

## THE INS AND OUTS OF ENTRANCE AND EXIT CRITERIA

March 26, 2015 – ASTQB Webinar

**RANDALL W. RICE, CTAL**  
**RICE CONSULTING SERVICES, INC.**

### A LITTLE BACKGROUND

- This topic came about in response to a question from one of our certified testers who is a test manager.
- I volunteered to answer the question, “I am interested in obtaining best practice information on entrance and exit criteria and measurements for when to stop testing. Would you know of any sources that you can direct me to find this information?”
- My answer turned into a long response, which I turned into an article recently published in the ASTQB Newsletter:  
[http://www.astqb.org/press-room/ISTQB\\_Certification\\_News\\_2015\\_1.html](http://www.astqb.org/press-room/ISTQB_Certification_News_2015_1.html)



# The Ins and Outs of Entry and Exit Criteria

3/26/15

## MY BACKGROUND

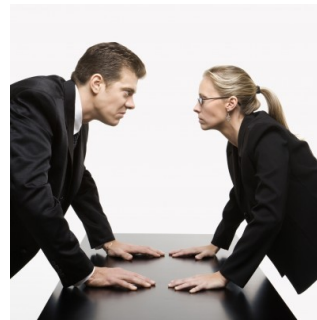
- Software Developer, Designer and Project Manager (13 years)
- QA and Test Manager (2 years)
- Consultant and Trainer in QA, Testing, SDLC, and other related topics (25 years)
- I've seen the good, the bad and the ugly!



3

## THE BRUTAL TRUTH

- Defining entry and exit criteria may be easy.
- Following the criteria may be difficult!
  - People (often in high positions) may want to reduce or bypass the criteria as the delivery deadlines get closer.
  - Testers may want to hold firm, but get over-ridden by stakeholders with more influence.



4

# The Ins and Outs of Entry and Exit Criteria

3/26/15

***The challenge with exit criteria is that despite all the good intentions, people on projects tend to find ways to justify releasing items before they are ready to be released.***

***The most common driver of this effect is the release schedule.***

*“The bitterness of poor quality remains long after the sweetness of meeting the schedule has been forgotten.” (Source Unknown)*

5

## MY OBSERVATION

- In 100% of the projects I have worked with, where exit criteria has been reduced or bypassed over the objections of the test team, the results have been one or more of the following:
  - High levels of defects experienced by users/customers
  - Higher costs to fix the defects in production
  - High numbers of residual defects in the product
    - Some defects never get fixed.
  - Loss of business, negative publicity and trust in the company
  - In some cases, the releases have been de-installed
  - In some cases, complete project failure and the firing of high-level IT management and/or contractor.

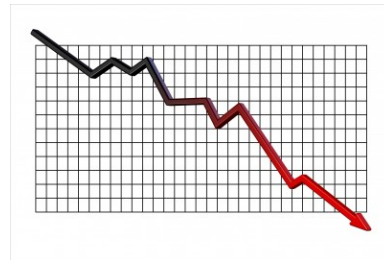
6

# The Ins and Outs of Entry and Exit Criteria

3/26/15

## REALITY CHECK #1

- No matter how good your testing and review efforts are, if you don't establish and follow good entry and exit criteria, your success in delivering quality software and systems will be greatly diminished.



7

## MY ASSERTION

- The ability to deal with entry and exit criteria in a healthy way is deeply rooted in human factors, such as:
  - Over-optimism that problems can be resolved post-release
  - Belief that customers won't mind a little inconvenience
  - Meeting the deadline trumps everything else
  - Over-promising delivery dates and scope to stakeholders
  - Stakeholders that demand delivery regardless of the quality of the product



8

# The Ins and Outs of Entry and Exit Criteria

3/26/15

## THE BOTTOM LINE

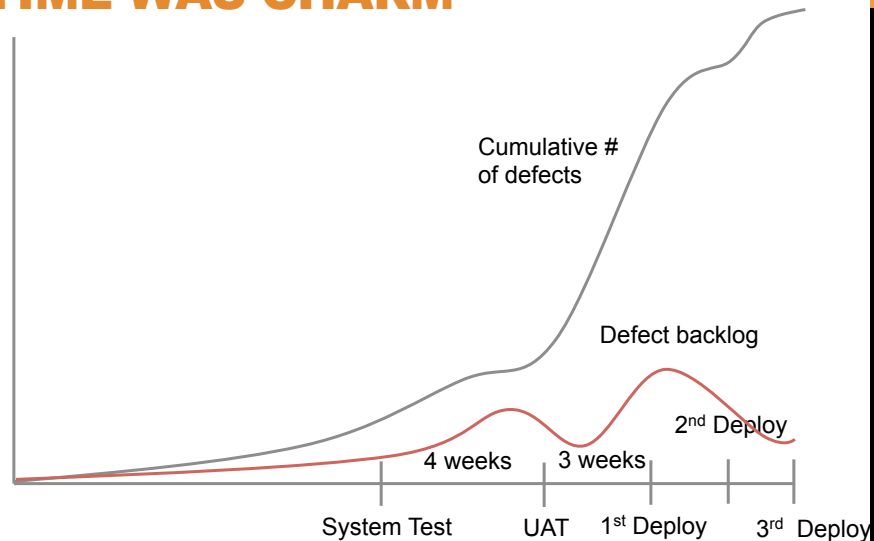
- “Cheating” the exit criteria doesn’t work and introduces high risk to the organization and its customers and stakeholders.
- While the release schedule is an important consideration for releasing software, it should not be the only criterion.
- “Just because testing stops, doesn’t mean problems stop.”

