scruminc.

AGILE TESTING PRACTICES

Building quality in for faster releases

Hosts: Joe Justice
Dave Sharrock

scruminc.: Who We Are

Scrum Inc. is the Agile leadership company of Dr. Jeff Sutherland, co-creator of Scrum. We are based in Cambridge, MA.

42 agile

The Agile Coaching Company

scruminc.

EXTREME MANUFACTURING

We maintain the Scrum methodology by:

- Capturing and codifying evolving best practices
- Conducting original research on organizational behavior
- Adapting the methodology to an ever-expanding set of industries, processes and business challenges

We also help companies achieve the full benefits of Scrum through our comprehensive suite of support services:

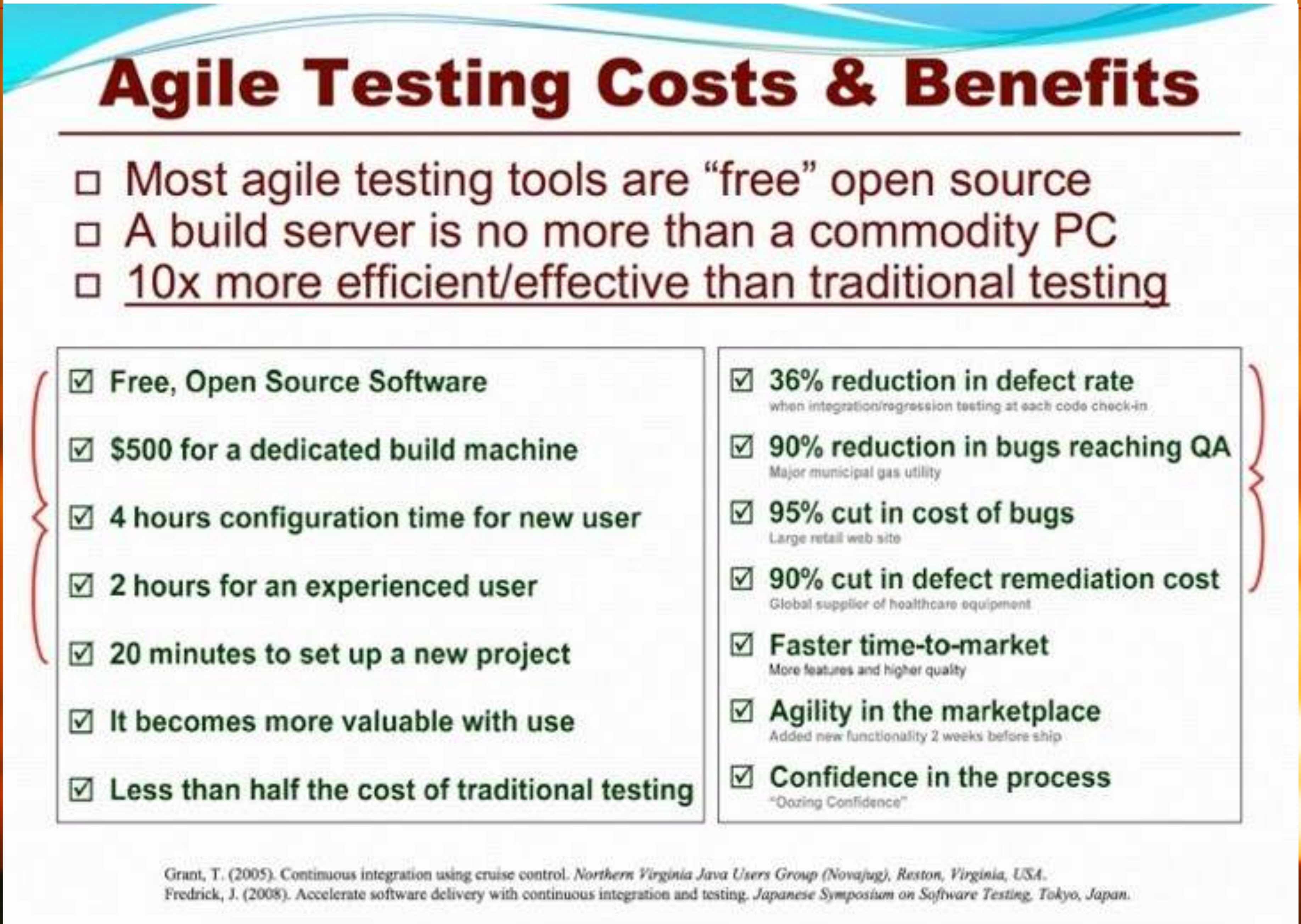
- Training (Scrum Master, Product Owner, Agile Leadership, online courses, etc.)
- Consulting (linking Scrum and business strategy, customizing Scrum)
- Coaching (hands-on support to Scrum teams)
- Publishing and new content development

We run our services company using Scrum as the primary management framework, making us a living laboratory on the cutting edge of "Enterprise Scrum"

Find out more at www.scruminc.com.

Agenda

- Agile Testing Costs & Benefits.
- Top Agile Testing Practices.
  - Test Driven Development, Test First Development, DevTest, DevTestOps.
  - Unit Testing, Integration Testing, Regression Testing, User Acceptance Testing.
- Tools to get testing done quickly.
- Modern test tools for your Weird/Backward/Deprecated/Unique environment.
  - Cucumber and Ruby on the as400 with tools written by Cheezy, at Nationwide Insurance.
  - TDD in Hardware and Red Green Refactor.
- The Business case and numbers for exploratory testing, manual testing, and building quality in.
  - 24 times the cost to repair a bug that escaped the sprint.
- Eliminating Technical Debt with Elegant Testing.
- Outsourced QA?
- How to Crush your Competitors with Agile Testing.
  - Part of the Solution? Gated Check-in and Rapid Roll Back in hardware and Software.
- The mindset of managing a successful agile testing practice.
  - Code Craftpersonship, shared ownership, elegance, building quality in.
- MIND BLOWING CONCLUSION: DevTestOps.
  - You build your test, you build your code, you integration test your code, you check in your automated tests, you build your deployment package and its tests, you update your deployment environment, you deploy your tests, you deploy your code. You do all of this in a pair. When you do this, competitors can’t keep up.



Agile Testing Costs & Benefits

- □ Most agile testing tools are “free” open source
- □ A build server is no more than a commodity PC
- □ 10x more efficient/effective than traditional testing

☑ Free, Open Source Software

☑ $500 for a dedicated build machine

☑ 4 hours configuration time for new user

☑ 2 hours for an experienced user

☑ 20 minutes to set up a new project

☑ It becomes more valuable with use

☑ Less than half the cost of traditional testing

☑ 36% reduction in defect rate
when integration/regression testing at each code check-in

☑ 90% reduction in bugs reaching QA
Major municipal gas utility

☑ 95% cut in cost of bugs
Large retail web site

☑ 90% cut in defect remediation cost
Global supplier of healthcare equipment

☑ Faster time-to-market
More features and higher quality

☑ Agility in the marketplace
Added new functionality 2 weeks before ship

☑ Confidence in the process
“Oozing Confidence”

Grant, T. (2005). Continuous integration using cruise control. *Northern Virginia Java Users Group (Novajug), Reston, Virginia, USA.*
Fredrick, J. (2008). Accelerate software delivery with continuous integration and testing. *Japanese Symposium on Software Testing, Tokyo, Japan.*

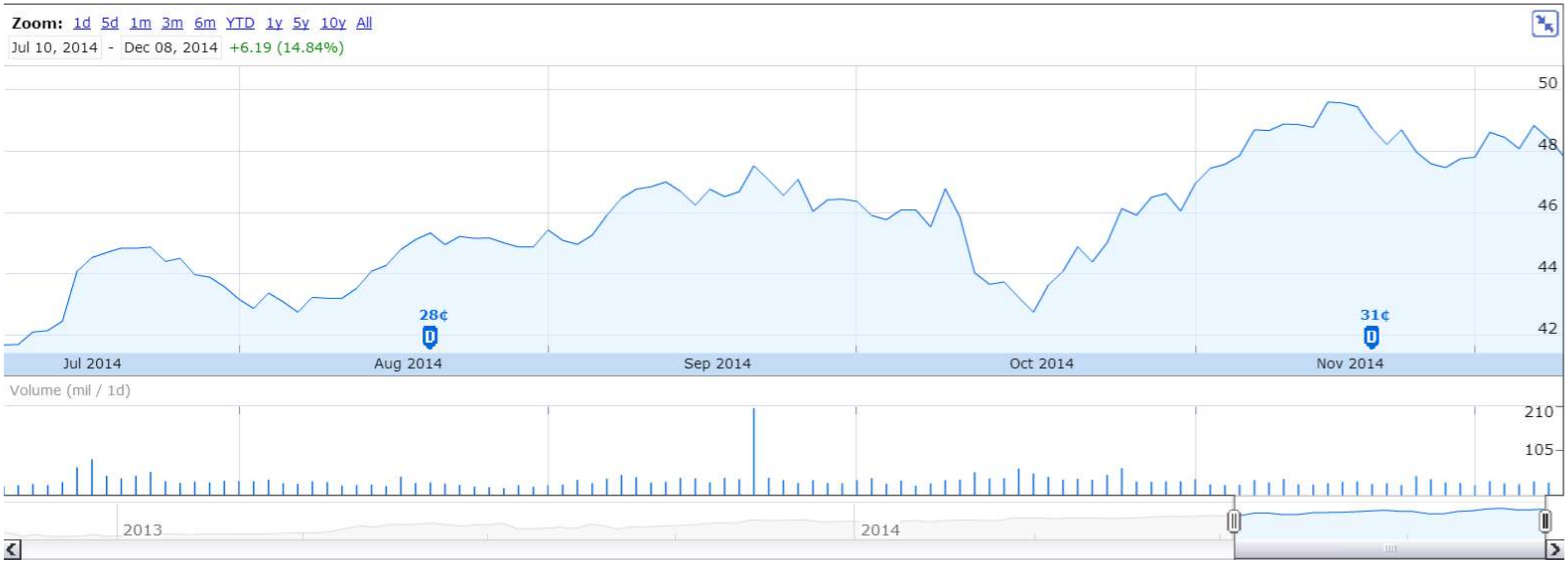
David Rico. Business Value of Agile Methods. 2013.

Winning at Agile Testing: be2

- - move from 8-12 hours unplanned downtime per release every 90 days to 1 hours planned downtime (2 hours for DB updates) every 2 weeks
  - eliminated hot fixes as a normal release occurrence
  - eliminated need for post-release testing (I know, who thought there was ever a need for this!) by PO group
  - went from <1% test coverage to getting on for 30-50% (1000’s of tests) in about 18 months using DoD and a concerted effort/understanding from team

WSJ: Microsoft CEO bans test teams

- Dev after test is too slow! Banned “to cut costs while building software faster” –WSJ.
- Triggered immediate stock climb, peaking at 10% shareholder gain. -NYSE



Source: Wall Street Journal: http://blogs.wsj.com/cio/2014/07/15/microsoft-plots-agile-development-course-as-talk-on-job-cuts-loom/ Source: NYSE via google finance. Source: Bloomberg: http://www.bloomberg.com/news/2014-07-10/microsoft-ceo-nadella-outlines-vision-with-more-overhauls.html

Top Agile Testing Practices

1 Test inside the Sprint

2 Automate tests

I. Observed norm for teams releasing each sprint.

3 Gated Check-In, Rapid Roll Back

4 TDD

II. Observed norm in competitive markets.

5 Red, Green, Refactor

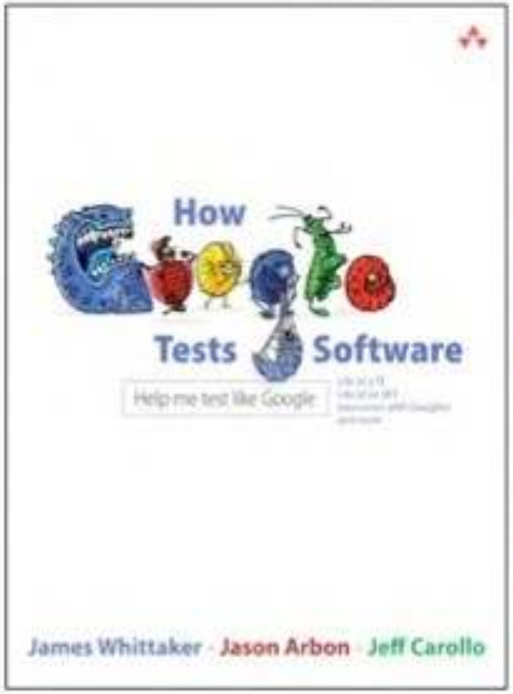
6 DevTestOps

III. Observed in the fastest companies.

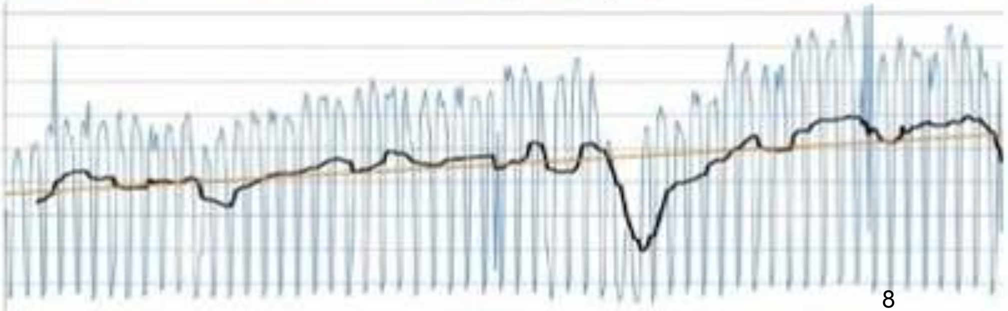
scruminc.

How Google Tests Software

- □ Google early adopter of agile methods and Scrum
- □ Google also uses agile testing at enterprise scale
- □ 15,000 developers run 75+ million tests per day

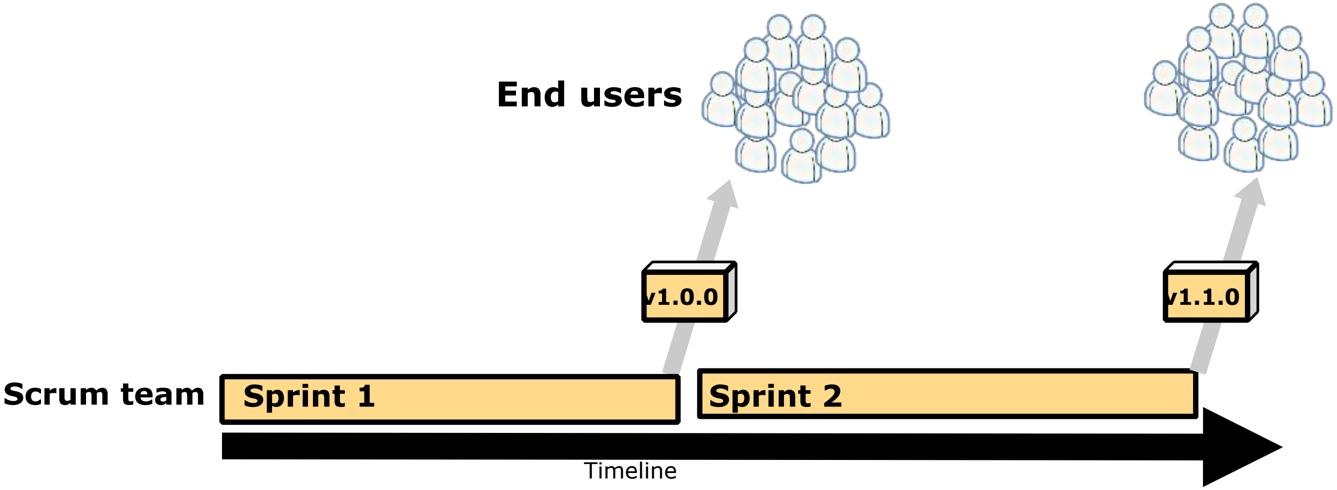

How Google Tests Software
Help me test like Google
James Whittaker · Jason Arbon · Jeff Carollo

- **15,000+ developers in 40+ offices**
- **4,000+ projects under active development**
- **5,500+ submissions per day on average**
- **Single monolithic code tree with mixed language code**
- **Development on one branch - submissions at head**
- **All builds from source**
- **20+ sustained code changes per minute with 60+ peaks**
- **50% of code changes monthly**
- **75+ million test cases run per day**



Micco, J. (2013). *Continuous integration at google scale*. Eclipse Con, Boston, MA.
Whittaker, J., Arbon, J., & Carollo, J. (2012). *How google tests software*. Upper Saddle River, NJ: Pearson Education.

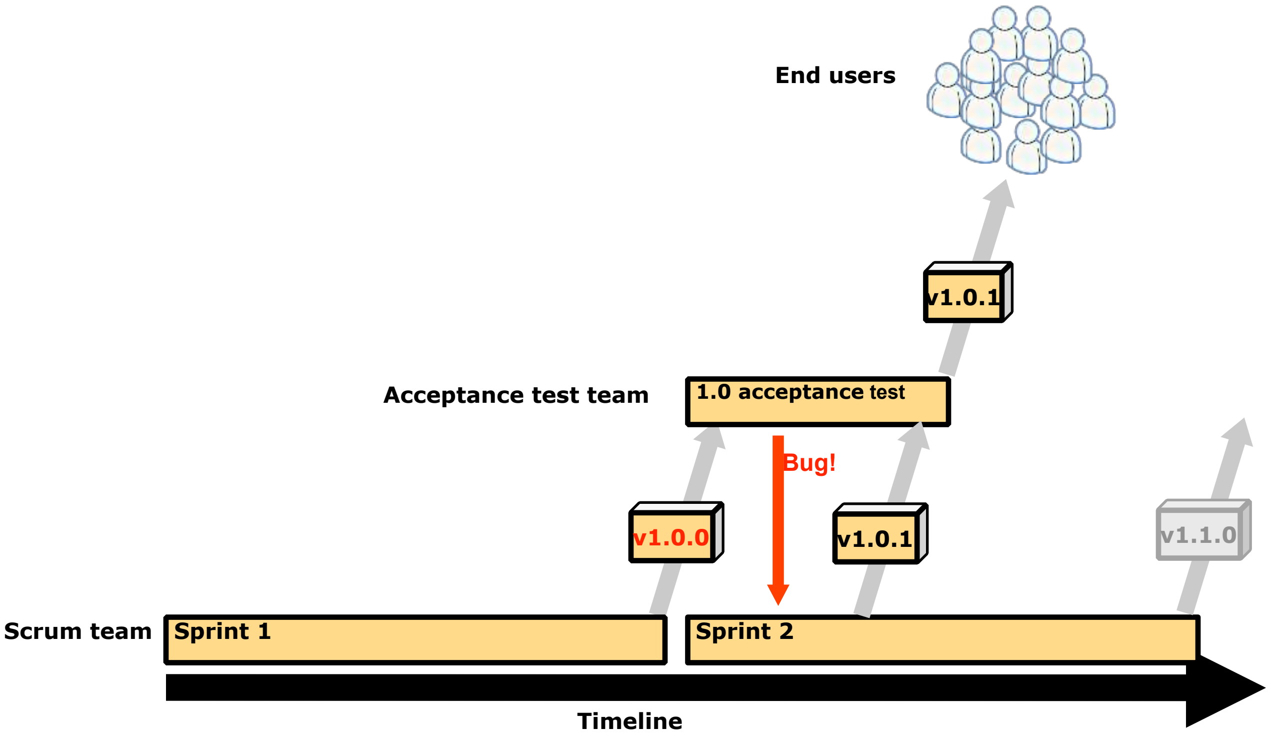
Testing – Ideal Case



Source, Adapted From: Henrik Kniberg

scruminc.

Testing – The Sad, Common Alternative



Source, Adapted From: Henrik Kniberg

Agile Testing

- We know you aren't doing it, most of the teams we work with have this as their #1 reason for failing the sprint (work carrying over), so let's give you to the tools to make it happen.

Tools to make this happen quickly

- Cucumber
  - Plain language DSL tool.
  - Product Owners can use directly.
- FitNesse
  - Wiki driven ATD framework.
  - Collaborative and information sharing.
- The PowerTools
  - 2nd gen test framework.
  - Possibly the most elegant, compact.
- docker
  - 2nd gen Virtual Machine service.
  - Ultra compact VM files.
  - Automate tests across thousands of test instances.

Central Desktop

- Venture company with no testers.
- Cucumber experts writing scripts with the product owners to accompany new backlog stories.
- Developers get a User Story and a red test.
- The PO says "Dev team, turn these green and you are done."
- CTO is chief PO, and can hit the deploy button at the end of any sprint.
- Enabled by policy, early on, Remediate Technical Debt as soon as it is found.

Acceptance Test Driven Development (ATDD) Tools: Fit and Cucumber

FIT (Framework for Integrated Test) and Fitnesse (Wiki front end)

- Test specified in table format
- Developers generates classes ("fixtures") to hook into application
- Users/testers use Wiki or Excel to specify inputs and outputs

numerator	denominator	quotient
10	2	5
12.6	3	4.2
100	4	25 expected 24 returned

Cucumber

- Tool for natural language scenarios

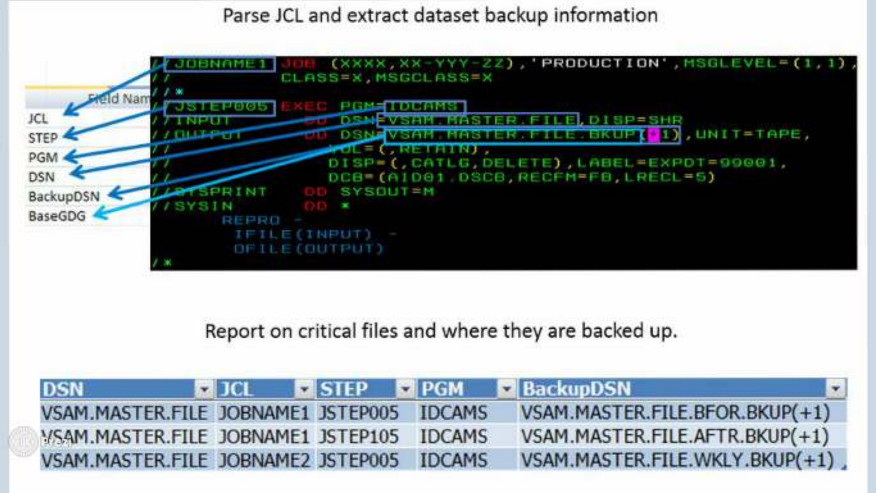
In order to ensure my account is correct
As a Registered User
I want to check my recent activity

Scenario: Recent Account Activity
Given I am a registered user "Jsmith"
And I am logged in with password "xyx123"
And I have had account activity in the last 45 days
And I am on the home page
When I click the "Recent Activity" button
Then I should see the "Account Activity" Page
And I should see a list of my activity over the last 45 days

Scenario: No Recent Account Activity
Given I am a registered user "Jsmith"
And I am logged in with password "xyx123"
And I have had no account activity in the last 45 days
And I am on the home page
When I click the "Recent Activity" button
Then I should see the "Select Account Activity Period" Page
And I should get a message: *"You had no activity in the last 45 days, please select a time frame to review"*

Modern test tools are available for your weird/ backward/deprecated/unique environment.

- **Cucumber and Ruby on the as400**: with tools by Jeff “Cheezy” Morgan, at Nationwide Insurance.

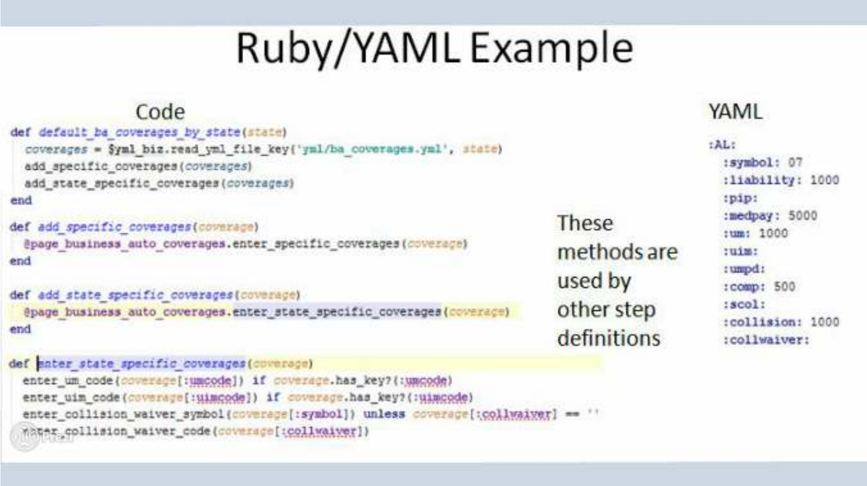


Parse JCL and extract dataset backup information

```
//JOBNAME1 JOB (XXXX,XX-YYY-ZZ),'PRODUCTION',MSGLEVEL=(1,1),
//             CLASS=X,MSGCLASS=X
//*
//JSTEP005 EXEC PGM=IDCAMS
//INPUT      DD DSN=VSAM.MASTER.FILE,DISP=SHR
//OUTPUT     DD DSN=VSAM.MASTER.FILE.BKUP(+1),UNIT=TAPE,
//                VOL=(,RETAIN),
//                DISP=(,CATLG,DELETE),LABEL=EXPDT=99001,
//                DCB=(A1D01.DSCB,RECFM=FB,LRECL=5)
//SYSPRINT   DD SYSOUT=M
//SYSIN      DD *
    REPRO -
     IFILE(INPUT) -
     OFILE(OUTPUT)
/*
```

Report on critical files and where they are backed up.

DSN	JCL	STEP	PGM	BackupDSN
VSAM.MASTER.FILE	JOBNAME1	JSTEP005	IDCAMS	VSAM.MASTER.FILE.BFOR.BKUP(+1)
VSAM.MASTER.FILE	JOBNAME1	JSTEP105	IDCAMS	VSAM.MASTER.FILE.AFTR.BKUP(+1)
VSAM.MASTER.FILE	JOBNAME2	JSTEP005	IDCAMS	VSAM.MASTER.FILE.WKLY.BKUP(+1)



Ruby/YAML Example

Code

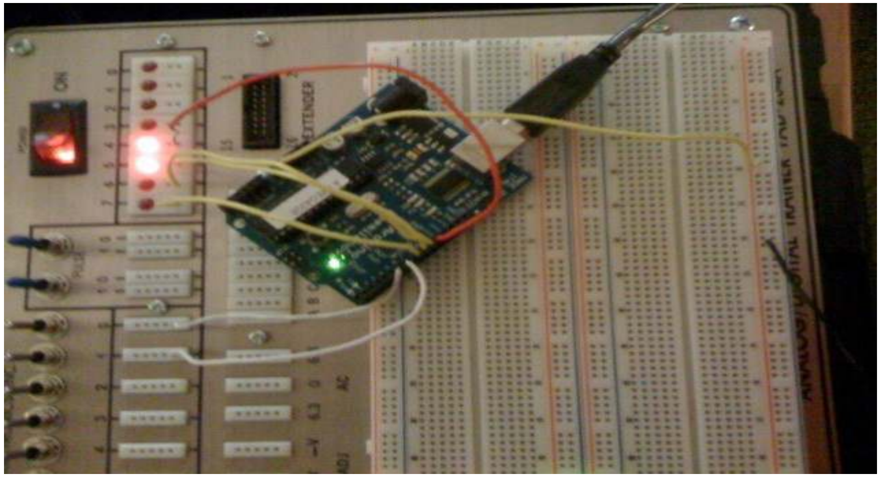
```ruby
def default_ba_coverages_by_state(state)
  coverages = $yml_biz.read_yml_file_key('yml/ba_coverages.yml', state)
  add_specific_coverages(coverages)
  add_state_specific_coverages(coverages)
end

def add_specific_coverages(coverage)
  @page_business_auto_coverages.enter_specific_coverages(coverage)
end

def add_state_specific_coverages(coverage)
  @page_business_auto_coverages.enter_state_specific_coverages(coverage)
end

def enter_state_specific_coverages(coverage)
  enter_um_code(coverage[:umcode]) if coverage.has_key?(:umcode)
  enter_uim_code(coverage[:uimcode]) if coverage.has_key?(:uimcode)
  enter_collision_waiver_symbol(coverage[:symbol]) unless coverage[:collwaiver] == ''
  enter_collision_waiver_code(coverage[:collwaiver])
```

These methods are used by other step definitions

YAML

```yaml
:AL:
  :symbol: 07
  :liability: 1000
  :pip:
  :medpay: 5000
  :um: 1000
  :uim:
  :umpd:
  :comp: 500
  :scol:
  :collision: 1000
  :collwaiver:
```

- **Arduino Test Fixture for Scrum Hardware.**
  - Red Light, Green Light.
  - Test Logic onboard.
  - $24 USD.
  - Ships same day.



Sources: Nationwide's Application Development Center, Danielle Roecker, Team WIKISPEED.

Simplifying critical decision making

SYSTEMATIC

Systematic

- Already 98% on time and on budget, CMMI level 5.
- How to improve from there? Half cost to deliver, project management cost down 80%, and reduce defects by 40%!
- How to do that? By monitoring the metric that is king of them all:
- Time to fix a bug.
  - Monitored by control charts.
  - If time went over 2 hours, Scrum Masters immediately performed root cause analysis and systemized the resolution.
    - Carsten Jakobson, their internal process champion, has paired to published 5 papers documenting this practice and it’s success, peer reviewed.
    - Measured bug fix time ***24x less*** if resolved in sprint.
    - Couldn’t afford NOT to exploratory test, manual test, automated test, and BUILD QUALITY IN.
      - Repeated at Palm on web operating system code.

Sources: IEEE digital Library, “Scrum and CMMI Going From Good To Great”. Jeff Sutherland and Carsten Jakobson.

Gartner - Technical Professional Advice
2012 Planning Guide: Application Delivery Strategies

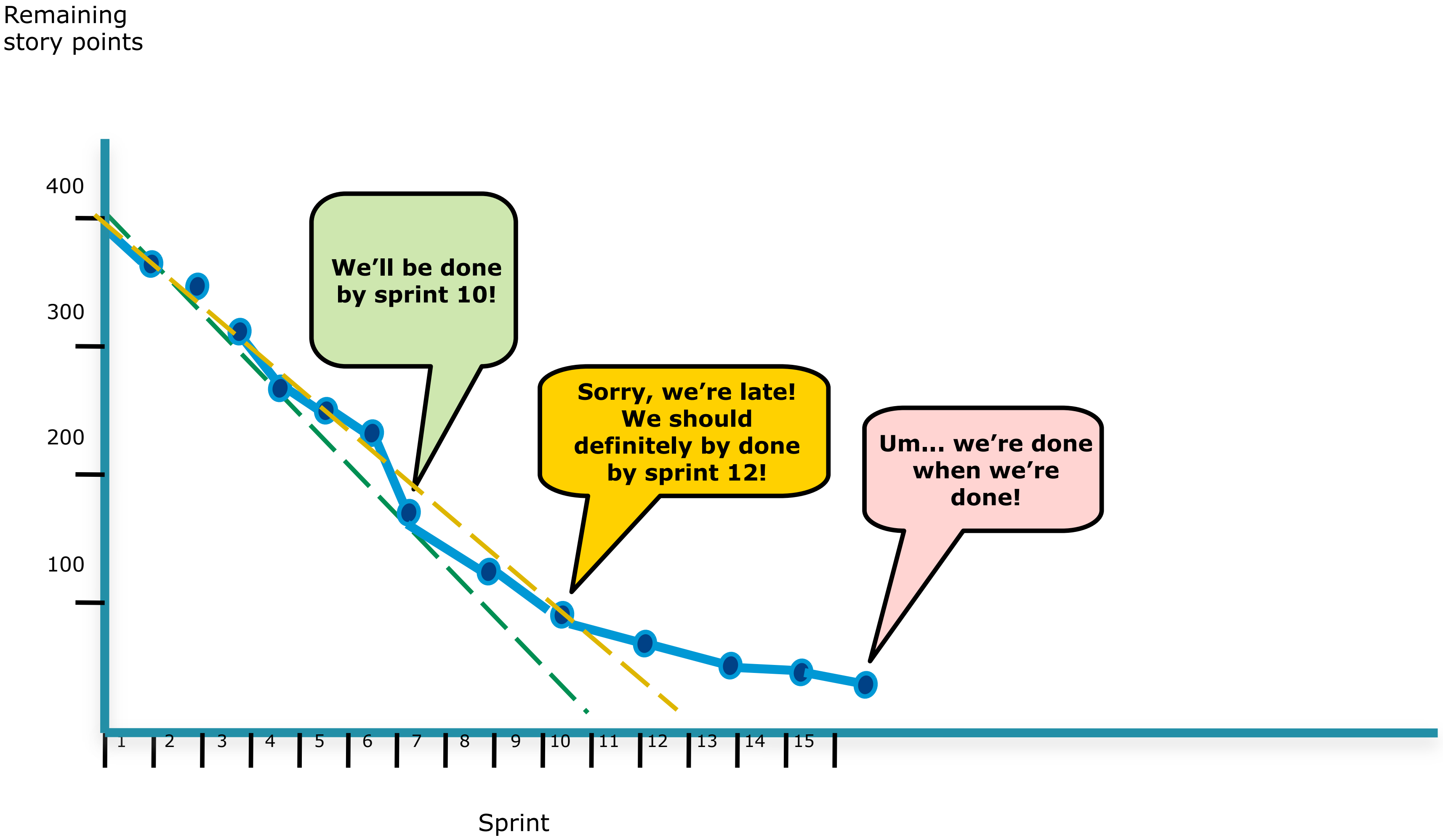
- Business users are losing patience with old-school IT culture. Relationships are tense and resentful. Legacy systems and practices impede agility. Bottom line - GET AGILE
- **Adopt a product perspective.**
- **Say goodbye to waterfall.**
- **Improve cross-competency collaboration.**
- **Launch a deep usability discipline.**
- **Start a technical debt management program.**

Gartner.
CIO Leadership Forum

An executive-only program, designed to explore critical business, technology and leadership strategies.

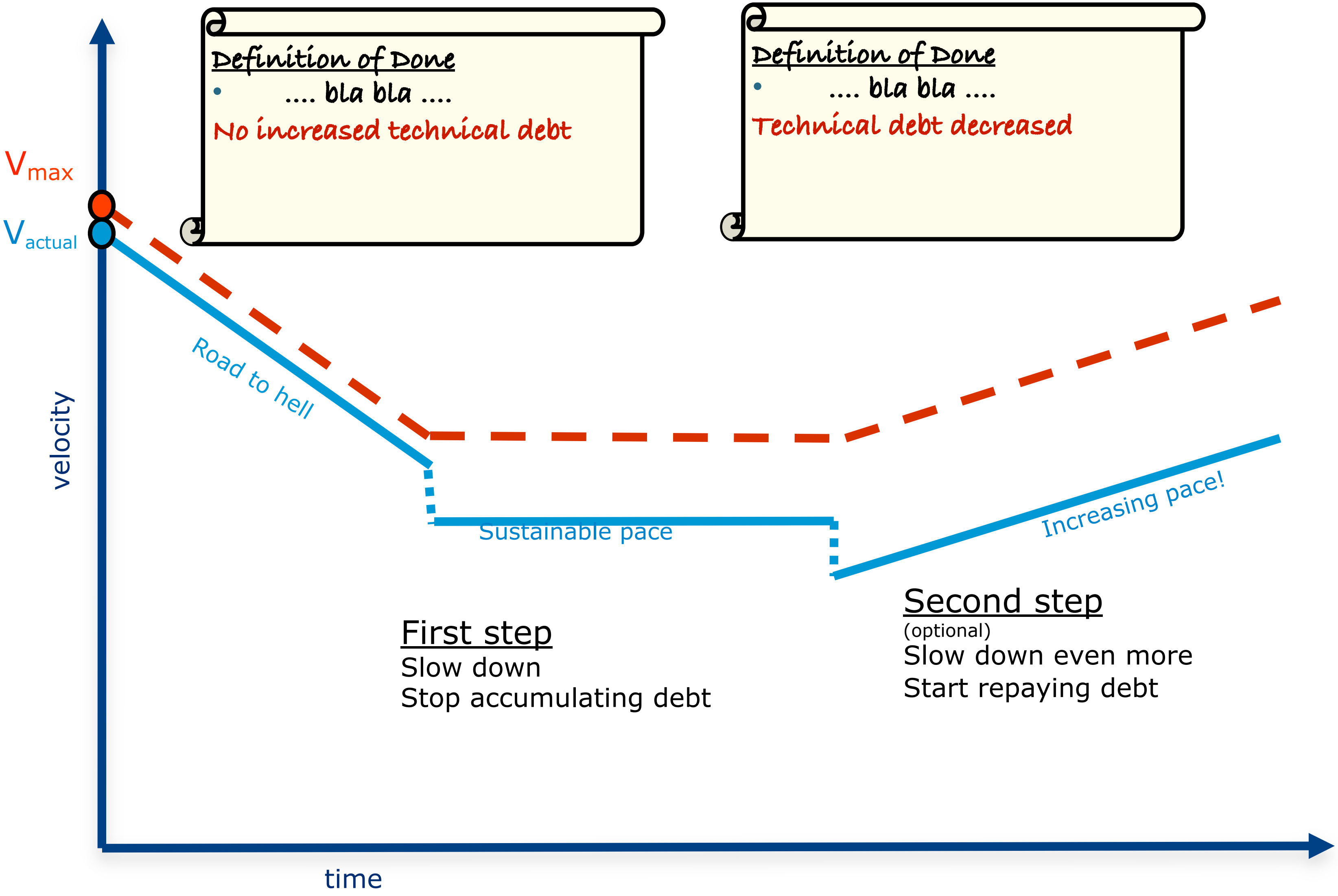
27-28 February, Dubai | 5-7 March, London | 25-27 March, Scottsdale

Technical Debt & Release Planning



Source: Adapted from Henrik Kniberg

Dealing with Technical Debt



Definition of Done

- .... bla bla ....

No increased technical debt

Definition of Done

- .... bla bla ....

Technical debt decreased

First step

Slow down

Stop accumulating debt

Second step

(optional)

Slow down even more

Start repaying debt

Source: Adapted from Henrik Kniberg

The Mindset of Managing a Successful Agile Testing Practice

A **Elegance.** UI tests that rely on click maps or control placement are brittle, UI tests that call the button click event or control programmatically (pinouts in hardware) are robust!

B **Shared Ownership.** Test doesn’t QA Dev’s solution, we don’t call our solution done until it meets our Definition of Done.

C **Build Quality In.** Done solutions are not just tested, but refactored in place to (re)establish elegant architecture in that piece of the solution.

D **Craftspersonship.** Always inspired and inspiring to improve the Definition of Done:

- User Acceptance Testing,
- Unit testing,
- integration testing,
- regression testing,
- User Acceptance testing.

OUTSOURCED QA?

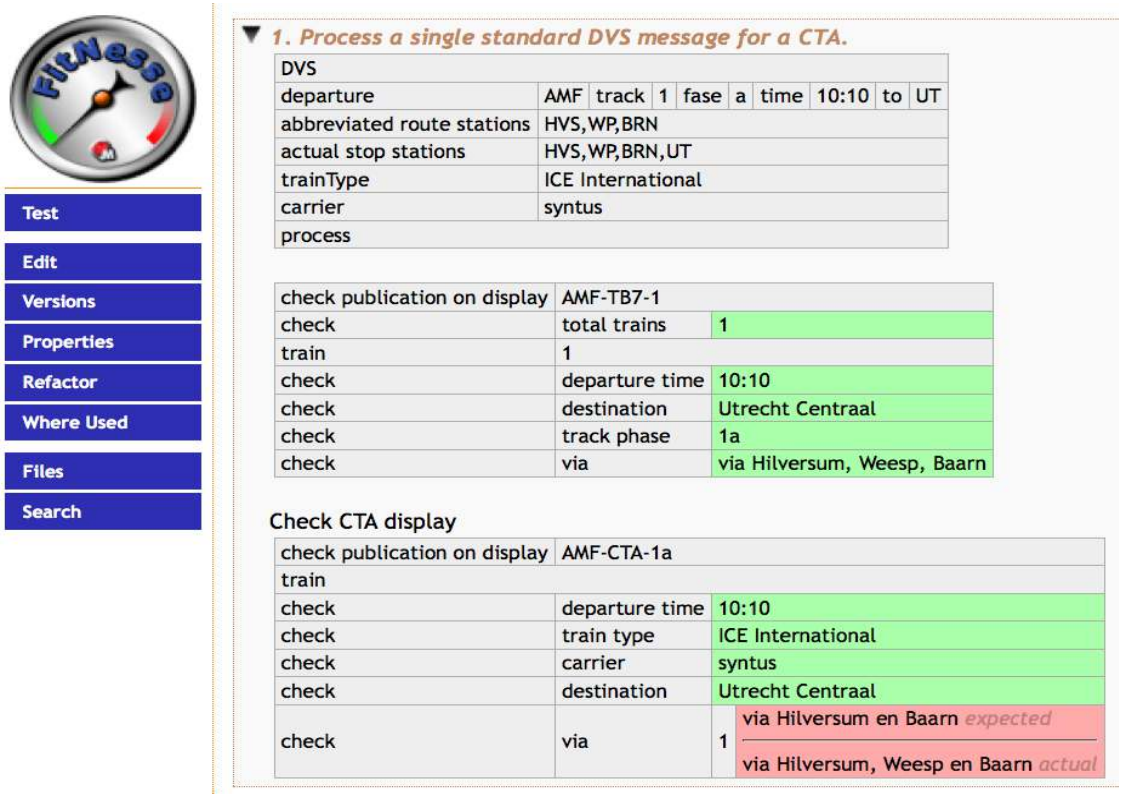
Xebia Case study: Building a new railway information system (ProRail)

2
21:35 Stoptrein
Leiden Centraal
via Naarden-Bussum, Weesp, Diemen Zuid, Duivendrecht en verder

- ProRail rescued a failed waterfall project to build a new scheduling system and automated railway station signs at all Netherlands railway stations
- An 8 person half-Dutch and half-Indian Scrum team started the project and established local velocity.
- After establishing local velocity at 5 times other waterfall vendors on the project, the Indian half of the team went back to India.
- Scrum teams run all XP practices inside the Scrum including intensive pair programming.
- The customer completes acceptance testing on all features during each Sprint.
- Done at the end of the Sprint means customer has accepted the code as ready for production.
- Defect rates are less than 1 per 1000 lines of code and steadily getting lower.

Integrating analysis, testing and implementation

- Analysts, testers and developers define together how a user story will be tested using FitNesse
- Testers write functional tests while the user story is being implemented
- Developers write 'fixtures', which link tests with the production code
- Regression tests are run continuously
- 95% of Bugs are found and fixed within the Sprint



▼ *1. Process a single standard DVS message for a CTA.*

DVS									
departure	AMF	track	1	fase	a	time	10:10	to	UT
abbreviated route stations	HVS,WP,BRN								
actual stop stations	HVS,WP,BRN,UT								
trainType	ICE International								
carrier	syntus								
process									

check publication on display	AMF-TB7-1	
check	total trains	1
train	1	
check	departure time	10:10
check	destination	Utrecht Centraal
check	track phase	1a
check	via	via Hilversum, Weesp, Baarn

Check CTA display

check publication on display	AMF-CTA-1a		
train			
check	departure time	10:10	
check	train type	ICE International	
check	carrier	syntus	
check	destination	Utrecht Centraal	
check	via	1	via Hilversum en Baarn *expected* via Hilversum, Weesp en Baarn *actual*

Crush Your Competitors with Agile Testing

- If you are the CEO, like Microsoft: BAN TEST TEAMS.
- If you are the Director of Engineering:
  - Identify and encourage Test Leads as XP practices champions in each team.
  - Configure build servers for each team yourself, or bring in a consultant to do it.
    - Turn on Gated Check-in.
    - Make Automated Test Coverage visible, measure each team's increase in code coverage.
- If you are the team Test Lead or XP practices champion:
  - Read up on elegant architecture and radical refactoring.
  - Practice your own ability as DevTestOps.
  - Share with everyone interested. Host a book club.
  - Make visible the number of people on your team who can dev, test, deploy.
- If you are the team Product Owner:
  - Write backlog in a test tool as a failing test.
  - Demand all estimates include no increased technical debt.
  - Demand Definition of Done includes Refactor In Place.
- If you are the team Scrum Master:
  - Measure time to fix a bug.
    - Like Systematic, If more than 2 hours for a bug, immediately perform root cause analysis and systematize the resolution.

Questions?

Wallabies

scruminc.

Stay Connected

Scruminc.com

- For up coming events, new content releases, and more!

ScrumLab

- articles, online courses, tools, and papers on all things scrum
- www.scruminc.com/scrumlab

Blog

- http://www.scruminc.com/category/blog/

Online Courses

- advance your scrum with our online courses. Visit the Scrumlab store to view upcoming topics.

Twitter, Facebook, and G+

- @ScrumInc, @JoeJustice0, scrum and scrum inc.

Acceptance Tests

Modern Agile Acceptance Model

- The move toward Acceptance Test Driven Development requires a more complete model of progressing requirements acceptance
- Example model (sharper definition of done)
  - Conditions of Satisfaction - broad terms
    - Acceptance Criteria - further refined
      - Examples - actual scenarios or data
        - Executable Examples - Executable Spec

Simple Scenarios

- Suppose we are creating a registration function
- It consists of specifying email (as User ID) and Password

Enter your Email and Password to Register	
Email Address:	
Password:	
	Register

Acceptance
Condition of Satisfaction

- Conditions of Satisfaction are high level criteria established with an initial Epic/Feature/User Story
  - In our previous example, the conditions of acceptance for the password would be:
    - Ensure the password is not easy to crack.
- The PO would define the conditions of acceptance in concert with business stakeholders

Enter your Email and Password to Register	
Email Address:	
Password:	
	Register

Acceptance
Examples Make It Real

- Actual examples are the ultimate way to communicate requirements:

Password	Expected	Expected Message	Comment
abc123	Invalid	Your password must be at least 8 characters and no more than 12 charcters long	
abcdefghi	Invalid	Your password must contain at least one character and one number	
1aaaaaaaa	Valid		Why valid?
ajx972dab	Valid		

- The PO works closely with a tester, developer, and business stakeholder to define these criteria

Acceptance

Acceptance Criteria

- Acceptance Criteria specifies the details that the team can then build against:
  - Following the example for Password:
    - Must be at least 8 characters and no more than 12
    - Must contain only alpha-numberics and the period
    - Must contain at least one digit
    - Must contain at least one character
    - Etc. (there may be more criteria)
- The PO works closely with a tester, developer, and business stakeholder to define these criteria

Enter your Email and Password to Register	
Email Address:	
Password:	
	Register

scruminc.

MIND BLOWING NEXT STEP

- DevTestOps

Conclusions

- If you are the CEO, like Microsoft: BAN TEST TEAMS.
- If you are the Director of Engineering:
  - Identify and encourage XP practices champion on each team.
  - Configure build servers for each team yourself, or bring in a consultant to do it.
    - Turn on Gated Check-in.
    - Make Automated Test Coverage visible, measure each team's increase in code coverage.
- If you are the team XP practices champion:
  - Read up on elegant architecture and radical refactoring.
  - Practice your own ability as DevTestOps.
  - Share with everyone interested. Host a book club.
  - Make visible the number of people on your team who can dev, test, deploy.
- If you are the team Product Owner:
  - Write backlog in a test tool as a failing test.
  - Demand all estimates include no increased technical debt.
  - Demand Definition of Done includes Refactor In Place.
- If you are the team Scrum Master:
  - Measure time to fix a bug.
    - Like Systematic, If more than 2 hours for a bug, immediately perform root cause analysis and systematize the resolution.

Distributed Testing?

OUTSOURCED QA?

Xebia Case study: Building a new railway information system (ProRail)

2
21:35 Stoptrein
Leiden Centraal
via Naarden-Bussum, Weesp, Diemen Zuid, Duivendrecht en verder
Voor Amsterdam C overstappen in Weesp
Vanaf Schiphol stoptrein

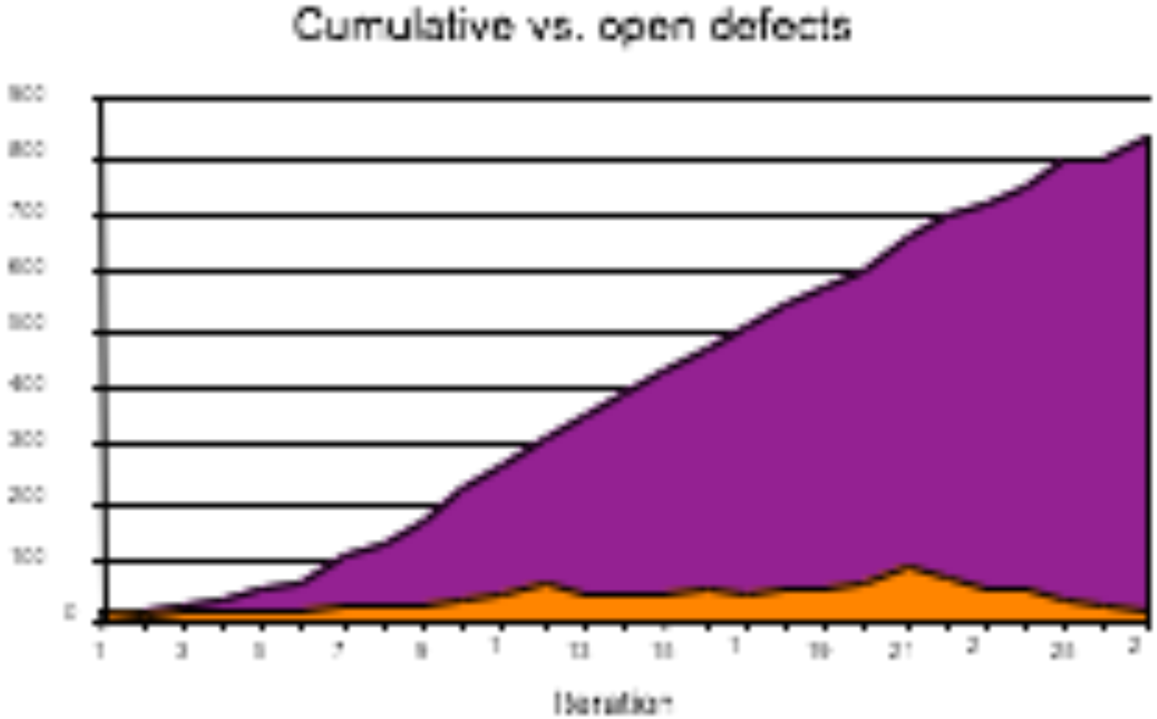
- ProRail rescued a failed waterfall project to build a new scheduling system and automated railway station signs at all Netherlands railway stations
- An 8 person half-Dutch and half-Indian Scrum team started the project and established local velocity.
- After establishing local velocity at 5 times other waterfall vendors on the project, the Indian half of the team went back to India.
- Scrum teams run all XP practices inside the Scrum including intensive pair programming.
- The customer completes acceptance testing on all features during each Sprint.
- Done at the end of the Sprint means customer has accepted the code as ready for production.
- Defect rates are less than 1 per 1000 lines of code and steadily getting lower.

Testing Implementation

- New Product Development:
  - WIKISPEED Red/Green card example.
    - Start with epics phrased as tests
    - Split until experiment can be run and tested in less than a sprint.
- Known Product Development:
  - How to rescue a failed Scrum.
    - Add testers to the Scrum team to meet the real definition of done, DEPLOYABLE.

Xebia/ProRail Defect Tracking

- Defect rate gets lower and lower as code base increases in size
- 95% of defects found inside iteration are eliminated before the end of the iteration



- TDD, pair programming, continuous integration. Same tools and techniques onshore and offshore.
- Daily Skype video Scrum meeting of team across geographies.
- SmartBoards, wikis, and other tools used to enhance communication.
- Indians say it feels exactly the same in India as it does in Amsterdam. They do the same thing in the same way.

Categories of Testing

- User Acceptance Testing,
- Unit testing,
- integration testing,
- regression testing,
- User Acceptance testing.

The mindset of managing a successful agile testing practice.

- Code Craftpersonship, shared ownership, elegance, building quality in.

Good Agility Starts with Good Testing

- Gated Check-in and Rapid Roll Back in hardware and Software.

Agile Testing

How is testing approached differently in traditional and agile contexts?

Requirements

Build

Testing?

Deploy

RBTD repeat – intermediate.

RTBTDTR repeat – skillful.

scruminc.

scruminc.

Velocity and Technical Debt

The numbers for exploratory testing, manual testing, automated testing, and building quality in:

- It costs 24x as much to fix a bug that escapes the current sprint.
- Corroborated studies in the U.S. and Europe.
- Found at Palm and Systematic.
- The solution?
  - Testing in the sprint saves 24 times the bug fix time, implemented by testers joining dev team and only testers can move cards across backlog.

Sources: IEEE digital Library, “Scrum and CMMI Going From Good To Great”. Jeff Sutherland and Carsten Jakobson.