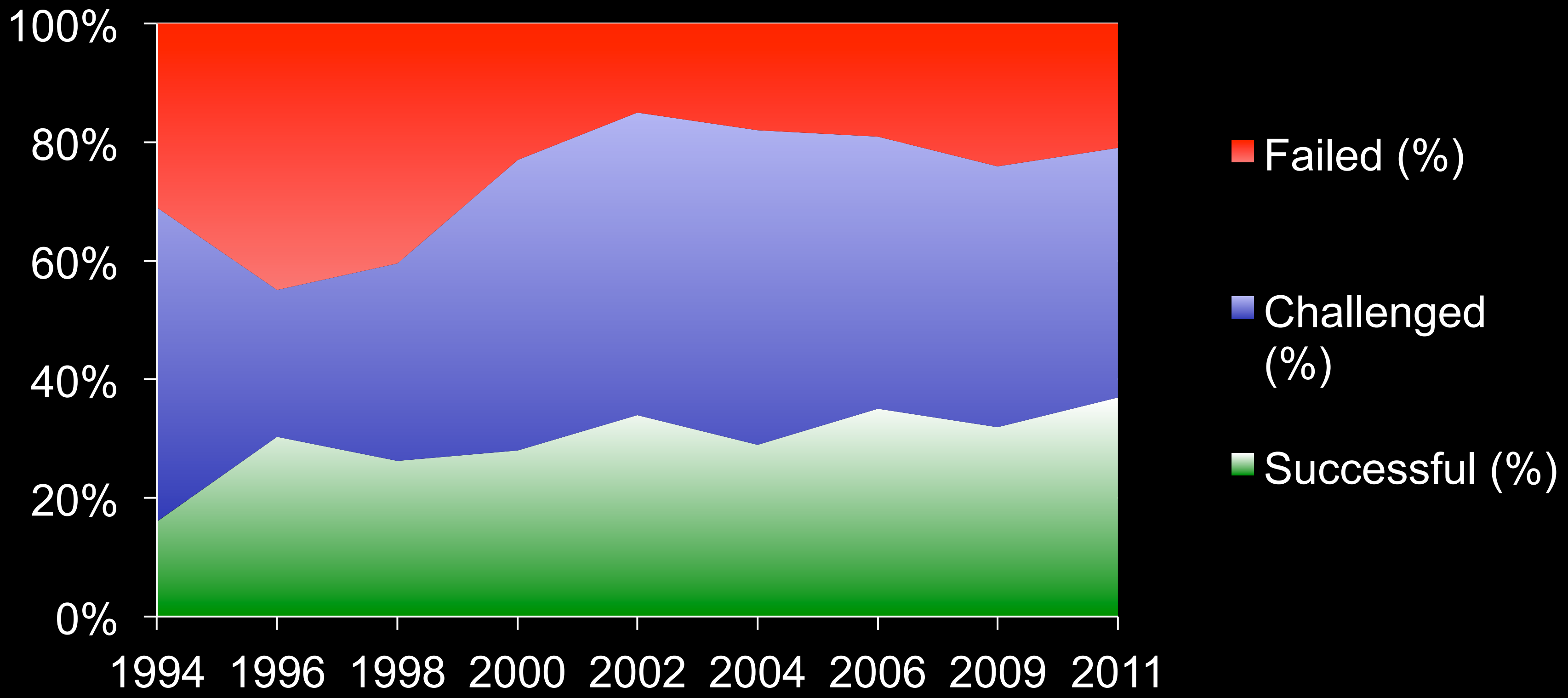
Reducing Risk in Software Projects Using Behavior Based Requirements

Jeffrey Davidson, CSPO PMC
e: jeffrey@davidson.net
b: goodrequirements.com

UT DALLAS
6th Annual Project Management Symposium
August 16, 2012

Standish Group, CHAOS Reports



Standish Group, CHAOS Reports

63%

Challenged or Failed

Forrester Research

40%

Dissatisfied

Standish Group, CHAOS Reports

45%

Features Never Used

Stable Requirements

<500

Function Points

Today's Agenda

- Thinking like an investor
- Communicating through examples

Thinking Like an Investor

IT Complexity Crisis

$3 trillion

Global impact of IT failures

Feature Injection

- Hunt the value
- Inject the features
- Spot the examples

Feature Injection: Hunt the Value

Models

- Business Value
- Specific
- Communicated

Feature Injection: Inject the Features

Features

- Focused
- Drives specific output

Feature Injection: Spot the Examples

Examples

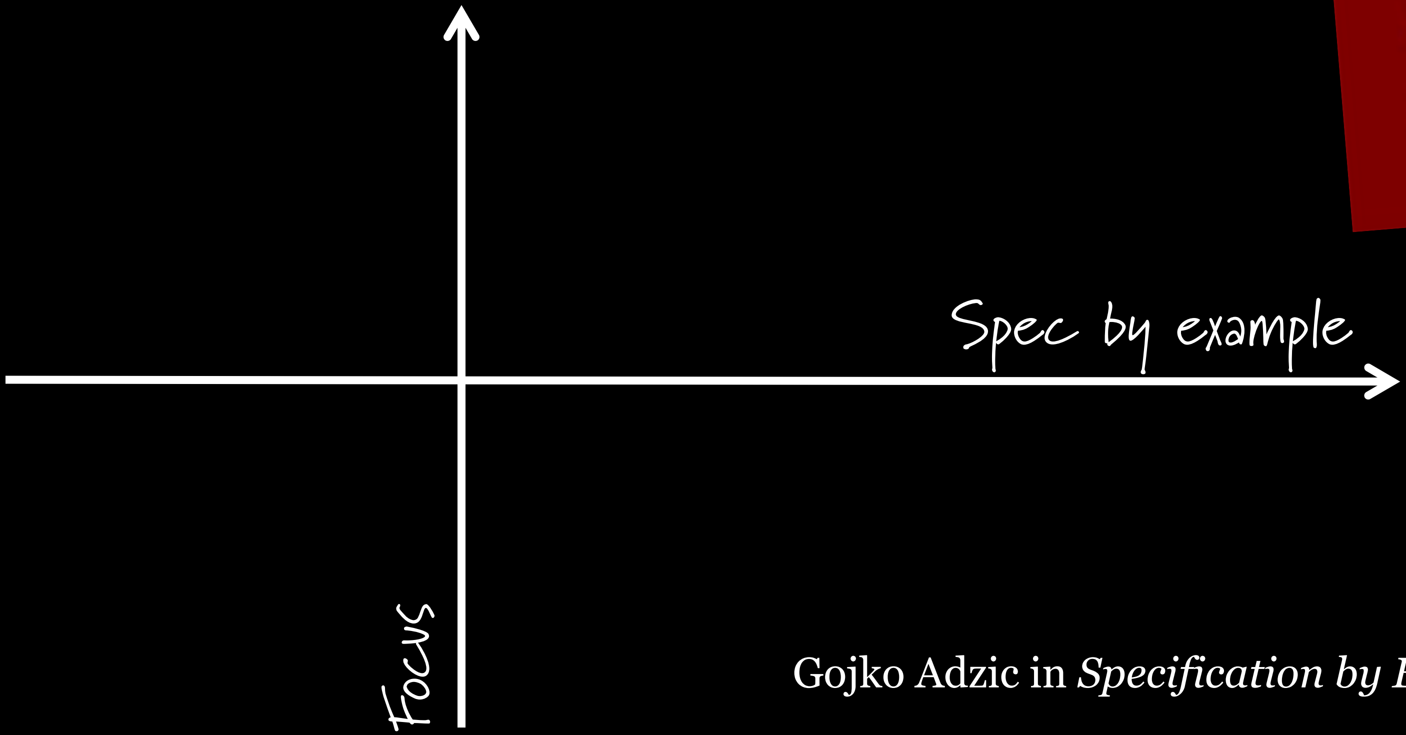
- Negative paths
- Communication
- Validation

Communicating with Examples

Stories

Built right or Right product?

Built Right



Right Product

Gojko Adzic

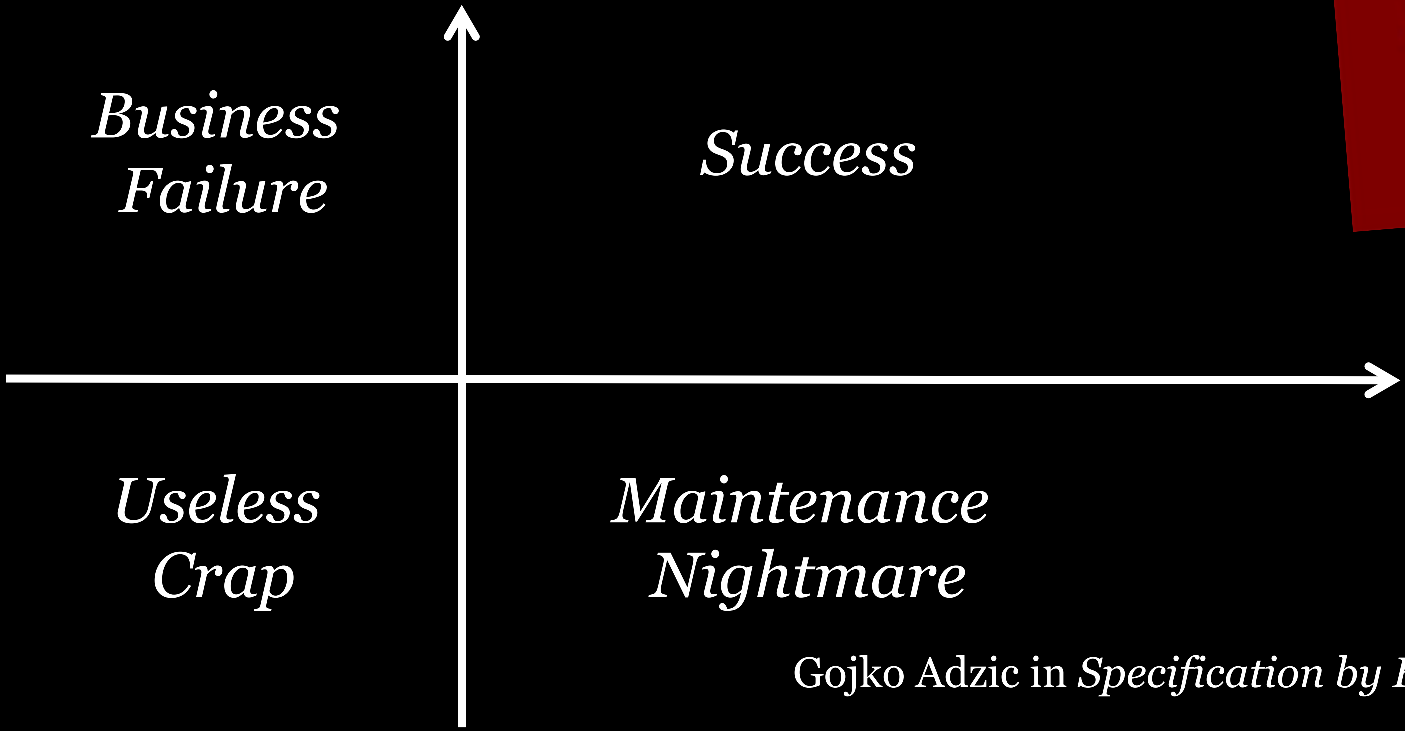
SPECIFICATION BY EXAMPLE

How successful teams deliver the *right* software

MANNING

Gojko Adzic in *Specification by Example*, 2011

Built right or Right product?



Built Right

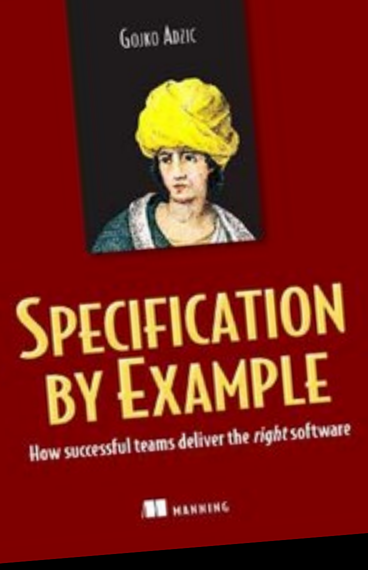
Business Failure

Success

Right Product

Useless Crap

Maintenance Nightmare



Gojko Adzic in *Specification by Example*, 2011

Simple?