- William E. Perry



9

## EXAMPLE 1 – THIRD TIME WAS CHARM

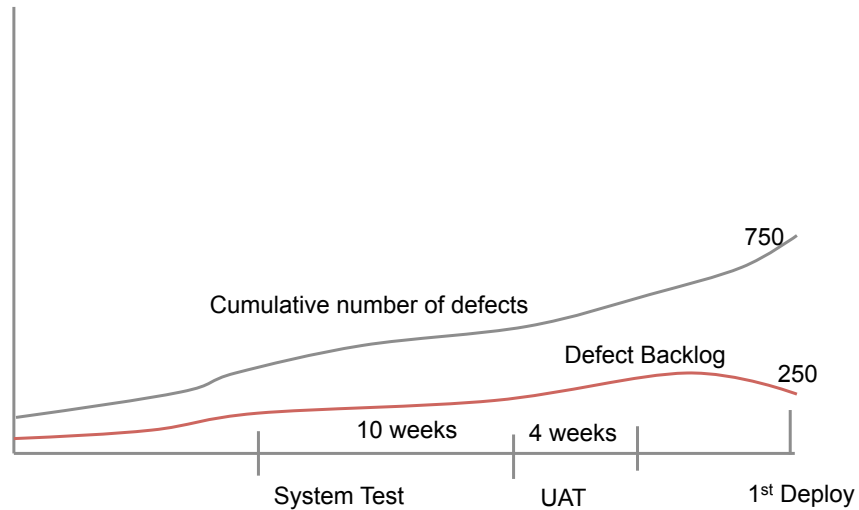


10

# The Ins and Outs of Entry and Exit Criteria

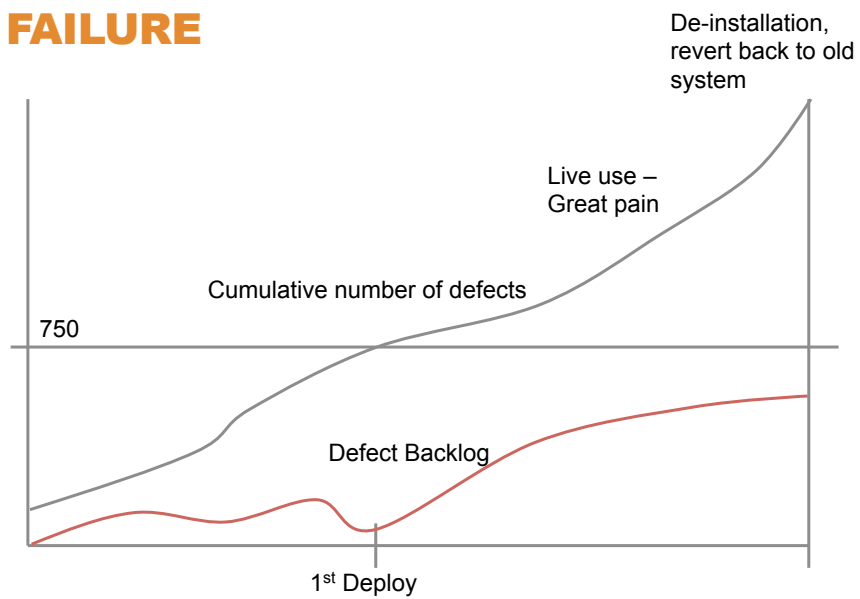
3/26/15

## EXAMPLE 2 – DEFECTS? WHAT DEFECTS?



11

## EXAMPLE 2 – PROJECT FAILURE



12

# The Ins and Outs of Entry and Exit Criteria

3/26/15

## THE POSITIVE SIDE

- If you can establish and follow reasonable entry and exit criteria, you have a much higher chance of delivering a project on time, within budget and with high levels of quality.



13

## ENTRY AND EXIT CRITERIA TRANSCEND LIFE CYCLE APPROACH

- These criteria are not just for sequential lifecycles!
- They apply to agile methods and Commercial Off-the-shelf products as well.



14

# The Ins and Outs of Entry and Exit Criteria

3/26/15

## WHAT ARE ENTRY AND EXIT CRITERIA?

- **Entry Criteria** - The set of generic and specific conditions for permitting a process to go forward with a defined task, e.g. test phase.
  - The purpose of entry criteria is to prevent a task from starting which would entail more (wasted) effort compared to the effort needed to remove the failed entry criteria. [Gilb and Graham]
    - ISTQB Glossary Version 1.5

15

## WHAT ARE ENTRY AND EXIT CRITERIA? (2)

- **Exit Criteria** - The set of generic and specific conditions, *agreed upon with the stakeholders* for permitting a process to be officially completed.
  - The purpose of exit criteria is to prevent a task from being considered completed when there are still outstanding parts of the task which have not been finished.
  - Exit criteria are used to report against and to plan when to stop testing.
    - [After Gilb and Graham]ISTQB Glossary Version 1.5

16



# The Ins and Outs of Entry and Exit Criteria

3/26/15

## EXAMPLE ENTRY CRITERIA

- **Component testing performed with 100% code coverage and 100% decision coverage.**
- **Integration testing performed with components that have interactions to the extent that all pairs of related conditions are tested.**
- **Performance testing of each system component must demonstrate that it meets or exceeds performance requirements.**
- **Test cases have been defined for each requirement that adequately verify the conditions described in the requirements.**

17

## EXAMPLE ENTRY CRITERIA (2)

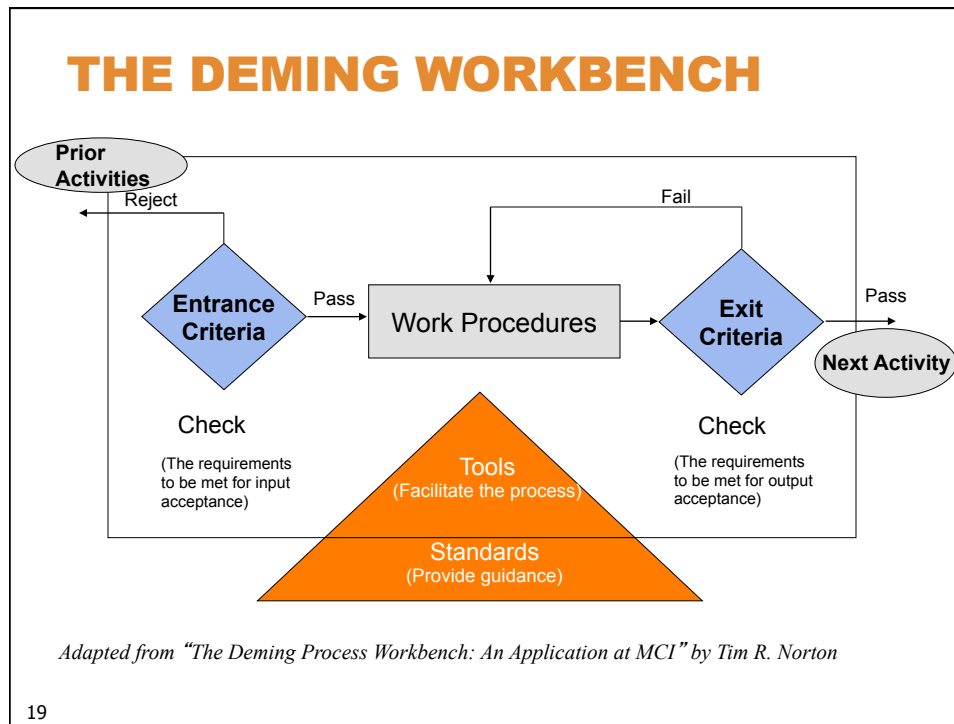
- **Project stakeholders have approved all test objectives.**
- **A risk assessment has been performed on the system and the tests have been prioritized based on relative risks.**
- **The test manager has approved promotion to the system test environment.**



18

# The Ins and Outs of Entry and Exit Criteria

3/26/15



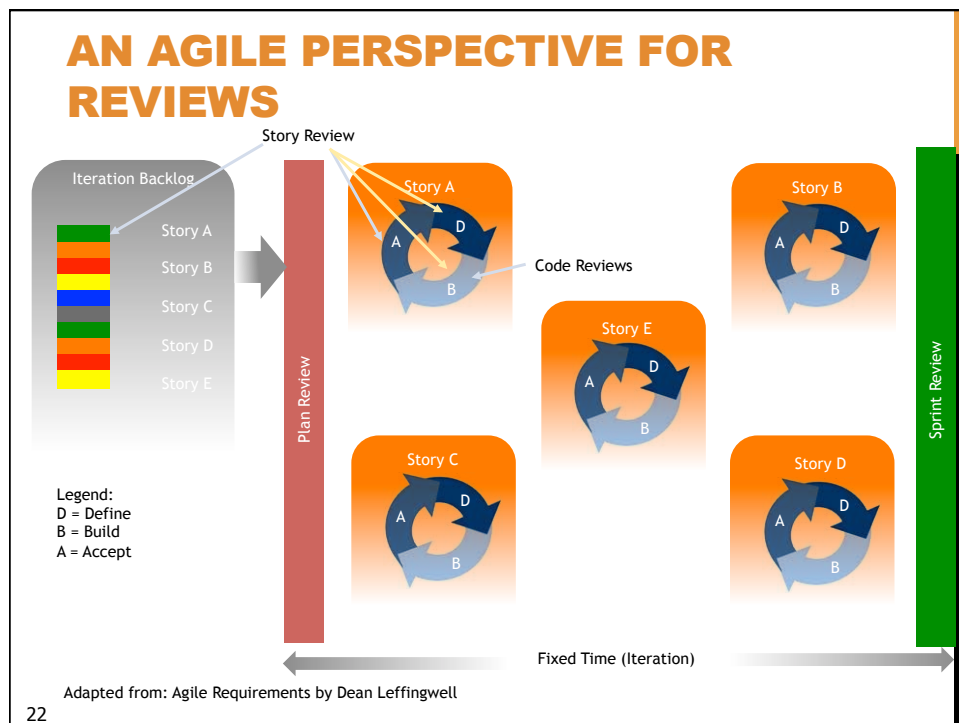
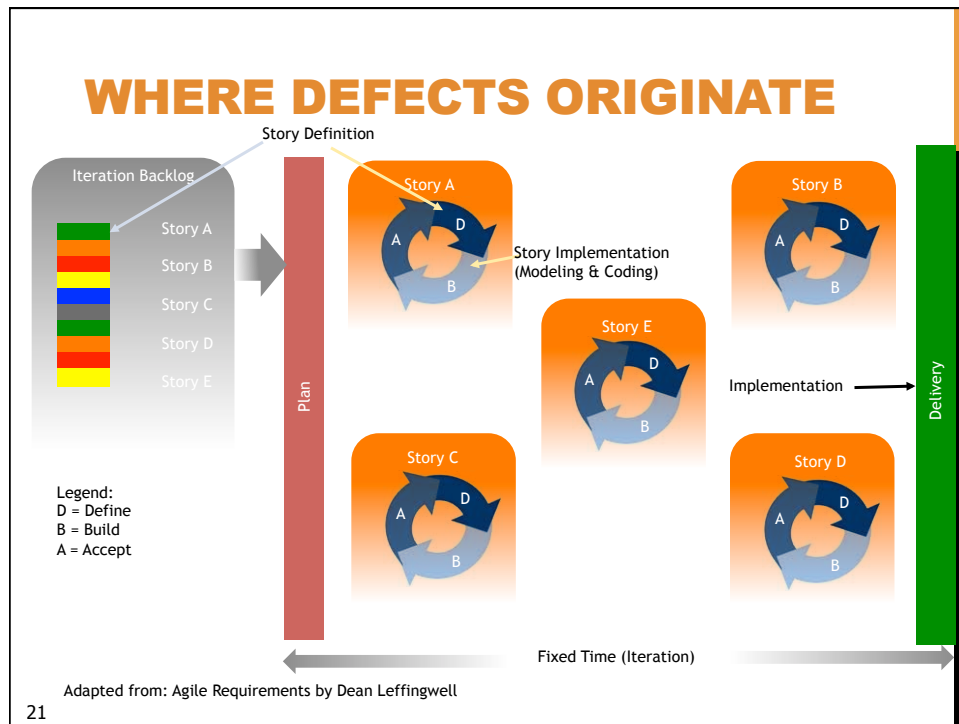
## FOR AGILE PROJECTS...

- **Think about how many times you have seen:**
  - Coding start on a flawed user story?
  - Code that passes test-driven development assertions, but fails to achieve important attributes such as usability or performance?
  - Integration that has to be re-worked or re-defined because the focus was on writing code?
  - Delivery of a feature that did not meet user needs, even though all the steps were followed in the sprint?

20

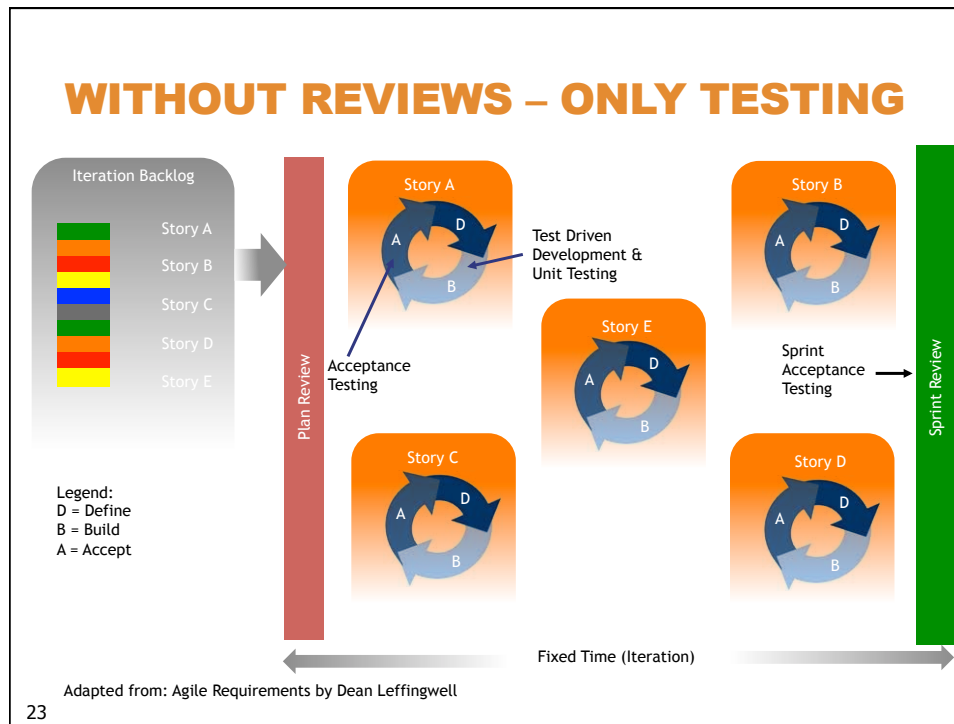
# The Ins and Outs of Entry and Exit Criteria

3/26/15



# The Ins and Outs of Entry and Exit Criteria

3/26/15



## TESTING ALONE IS NOT ENOUGH

- “Testing by itself without any pre-test inspections or static analysis is not sufficient to achieve high quality levels.”
- “However modern risk-based testing by certified test personnel with automated test tools who also use mathematically-derived test case designs and also tools for measuring test coverage and cyclomatic complexity can do a very good job and top 65% in defect removal efficiency for the test stages of new function test, component test, and system test.”

Capers Jones, *Software Defect Origins And Removal Methods*,  
<http://namcookanalytics.com/software-defect-origins-and-removal-methods/>

24

# The Ins and Outs of Entry and Exit Criteria

3/26/15

## EXAMPLE EXIT CRITERIA

- 100% of the planned tests have been performed.
- No critical defects remain unresolved.
- All major project risks have been mitigated or have contingencies.
- Technical support is trained and comfortable with being able to support the application in production.
- The defect discovery rate is less than 2 defects each test cycle.
- Regression testing is performed on every test cycle, including the final test cycle.
- All regression defects that are “Critical” or “Major” have been successfully resolved and retested

25

## HOW CAN WE DEFINE A GOOD SET OF ENTRY AND EXIT CRITERIA?

- **Consider:**
  - The requirements of your project processes and SDLC
  - The activity or activities being performed
  - The stakeholder needs and desires (negotiation may be required.)
  - Your resources – time, tools and people
  - The ability to measure the criteria
  - The relative risk
  - The ability to follow the criteria
  - Have an end-game process for when people may want to relax or strengthen the criteria at the end of a project.

26

# The Ins and Outs of Entry and Exit Criteria

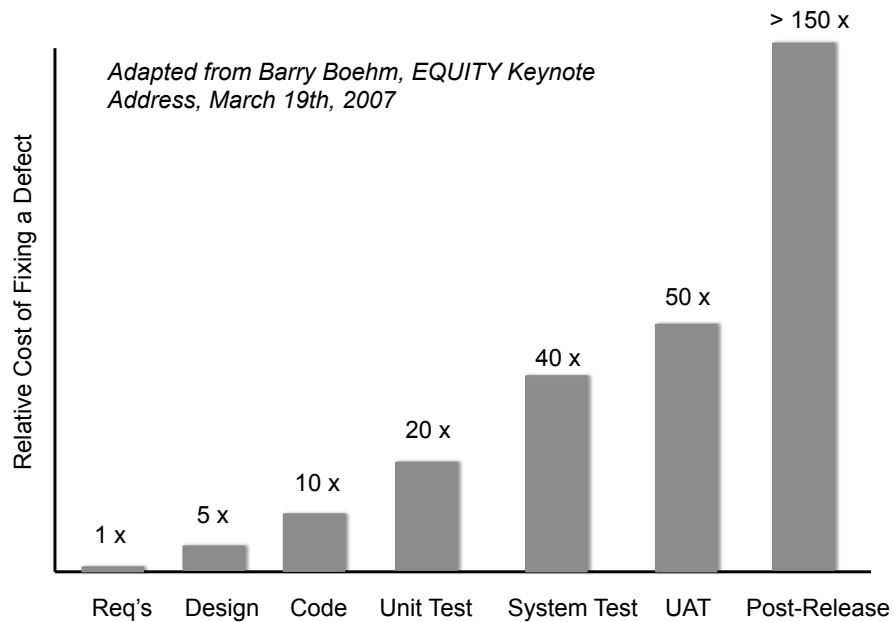
3/26/15

## THE MAIN PRINCIPLE – DEFECT CONTAINMENT

- When defects in software or other related products (requirements, user stories, design, models, etc.) are not found early, they propagate to become the basis for other defects as work is done in subsequent activities.
  - This is the reason for the 1:10:100 rule.



27



28

# The Ins and Outs of Entry and Exit Criteria

3/26/15

## ANOTHER MAJOR PRINCIPLE

- You need more than one set of entry and exit criteria.
- Because – You need more than one QC activity to find defects.
  - Think of these as “filters.”
- Plus, defects are just one thing you are looking for. Don't forget the non-functional attributes – usability, maintainability, performance, etc.



29

## THE COFFEE POT ANALOGY



**The Problem: Grounds in the Coffee**

**How to Solve the Problem?**

Solution #1 – Use tweezers and remove grounds one at a time.

Solution #2 – Use a better filter (or a series of filters)!

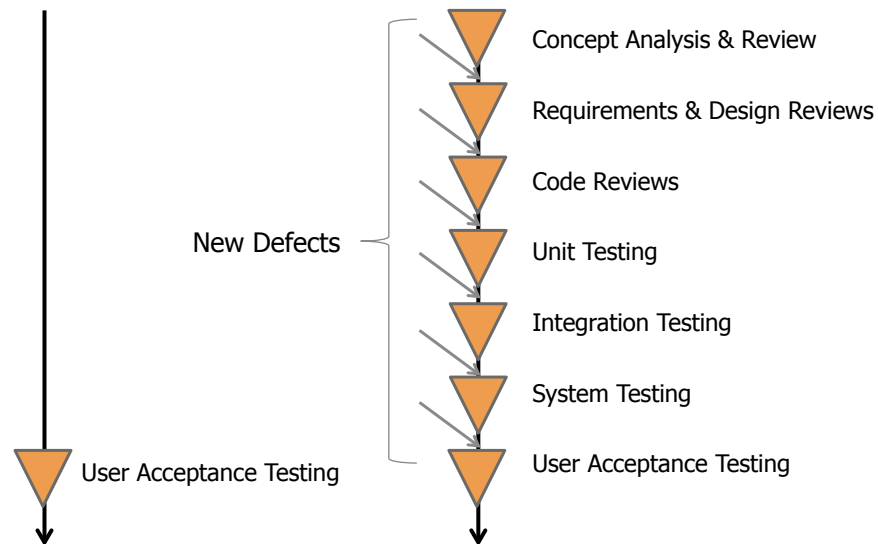
Source: Alka Jarvis and Vern Crandall, *Inroads to Software Quality*

30

# The Ins and Outs of Entry and Exit Criteria

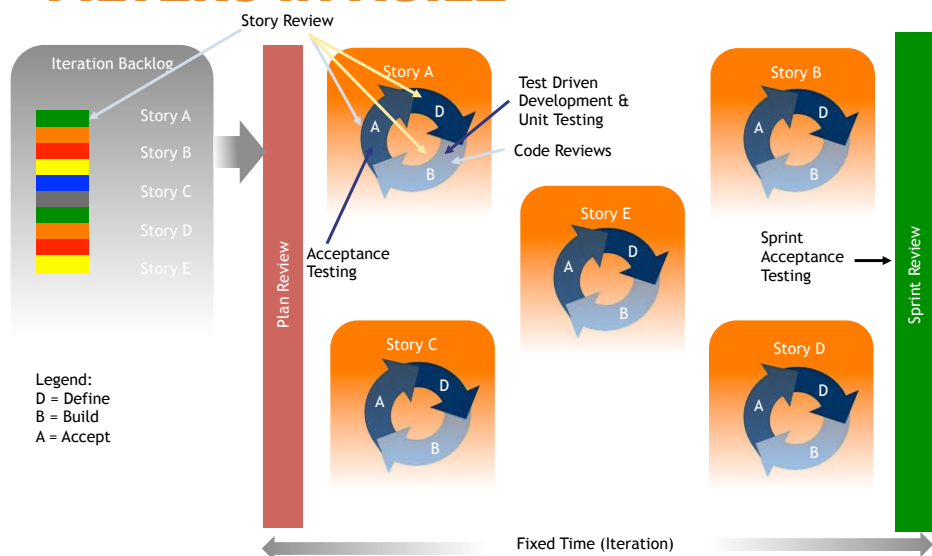
3/26/15

## THE EFFECT OF MULTIPLE FILTERS



31

## FILTERS IN AGILE



Adapted from: Agile Requirements by Dean Leffingwell

32



# The Ins and Outs of Entry and Exit Criteria

3/26/15

## THE COST OF FINDING DEFECTS - AN EXAMPLE

Big Bang Approach		Lifecycle Approach	
Accumulated Test Cost	Accumulated Defects per 1KLOC	Accumulated Defects per 1KLOC	Accumulated Test Cost
0	20	10	10
		Cost to find = \$1	
0	40	15	25
		Cost to find = \$1	
0	60	18	42
		Cost to find = \$1	
480	12	4	182
		Cost to find = \$10	
<b>1,680</b>	0	0	<b>582</b>
		Cost to find = \$100	

Adapted from *Effective Methods for Software Testing* by William E. Perry

33

## THE BIG QUESTION: WHEN TO STOP TESTING?

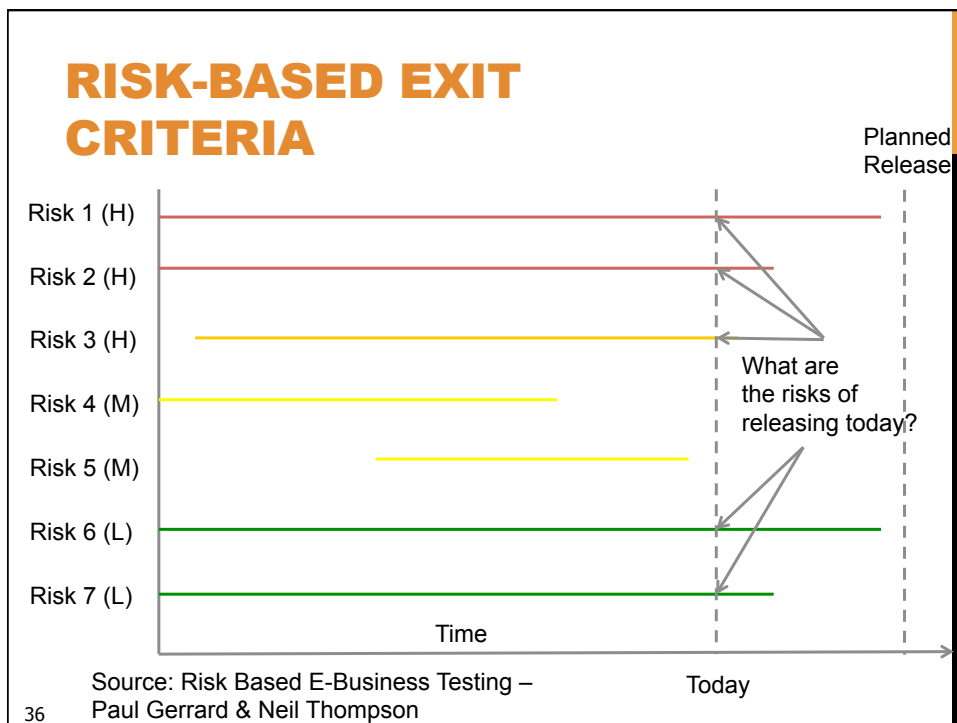
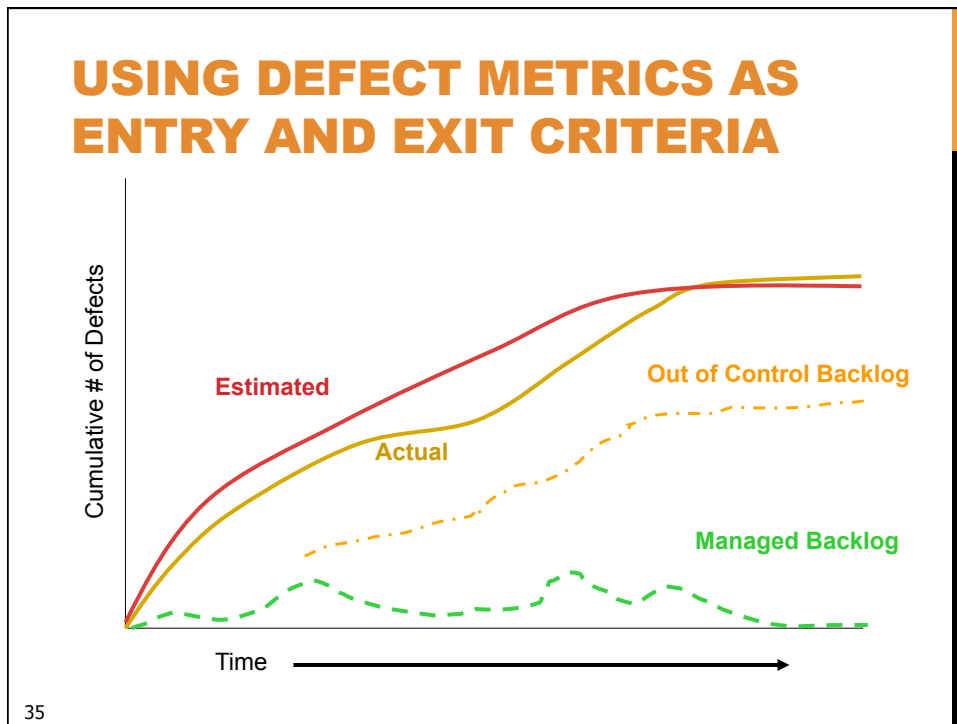
- “As part of results reporting and exit criteria evaluation, the Test Manager can measure the degree to which testing is complete.
- This should include tracing test cases and discovered defects back to the relevant test basis.
- For example, in risk-based testing, as tests are run and defects found, testers can examine the remaining, residual level of risk.
  - This supports the use of risk-based testing in determining the right moment to release.
- Test reporting should address risks covered and still open, as well as benefits achieved and not yet achieved.”

ISTQB Advanced Test Manager syllabus

34

# The Ins and Outs of Entry and Exit Criteria

3/26/15



# The Ins and Outs of Entry and Exit Criteria

3/26/15

## HOW EFFECTIVE IS YOUR CRITERIA?

- “Finally, during test closure, the Test Manager should evaluate metrics and success criteria which are pertinent to the needs and expectations of the testing stakeholders, including the customers’ and users’ needs and expectations in terms of quality.
- Only when testing satisfies these needs and expectations can a test team claim to be truly effective.”

ISTQB Advanced Test Manager syllabus

37

## HELPFUL METRICS

- ***Defect Detection Percentage (DDP)***
  - This is the number of defects found in your organization divided by the total number of defects found, including those found after release by the users.
  - This is a purely historic metric, but helps you to know how effective your testing is, and if you are improving or not.

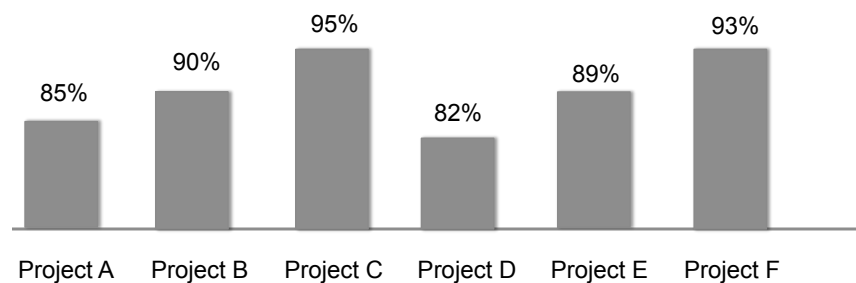
38

# The Ins and Outs of Entry and Exit Criteria

3/26/15

## DDP EXAMPLE

$$\frac{85 \text{ Defects found in our testing}}{85 \text{ Defects found in our testing} + 15 \text{ Defects found after release}} = \frac{85}{100} = 85\%$$



39

## NOT ALL DEFECTS FOUND ARE FIXED

- **Defect Fixed Percentage (DFP) or Defect Removal Efficiency (DRE)**
  - This is the number of defects found and fixed in your organization divided by the total number of defects found and fixed, including those found after release by the users.
  - Like DDP, this is in hindsight, but helps you to know how effective your testing is, how well the defects are being fixed, and if you are improving overall or not in finding and fixing defects.

40

# The Ins and Outs of Entry and Exit Criteria

3/26/15

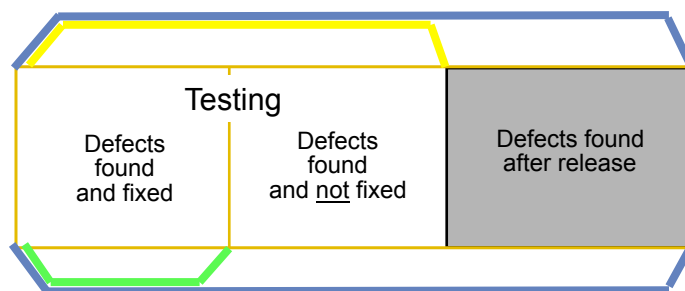
## A DISTINCTION IN TERMS

- **Defect Removal Efficiency (DRE), has also been called Defect Detection Percentage (DDP).**
  - In this presentation, I am using the term DFP – Defect Fixed Percentage.
- **An issue that can be found in testing literature is the distinction of using these metrics as ways to determine test effectiveness as opposed to test efficiency.**
  - For example, a case could be made that while a set of tests might be very efficient in terms of resources and coverage, those tests may be ineffective at finding defects.
- **I consider DRE and DDP to be measures of test effectiveness, not efficiency.**

41

## DDP VERSUS DFP

Defect Detection Percentage = 
$$\frac{\text{Defects found by testing}}{\text{all defects found}}$$



Defect Fix Percentage = 
$$\frac{\text{defects fixed before release}}{\text{all defects found}}$$

42

# The Ins and Outs of Entry and Exit Criteria

3/26/15

## INDUSTRY DATA

- “Successful quality control stems from a synergistic combination of defect prevention, pre-test defect removal, and test stages.
- The best projects in the industry circa 2012 combined defect potentials in the range of 2.0 defects per function point with cumulative defect removal efficiency levels that top 99%.
- The U.S. average circa 2012 is about 5.0 bugs per function point and only about 85% defect removal efficiency.”

Capers Jones, *Software Defect Origins And Removal Methods*,  
<http://namcookanalytics.com/software-defect-origins-and-removal-methods/>

43

## WHY MEASURE DDP AND DRE?

- “When companies that do not measure DRE are studied by the author during on-site benchmarks, they are almost always below 85% in DRE and usually lack adequate software quality methodologies.”

Capers Jones, *Software Defect Origins And Removal Methods*, <http://namcookanalytics.com/software-defect-origins-and-removal-methods/>

44

# The Ins and Outs of Entry and Exit Criteria

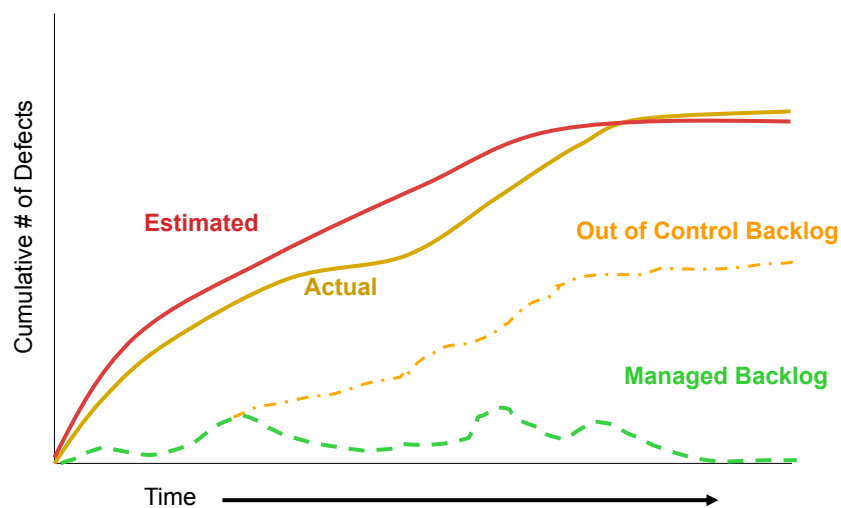
3/26/15

## MEAN TIME BETWEEN FAILURE (MTBF)

- This metric can be applied in various levels of testing, but the idea is to know how long do you test (as a team) before you find a defect.
- At first, the times are normally short – sometimes within minutes.
- Then, you measure when the next failure is seen. Once you get to the point where you are testing for days and only finding a few minor defects, your MTBF is probably a day or more and your tests have pretty much done their job.
- The defect discovery curve has leveled out. An example is seen in the following slide.

45

## USING DEFECT METRICS AS ENTRY AND EXIT CRITERIA



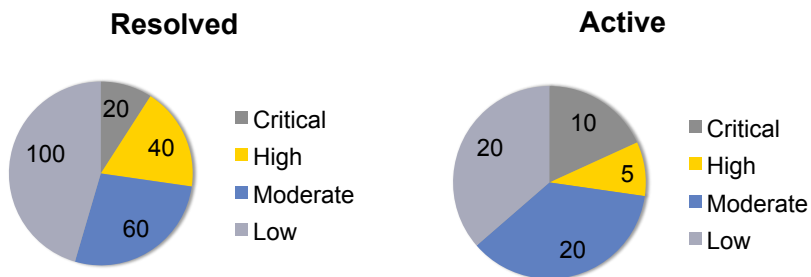
46

# The Ins and Outs of Entry and Exit Criteria

3/26/15

## NUMBER OF DEFECTS RESOLVED BY SEVERITY LEVEL

- This is a very important metric because it tells how known critical defects have been resolved and re-tested.



47

## NUMBER OF DEFECTS OUTSTANDING BY STATUS

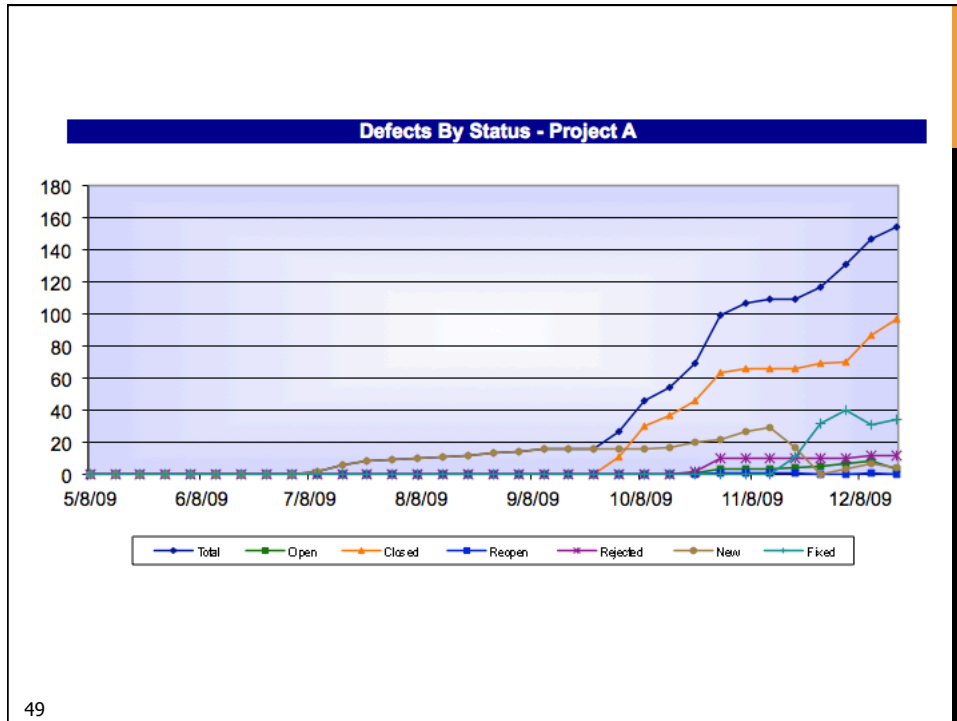
- Likewise, this metric is important because it tells how many known critical defects have not been resolved.
- Ideally, this number should be zero.
  - This is not to say you have zero defects in the application, but that you have no *known* critical defects.

48



## The Ins and Outs of Entry and Exit Criteria

3/26/15



## PERCENTAGE OF TESTS THAT EVENTUALLY PASS

- You want all your tests to pass eventually.
- However, some tests that reveal minor defects may go unresolved before release.
- The danger here is “death by a thousand paper cuts” in which too many minor defects can have an overall devastating effect.

50

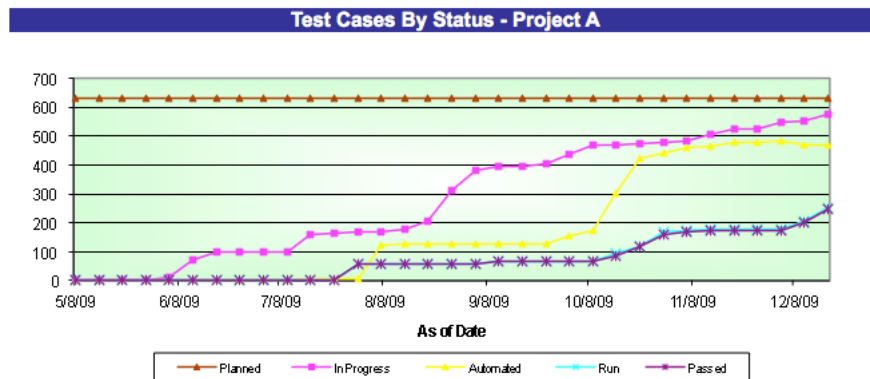
# The Ins and Outs of Entry and Exit Criteria

3/26/15

## NUMBER OF TESTS THAT CONTINUE TO FAIL

- Once again, you must consider the severity level of the failures.
- So, it is good to know which percentage of continued failures are due to major defects versus minor ones.

51



52

# The Ins and Outs of Entry and Exit Criteria

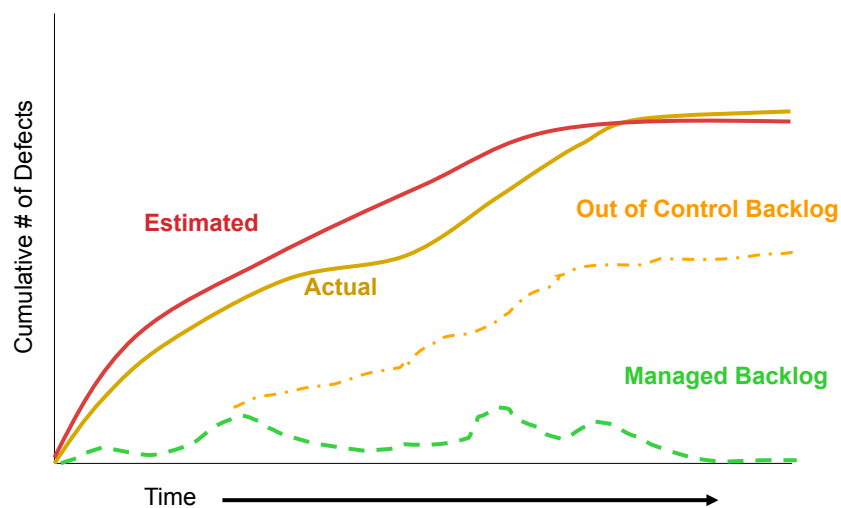
3/26/15

## THE DEFECT BACKLOG

- One of the best ways to tell if a product or system is ready for release is to look at the size of the defect backlog.
- The defect backlog is the count of defects assigned to development or other area for resolution.
- When the defect backlog is constantly increasing, it means that testers are finding defects faster than they can be resolved.
- Plus, each resolved defect must be re-tested. In extreme cases, re-testing might mean repeating the entire test.

53

## OUT OF CONTROL DEFECT BACKLOG

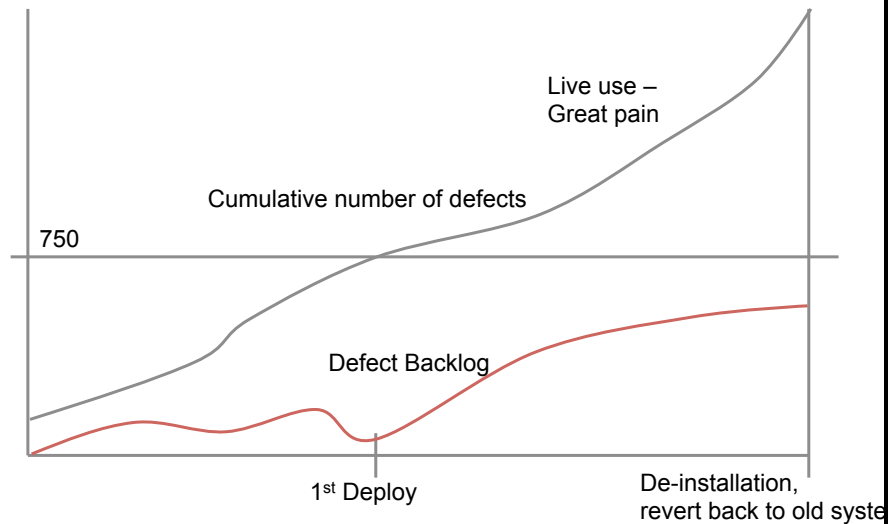


54

## The Ins and Outs of Entry and Exit Criteria

3/26/15

### EXAMPLE 2 – PROJECT FAILURE



### REALITY CHECK #2

- When software is released in face of a climbing defect backlog, the problems will be experienced by actual users.
- Although risky, some companies have released software in this condition.
- The result is almost always very costly and sometimes results in de-installing the release.

56

# The Ins and Outs of Entry and Exit Criteria

3/26/15

## BEST PRACTICES FOR RELEASING SOFTWARE

- **Have a defined and orderly release process.**
  - This is even more essential when you have many people working in separate areas, all trying to get their work ready for release.
- **Have well-defined exit criteria that align with the risk of the project.**
- **Perform a risk assessment just before release.**
- **Perform a pre-implementation review/walkthrough (this could be checklist-driven) to make sure everything is in place.**

57

## BEST PRACTICES FOR RELEASING SOFTWARE (2)

- **Perform configuration testing in a test environment or environments that closely mirror the production environment.**
- **Deploy first to a smaller, low risk production environment as a pilot or beta project.**
  - This reduces the risk of a large-scale failure. If something does go wrong, the damage can be contained.

58

# The Ins and Outs of Entry and Exit Criteria

3/26/15

## WHERE MONEY IS WASTED

- “The software industry spends more money on finding and fixing bugs than for any other known cost driver.
- This should not be the case.
- A synergistic combination of defect prevention, pre-test defect removal, and formal testing can lower software defect removal costs by more than 50% compared to 2012 averages.
- These same synergistic combinations can raise defect removal efficiency (DRE) from the current average of about 85% to more than 99%.”

*Capers Jones, Software Defect Origins And Removal Methods, <http://namcookanalytics.com/software-defect-origins-and-removal-methods/>*

59

## CONCLUSION

- **Many people see the job of testers as that of finding defects.**
  - Actually, the job of testers is to find the evidence of possible defects by causing the software to fail.
  - The cause of the failure may or may not be a defect.
- **It is only when testers get involved in reviews and inspections that they have the opportunity to find defects because they can see the defects in the requirements, code, design, test cases, and other work products.**

60

# The Ins and Outs of Entry and Exit Criteria

3/26/15

## CONCLUSION (2)

- **Each organization must establish and evaluate its own criteria.**
  - Then, the project team must be accountable for how well it enforces those criteria.
  - Otherwise, even though the criteria are defined, they have little overall effect in assuring quality releases.



61

## THANKS FOR ATTENDING!

- **Slides can be found at:**
  - [Randallrice.blogspot.com](http://Randallrice.blogspot.com)
- **Thanks to reviewers and editor for the article:**
  - Taz Daughtrey
  - Frank Roland
  - Dave Stevens



62

# The Ins and Outs of Entry and Exit Criteria

3/26/15

## BIO - RANDALL W. RICE

- Over 35 years experience in building and testing information systems in a variety of industries and technical environments
- ASTQB Certified Tester – Foundation level, Advanced level (Full)
- Director, American Software Testing Qualification Board (ASTQB)
- Chairperson, 1995 - 2000 QAI's annual software testing conference
- Co-author with William E. Perry, *Surviving the Top Ten Challenges of Software Testing and Testing Dirty Systems*
- Principal Consultant and Trainer, Rice Consulting Services, Inc.



63

## CONTACT INFORMATION

**Randall W. Rice, CTAL**  
**Rice Consulting Services, Inc.**  
**P.O. Box 892003**  
**Oklahoma City, OK 73189**  
**Ph: 405-691-8075**  
**Fax: 405-691-1441**  
**Web site: [www.riceconsulting.com](http://www.riceconsulting.com)**  
**e-mail: [rrice@riceconsulting.com](mailto:rrice@riceconsulting.com)**



64