How they say → What they say
Example-Driven Development

- Feedback Loops:

  - Potentially Executable Specifications
  - Example Interpreter
  - Product
  - Specification Compliance Report

Isn’t This “Just” Test Automation By Another Name?

Testing by An Another Name

Yes, it is “just” testing but naming is everything.

It changes our perceptions and expectations

Tests don’t need to be readable.

Examples and Specifications do!
Agenda

• Survey
  – The State of Your Test Automation

• Challenges of Test Automation
  – Why is Test Automation So Hard

• Rethinking the Development Process
  – Example-Driven Development

• Managing Scope vs Detail
  – How to Define Effective Examples

• Achieving a Test-Friendly Architecture
  – Example-Driven Architecture

Test Automation Pyramid

• Large numbers of very small unit tests
  – Ensures integrity of code

• Smaller number of functional tests for major components
  – Verify integration of units

• Even fewer tests for the entire application & workflow
  – Ensure application(s) support users’ requirements

• Tools to support effective exploratory testing

Pyramid originally proposed by Mike Cohn
Behavior Specification at Right Level

Need to specify at the right detail and scope.

- High Detail
  - Unit Tests
  - Component Tests
  - System Tests

- Low Detail
  - Component Tests
  - System Tests

Scope: Broad to Narrow
Detail: High to Low

http://pnsqc2014.gerardm.com
Copyright 2014 Gerard Meszaros
Behavior Specification at Right Level

- Specify broad scope at minimum detail
- Specify most detailed req'ts at narrowest scope

Multi-Use Case Workflows  Incomplete Spec

Transactions (Use Cases)  Rules & Algorithms

Use Cases & User Stories

- Configure Notification Rules
- Suspend Notification
- Resume Notification
- Process Transaction
- Notify based on charge Type
- Notify based on charge Continent
- Notify based on charge Country
- Notify based on charge City & State
- Notify based on change City & State
- Transaction Settlement

http://pnsqc2014.gerardm.com
Example: Testing Notifications - 1

Given:
User and Accounts

When:
Notification Rule is Configured

Then:
Notification Rule is Active

Example: Testing Notifications - 2

Use Case:
Process Transactions

When: The Transactions to be processed

Then: Expected Notifications
Issues With This Test

• **Difficult to understand which TX’s should notify**
  – because cause (rules) and effect (notification) are far apart

• **Only verifies one simple combination of rules**
  – We will require many more tests to test all the other combinations
  – Lots of repetition of workflow & data across test cases

• **Simplest workflow;**
  – More complex workflows will be even longer and harder to understand

• **Tests will take a long time to run**
  – Due to need to configure first, then run transaction processing

How Can We Improve Tests?

• **Make tests shorter**
  – Fewer steps

• **Make tests less detailed**
  – Omit unnecessary information

• **Focus tests on specific aspects of behaviour**
  – Test/Spec Algorithms & Rules directly
  – Test less functionality, but more thoroughly

Fine advice, but too vague
Difficult without Test-Friendly Architecture
How Can Spec. Mindset Help

- **Describe overall workflow**
  - with much less detail
- **Provide more details**
  - For individual transactions
- **Provide even more details**
  - For business rules.
- **But not embedded in the workflow**

---

Changing Level of Abstraction/Detail

- **Need to Reduce Detail or Reduce Scope**

- **High Detail, Broad Scope**
  - Rules & Algorithms
  - Transactions (Use Cases)
  - Multi-Use Case Workflows

- **Low Detail, Narrow Scope**
  - Incomplete Spec

- **High Detail, Narrow Scope**
  - Unmaintainable

- **Low Detail, Broad Scope**
  - Reduce Scope a Lot!
  - Reduce Detail a Lot!
Omitting Unnecessary Information

- Ask not what you can include in an example...

- Ask instead:

- What can I leave out of this example?

“If it isn’t essential to conveying the essence of the behavior, it is essential to not include it.”

Example:

**Specifying Workflow**

<table>
<thead>
<tr>
<th>Customer</th>
<th>User and Accounts</th>
</tr>
</thead>
</table>

System lists all available accounts for the authorized customer

<table>
<thead>
<tr>
<th>account</th>
<th>type</th>
<th>notification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10035692877</td>
<td>checking</td>
<td>disabled</td>
</tr>
<tr>
<td>10035692898</td>
<td>savings</td>
<td>disabled</td>
</tr>
<tr>
<td>20039928979</td>
<td>credit line</td>
<td>disabled</td>
</tr>
</tbody>
</table>

Customer sets notification threshold for all transactions from all locations to $10,000.00 on account 10035692877 via email to bob@me.com

Ensure no system issues

Ensure system log contains: Customer below set notification threshold for all transactions from all locations to $10,000.00 on account 10035692877