- "We can only hope to make reliable those things that we can understand.
- We can only consider a few things at a time.
- Intertwined things must be considered together.
- Complexity undermines understanding."

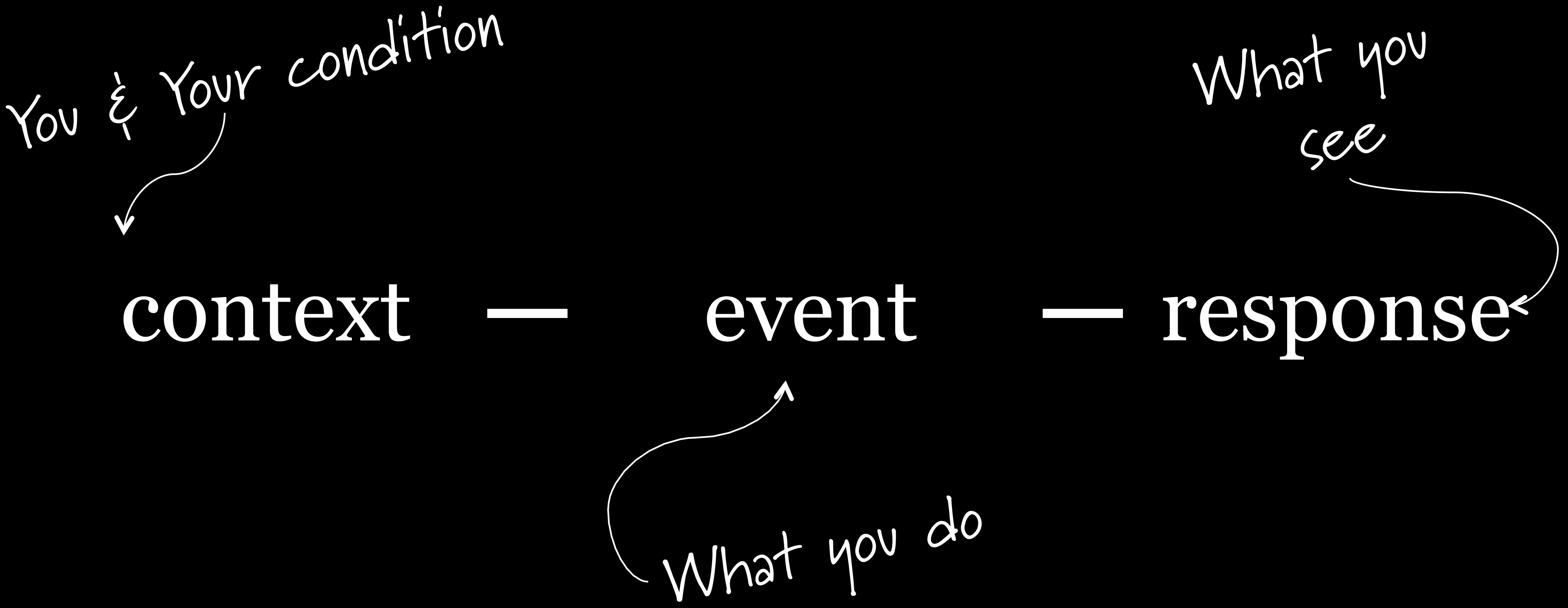
"Simple Made Easy" by Rich Hickey @ StrangeLoop 2011

Behavior Driven Development

BDD

- Behavior
- Story format

Simple Structure



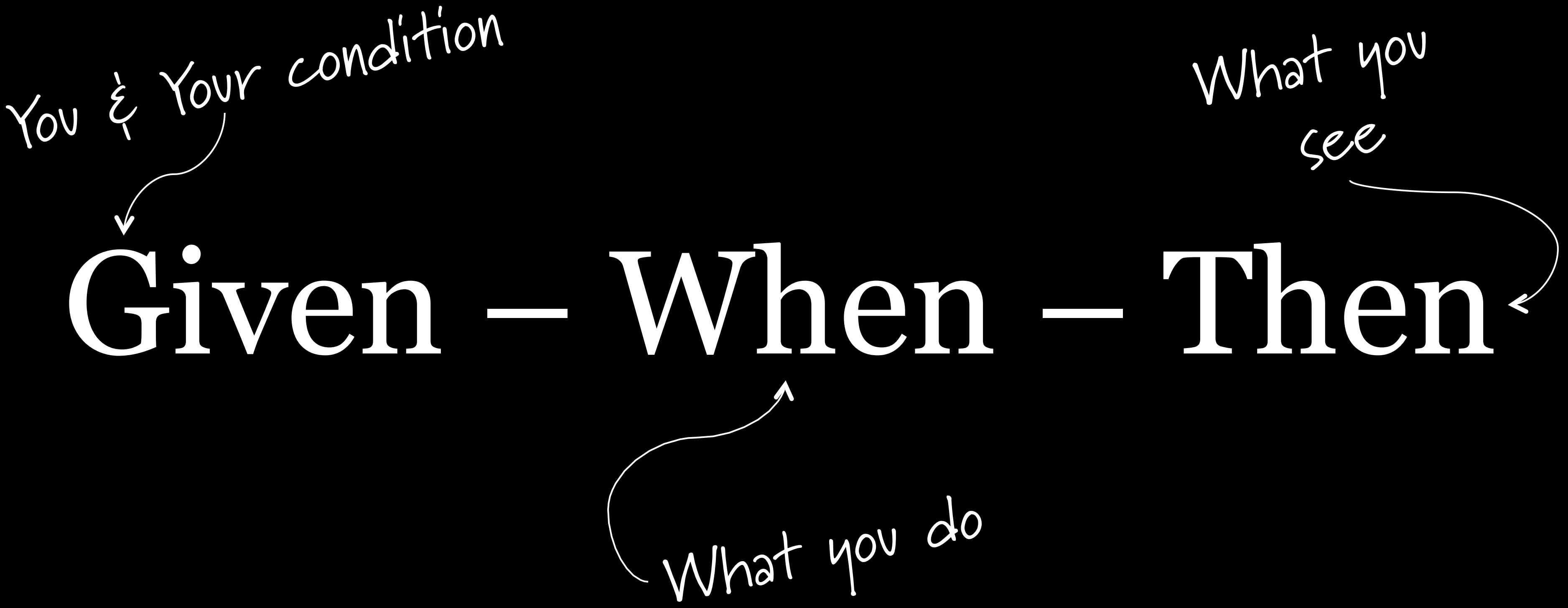
You & Your condition

context — event — response

What you see

What you do

Simple Structure



You & Your condition

What you see

Given – When – Then

What you do

Helpful Guidelines

Given: I am documenting reqs

When: I write req using **design agnostic behavior**

Then: My requirements may be reused

Helpful Guidelines

Given: I am documenting reqs

When: I write req using **natural language**

Then: My requirements will be understood

Helpful Guidelines

Given: I am documenting reqs

When: I write req using **business terms**

Then: My business partners will understand the req

Helpful Guidelines

Given: I am documenting reqs

When: I add **example scenarios with data**

Then: I can validate what has been built meets the goals

Who Benefits?

Everyone!

- Seriously, it helps everyone
  - Sponsors
  - Business partners
  - Users
  - Testers
  - Developers
  - Analysts

The Power of Stories

- Precise grammatical structure
- Discovery & understanding
- Captured conversation

Recap

- Find the value
- Model the output
- Communicate with behavior-based requirements

Recommended Resources

Gojko Adzic	BDD / Specification by Example gojko.net
Liz Keogh	BDD lunivore.com
Chris Matts	*Invented:* Feature Injection theitriskmanager.wordpress.com
Dan North	*Invented:* BDD dannorth.net

Share & Tell

This is licensed under Creative Commons Sharealike [CC BY 3.0]

- Please *use* it
- Please *share* it
- Please *improve* it
- As long as you credit me somewhere

Contact Information

Jeffrey Davidson, CSPO PMC

Principal Consultant, **Thought**Works®

President, IIBA Dallas

email	jeffrey@davidson.net
blog	http://goodrequirements.com
Twitter	@JeffreyGoodReq

Reducing Risk in Software Projects Using Behavior Based Requirements

Jeffrey Davidson, CSPO PMC
e: jeffrey@davidson.net
b: goodrequirements.com

UT DALLAS
6th Annual Project Management Symposium
August 16, 2012