System lists all available accounts for the authorized customer

<table>
<thead>
<tr>
<th>account</th>
<th>type</th>
<th>notification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10035692877</td>
<td>checking</td>
<td>disabled</td>
</tr>
<tr>
<td>10035692898</td>
<td>savings</td>
<td>disabled</td>
</tr>
<tr>
<td>20039928979</td>
<td>credit line</td>
<td>disabled</td>
</tr>
</tbody>
</table>

Notification settings for account 10035692877

<table>
<thead>
<tr>
<th>transaction type</th>
<th>transaction is initiated</th>
<th>threshold amount</th>
<th>via</th>
<th>address</th>
</tr>
</thead>
<tbody>
<tr>
<td>all</td>
<td>all</td>
<td>$10,000.00</td>
<td>email <a href="mailto:bob@me.com">bob@me.com</a></td>
<td></td>
</tr>
</tbody>
</table>

Given: User and Accounts

When: Notification Rule is Configured

Then: Notification Rule is Active
Example:

**Specifying Workflow**

<table>
<thead>
<tr>
<th>Time now is</th>
<th>9:30AM, 03/18/2008</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bank processes</td>
<td>debit to 10035692877 in the amount of $15,000.00</td>
</tr>
<tr>
<td>Bank processes</td>
<td>debit to 10035692877 in the amount of $9,000.00</td>
</tr>
<tr>
<td>Bank processes</td>
<td>debit to 10035692877 in the amount of $11,000.00</td>
</tr>
<tr>
<td>Bank processes</td>
<td>credit to 10035692877 in the amount of $9,999.99</td>
</tr>
<tr>
<td>Bank processes</td>
<td>credit to 10035692877 in the amount of $9,999.99</td>
</tr>
<tr>
<td>Bank processes</td>
<td>charge to 10035692877 in the amount of $11,000.00</td>
</tr>
</tbody>
</table>

New notifications sent to customer bobma

<table>
<thead>
<tr>
<th>type</th>
<th>account</th>
<th>timestamp</th>
<th>amount</th>
<th>email</th>
</tr>
</thead>
<tbody>
<tr>
<td>debit</td>
<td>10035692877</td>
<td>9:30AM, 03/18/2012</td>
<td>$15,000.00</td>
<td><a href="mailto:bobma@live.com">bobma@live.com</a></td>
</tr>
<tr>
<td>debit</td>
<td>10035692877</td>
<td>9:30AM, 03/18/2012</td>
<td>$11,000.00</td>
<td><a href="mailto:bobma@live.com">bobma@live.com</a></td>
</tr>
<tr>
<td>credit</td>
<td>10035692877</td>
<td>9:30AM, 03/18/2012</td>
<td>$13,000.00</td>
<td><a href="mailto:bobma@live.com">bobma@live.com</a></td>
</tr>
<tr>
<td>charge</td>
<td>10035692877</td>
<td>9:30AM, 03/18/2012</td>
<td>$11,000.00</td>
<td><a href="mailto:bobma@live.com">bobma@live.com</a></td>
</tr>
</tbody>
</table>

Use Case: Process Transactions

When: The Transactions to be processed

Then: Expected Notifications
Example:

**Specifying Workflow**

Time now is 9:30AM, 03/18/2008

- Bank processes debit to 10035692877 in the amount of $15,000.00
- Bank processes debit to 10035692877 in the amount of $9,000.00
- Bank processes debit to 10035692877 in the amount of $11,000.00

New notifications sent to customer bobma

<table>
<thead>
<tr>
<th>type</th>
<th>account</th>
<th>timestamp</th>
<th>amount</th>
<th>via</th>
<th>address</th>
</tr>
</thead>
<tbody>
<tr>
<td>debit</td>
<td>10035692877</td>
<td>9:30AM, 03/18/2012</td>
<td>$15,000.00</td>
<td>email</td>
<td>bobmaxlive.com</td>
</tr>
<tr>
<td>debit</td>
<td>10035692877</td>
<td>9:30AM, 03/18/2012</td>
<td>$11,000.00</td>
<td>email</td>
<td>bobmaxlive.com</td>
</tr>
</tbody>
</table>

Example:

**Specifying Workflow**

Customer sets notification threshold for all transactions from 10,000.00 to 50,000.00 on account 10035692877

Time now is 9:30AM, 03/18/2008

- Bank processes debit to 10035692877 in the amount of $15,000.00
- Bank processes debit to 10035692877 in the amount of $9,000.00
- Bank processes debit to 10035692877 in the amount of $11,000.00

New notifications sent to customer bobma

<table>
<thead>
<tr>
<th>type</th>
<th>account</th>
<th>timestamp</th>
<th>amount</th>
<th>via</th>
<th>address</th>
</tr>
</thead>
<tbody>
<tr>
<td>debit</td>
<td>10035692877</td>
<td>9:30AM, 03/18/2012</td>
<td>$15,000.00</td>
<td>email</td>
<td>bobmaxlive.com</td>
</tr>
<tr>
<td>debit</td>
<td>10035692877</td>
<td>9:30AM, 03/18/2012</td>
<td>$11,000.00</td>
<td>email</td>
<td>bobmaxlive.com</td>
</tr>
</tbody>
</table>
Example:

**Specifying Notification Workflow**

Given: User & Thresholds

<table>
<thead>
<tr>
<th>Time now is</th>
<th>9:00AM, 03/18/2008</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customer</td>
<td>bobma</td>
</tr>
<tr>
<td></td>
<td>sets notification threshold to $10,000.00 for all transactions to 10035692877</td>
</tr>
</tbody>
</table>

New notifications sent to customer bobma

<table>
<thead>
<tr>
<th>type</th>
<th>account</th>
<th>timestamp</th>
<th>amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>credit</td>
<td>10035692877</td>
<td>9:30AM, 03/18/2008</td>
<td>$15,000.00</td>
</tr>
<tr>
<td>credit</td>
<td>10035692877</td>
<td>9:30AM, 03/18/2008</td>
<td>$11,000.00</td>
</tr>
</tbody>
</table>

Example:

**Specifying Notification Workflow**

Given: User & Thresholds

<table>
<thead>
<tr>
<th>Time now is</th>
<th>9:00AM, 03/18/2008</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customer</td>
<td>bobma</td>
</tr>
<tr>
<td></td>
<td>sets notification threshold to $10,000.00 for all transactions to 10035692877</td>
</tr>
</tbody>
</table>

New notifications sent to customer bobma

<table>
<thead>
<tr>
<th>type</th>
<th>account</th>
<th>timestamp</th>
<th>amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>credit</td>
<td>10035692877</td>
<td>9:30AM, 03/18/2008</td>
<td>$15,000.00</td>
</tr>
<tr>
<td>credit</td>
<td>10035692877</td>
<td>9:30AM, 03/18/2008</td>
<td>$11,000.00</td>
</tr>
</tbody>
</table>
Specifying Notification Workflow

Example:

**Given:** User & Thresholds

**When:** Transactions Are Processed

**Then:** We Expect Notifications

---

**Specifying Suspension Workflow**

**Given** BobMa has set notification threshold to $10,000 for all transactions on his account

**When** the bank processes debit for 15,000 to his account

**Then** BobMa receives notification for debit 15,000

**When** BobMa suspends notification on his account

**And** the bank processes debit for 15,000 to his account

**Then** BobMa receives no notification

**When** BobMa resumes notification on his account

**And** the bank processes debit for 15,000 to his account

**Then** BobMa receives notification for debit 15,000
**Specifying Suspension Workflow**

Given BobMa has set notification threshold to $10,000 for all transactions on his account. When the bank processes debit for 15,000 to his account Then BobMa receives notification for debit 15,000

When BobMa suspends notification on his account And the bank processes debit for 15,000 to his account Then BobMa receives no notification

When BobMa resumes notification on his account And the bank processes debit for 15,000 to his account Then BobMa receives notification for debit 15,000

**Specifying Suspension Workflow**

Given BobMa has set up notification for all transactions on his account. When the bank processes an over-threshold-tx to his account Then BobMa receives notification for that tx

When BobMa suspends notification on his account And the bank processes an over-threshold-tx to his account Then BobMa receives no notification for that tx

When BobMa resumes notification on his account And the bank processes an over-threshold-tx to his account Then BobMa receives notification for that tx
Specifying Transaction Details

• So, where should omitted details go?

Rules & Algorithms

Multi-Use Case Workflows

Incomplete Spec

Transactions (Use Cases)

Inadequate

Broad

Narrow

Detail

Scope

High

Low

Specifying Transaction Details

Example:

Single Use Case Test

Use Case: Manage Notifications

Given: User & Accounts

When: Customer sets up threshold

Then: We Expect Logging & Updated Statuses

Medium Scope (Single Actor)

Medium Detail (Transaction, not UI):
Specifying Business Rules

- Need the more detail; need to reduce scope

**Diagram:**

- Multi-Use Case Workflows
- Transactions (Use Cases)
- Rules & Algorithms

**Example:**

### Business Rule Spec

**Threshold per Charge Type**

<table>
<thead>
<tr>
<th>Configuration</th>
<th>Process Transaction</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Configuration</strong></td>
<td><strong>Process Transaction</strong></td>
</tr>
<tr>
<td>Customer Accounts</td>
<td>Notification Required</td>
</tr>
<tr>
<td>Customer</td>
<td>Account</td>
</tr>
<tr>
<td>bobma</td>
<td>100372</td>
</tr>
<tr>
<td><strong>Customer Thresholds</strong></td>
<td></td>
</tr>
<tr>
<td>Customer</td>
<td>Account</td>
</tr>
<tr>
<td>bobma</td>
<td>100372</td>
</tr>
<tr>
<td>bobma</td>
<td>100372</td>
</tr>
<tr>
<td>bobma</td>
<td>100372</td>
</tr>
<tr>
<td>bobma</td>
<td>100372</td>
</tr>
</tbody>
</table>
Example:

Business Rule Spec
Threshold per Charge Type

Configuration

<table>
<thead>
<tr>
<th>CustomerAccount</th>
<th>Charge Type</th>
<th>Threshold</th>
<th>Added()</th>
</tr>
</thead>
<tbody>
<tr>
<td>bobma</td>
<td>100372</td>
<td>Checking</td>
<td>OK</td>
</tr>
<tr>
<td>bobma</td>
<td>100372 Travel</td>
<td>1,000</td>
<td>OK</td>
</tr>
<tr>
<td>bobma</td>
<td>100372 Restaurant</td>
<td>100</td>
<td>OK</td>
</tr>
<tr>
<td>bobma</td>
<td>100372 Groceries</td>
<td>264.23</td>
<td>OK</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Process Transaction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account</td>
</tr>
<tr>
<td>----------</td>
</tr>
<tr>
<td>100372</td>
</tr>
<tr>
<td>100372</td>
</tr>
<tr>
<td>100372</td>
</tr>
<tr>
<td>100372</td>
</tr>
<tr>
<td>100372</td>
</tr>
<tr>
<td>100372</td>
</tr>
<tr>
<td>100372</td>
</tr>
<tr>
<td>100372</td>
</tr>
</tbody>
</table>

Narrow Scope (Single Rule)
High Detail (Everything that matters)

When we ask NotificationRequired? with this transaction:

Then: The answer should be

Agenda

• Survey
  – The State of Your Test Automation

• Challenges of Test Automation
  – Why is Test Automation So Hard

• Rethinking the Development Process
  – Example-Driven Development

• Managing Scope vs Detail
  – How to Define Effective Examples

• Achieving a Test-Friendly Architecture
  – Example-Driven Architecture
Test-After Architecture

- Must test through User Interface
- Must load lots of data

Example-Driven Architecture

- Need to provide API’s to invoke functionality
- Bypassing the UI
- Supports workflow and use case / transaction example automation
EDA – Single Tx Examples

• Same API supports Transaction (Use Case) examples:
  - Configuration User Interface
  - Notification Rule Example
  - Notification Method Example
  - System Under Test
  - Configure Notification Threshold
  - Process Transaction
  - Should we Notify?
  - Do Notification.

EDA – Business Rules

• Encapsulation of Rules in “Rules Component”
  - which can be accessed directly via the Test
  - Configuration User Interface
  - Notification Rule Example
  - Notification Method Example
  - System Under Test
  - Configure Notification Threshold
  - Process Transaction
  - Should we Notify?
  - Do Notification.

http://pnsqc2014.gerardm.com
EDA – Business Rules

• Encapsulation of Rules in “Rules Component”
  • which can be accessed directly via the Test

• Component Accepts Rules via “Data Injection”
  – Simplify Automation by avoiding need to configure

• Ensures Design-for-testability
  – But Only When Automation is a Dev’t Responsibility

Quick Demo
Summary of Approach

• Business & Testers come up with examples; Developers help them formalize it.
  – Ensures everyone agrees on what needs to be built

• Manage Scope vs. Detail in examples
  » Workflow sequences
  » Transaction interactions
  » Business rules and algorithms
  – Avoids long, hard-to-understand examples

Summary of Approach

• Developers write interpreters for examples; design product code to support them
  – Ensures the architecture supports automation

• Developers write product code guided by execution of the examples
  – They know what needs to be built
  – They can run the examples frequently to know how they are doing
Summary of Benefits

- Examples force collaboration between all project roles.
- Built-in automation helps avoid Fragile Tests
- Automated Examples help us prevent defects being introduced through misunderstanding of requirements
- Automated Examples ensure complete “test” coverage of functionality

Closing Thoughts

- Are you automating to find defects or prevent them?
- Are your automated tests good examples?
  - Why not? What would you need to change?
- Are your tests low maintenance?
  - Why not? What causes them to break?
  - What could you change…
  - … to make them break less often?
  - …. to reduce the impact of breakage?
Thank You!
Gerard Meszaros
pnsqc2014@gerardm.com
http://www.xunitpatterns.com

Slides: http://PNSQC2014.gerardm.com

Call me when you:
• Want to transition to Agile or Lean
• Want to do Agile or Lean better
• Want to teach developers how to test
• Need help with test automation strategy
• Want to improve your test automation

Jolt Productivity Award winner - Technical Books

Available on MSDN: