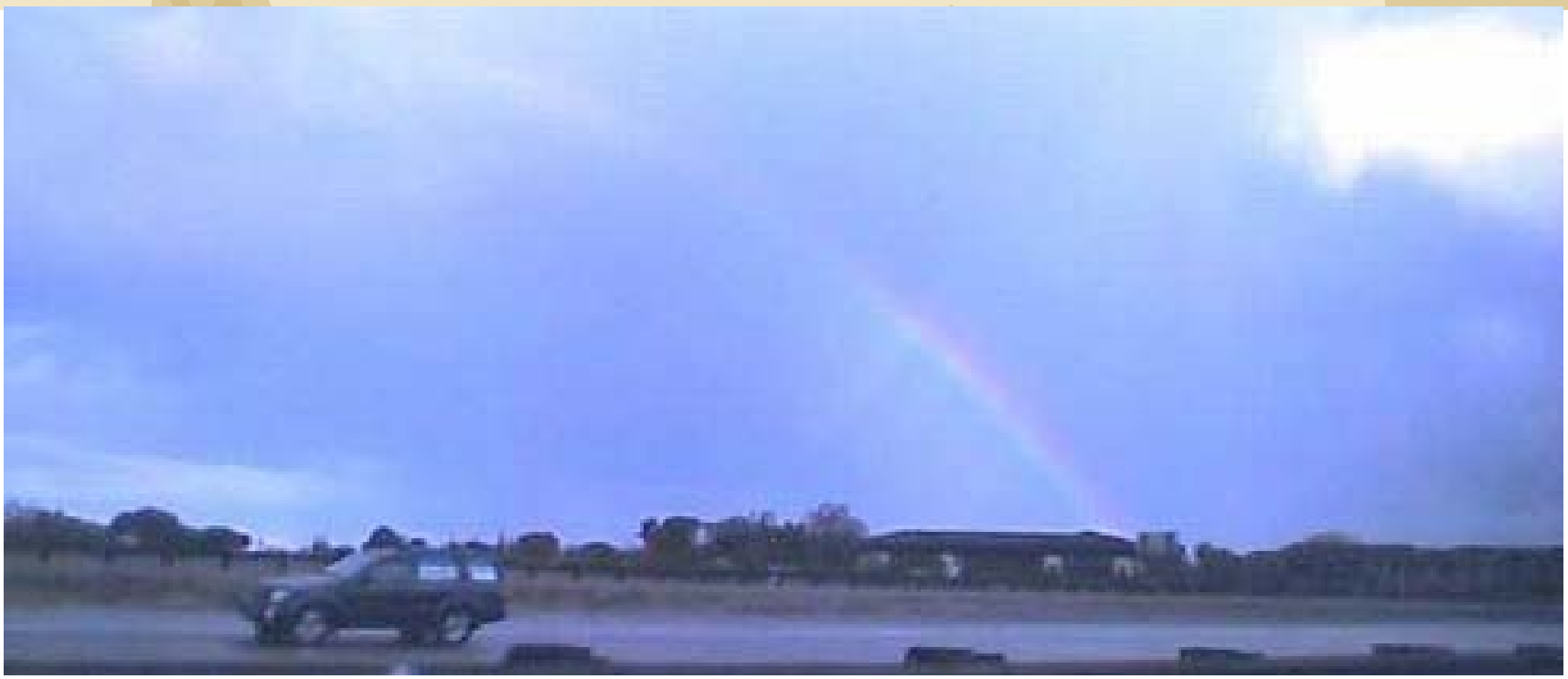
Agile Testing Challenges

Successful Testing on Agile Projects



RBCS

TIME TESTED.
TESTING IMPROVED.

www.RBCS-US.com

Agile Testing Challenges

- Agile lifecycles are becoming common
- What test strategies work well with Agile methodologies?
  - Risk-based testing
  - Automated testing, including functional regression testing
  - Reactive testing
  - Some test strategies work less well
- Even properly-chosen test strategies do not alleviate all the testing challenges of Agile projects
- Every lifecycle affects testing, and Agile methodologies create both benefits and challenges
- Let's look at the Agile challenges…

Volume and Speed of Change

- An Agile principle: welcome changing requirements, even late in development
  - Risk-based testing accommodates change
  - Careful automated testing can accommodate change, though GUI-based tests are often more sensitive
  - Lightweight reactive testing also accommodates change
- Testing challenges still arise from changes in the definition of the product and its correct behavior
  - Keep the test team informed of such changes
  - Account for differential impact of changes on testing
- Otherwise, changes can impose testing inefficiencies

Remaining Effective in Short Iterations

- Sequential lifecycles can provide test teams with a long period to develop and maintain their tests (though this doesn't always happen)
- Some iterative lifecycle models allow substantial periods of time between each iteration
- Agile methodologies move faster
- The pace and brevity further squeeze the test team's ability to develop and maintain tests
- GUI test automation techniques can be particularly sensitive
- Risk-based testing strategies also help, because of the focus on the important areas, allowing test teams to develop, maintain, and execute tests in risk priority order

Inconsistent or Inadequate Unit Testing

- Agile methodologies stress good, automated unit testing
- However, we still see two problems
  - Unit testing has limited bug-finding effectiveness, averaging 25 to 30%, while good system testing averages around 85%
  - Not all programmers do unit testing
- The short test execution periods on Agile sprints can compound the damage from highly buggy code on system test
- So, good unit testing is necessary—but not sufficient by itself

Increased Regression Risk

- In Agile lifecycles, code that worked in previous sprints can be affected by new features in subsequent sprints, increasing the risk of regression
- Good unit testing helps but doesn't remove all regression risk
- Therefore, we need effective regression testing at the system test level
- Combine risk-based testing with maintainable automated regression testing at the system test level
- Long-term test automation tasks (e.g., building frameworks) should happen outside of the sprints

Poor, Changing, and Missing Test Oracles

- Some people misinterpret Agile principles to mean: no documentation
- In these cases, testers on Agile projects receive insufficient test oracles
- Even with adequate test oracles, two Agile principles keep the challenge alive
  - Embrace change
  - Convey information in face-to-face conversations
- Some teams change the definition of correct behavior after test execution starts, and this isn't always communicated to the test team
- Test strategies cannot solve this; change management is required
- Test oracle problems impose test inefficiencies around 20-30%
- Test oracle problems reduce bug-finding effectiveness and can frustrate the testers

A Shifting Test Basis

- Requirements-based testing strategies cannot handle bad requirements specifications, because they require these as a test oracle and a test basis
- The test basis also provides a means to measure coverage and report results
- Risk-based testing evades both problems, since quality risk items provide the test basis
- The level of risk determines the number of test cases and the priority of the test cases
- The test team can report in terms of quality risk mitigation

Meeting Overload

- In some organizations, the Agile focus on lightweight documentation and face-to-face communication leads to a dramatic increase in meetings involving most leads and managers, reducing effectiveness and efficiency
- One manager said, "I'm surprised at the name *Agile*—it should be called *couch potato*. There are too many meetings. It's ironic that there are all these books explaining how simple it is."
- Excessive meetings can happen in any lifecycle
- Every organization, every project, and every lifecycle has to strike the right balance between documentation and meetings
- Embracing change should not mean analysis paralysis, with meetings endlessly revisiting previous decisions

Over-commitment and Sprint Durations

- Some Agile projects ritualize some rules, particularly sprint time deadlines, while not following other rules related to sustainable workload
- For example, if a project team continually over-commits, the test team gets squeezed on the last weekend of every sprint
- Fully resolving this challenge requires team and management maturity
- Risk-based testing can help the test team deal with over-commitment
  - Reduce the scope of testing based on risk
  - Slip low-risk tests into the next sprint

Blind Spots in the Sprint Silos: Organization

- Many RBCS clients adopting Agile methodologies have retained independent testing, which we continue to recommend
- One approach is to partition (some of) the test team across the sprints and create a dotted-line report to the sprint leader
- This helps, but even so there are still sprint silo challenges…

In the Sprint Silo

- The tester focuses entirely on sprint-related tasks
- The tester allocates and reallocates time based on (changing) sprint goals
- The sprint leader might re-direct the tester without consulting the test manager
- Some sprint leaders call on the tester work overtime to hit sprint deadlines
- The test effort for the sprint does not vary once the number of testers is determined

Sprint Silos Challenges

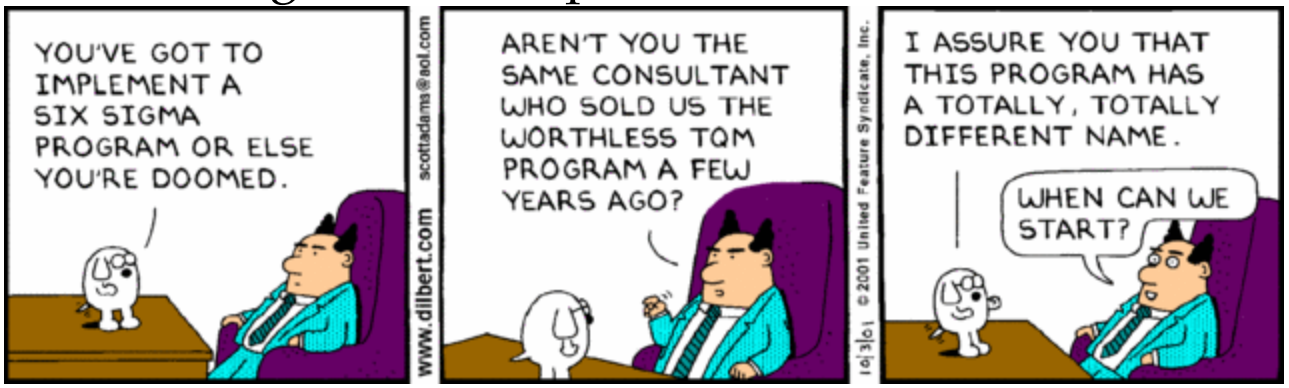
- The tester in the sprints sometimes lose independence
- If all testers are embedded in sprint teams:
  - The test team has reduced system-level perspective
  - Some testers make mistakes related to gaps and overlaps
  - Reduced ability to grow a consistent, powerful, maintainable test system
- The test manager can lose the ability to manage the workload of test resources, leading to low morale and higher turnover
- Some teams use a separate test sprint after each development sprint, but it can results in separating the test team from the developers, exchanging one silo for another

Dealing with Sprint Silos

- The test manager (of independent team) should introduce centripetal forces that bind the team together and makes its actions consistent, balancing the sprint-specific centrifugal forces
- Periodic stabilization sprints seem to help
- Separate groups of testers, outside of the sprint teams:
  - Work on longer-term projects (e.g., test automation)
  - Maintain a global perspective
  - Handle integration testing of sprint deliverables across the sprints

Gartner's Hype Cycle

- Technology Trigger: Idea generates significant press and interest
- Peak of Inflated Expectations: Frenzy of publicity typically generates over-enthusiasm and unrealistic expectations
- Trough of Disillusionment: Failure to meet expectations makes technique unfashionable
- Slope of Enlightenment: Some businesses start to see benefits and practical application
- Plateau of Productivity: The benefits become widely demonstrated and accepted
- In some organizations, Agile is at the peak of inflated expectations



Managing Agile Expectations

- Some test teams on Agile projects report to management that…
  - Quality is not higher, and perhaps is even lower
  - They are challenged by the issues in this presentation
  - They can't tolerate unlimited, unmanaged change
- This can create cognitive dissonance in the managers
  - Ultimately, these cognitive dissonance experiences will push these approaches along the Hype Cycle
  - In the short run, management might blame testing for the problems
- Testers and test managers must help their organizations understand and manage the challenges to manage expectations

Conclusions

- In 2002 I wrote, "The onus is on us as professional testers to help develop a proper role for systematic but lightweight test processes and adapt the best practices of testing within the context of Agile methods. Hopefully, this will happen before the end of this decade."
- There's still work to do but Agile methodologies create testing opportunities as well as challenges
- Good test strategies support the goals of Agile methodologies
  - Risk-based testing supports increased quality, increased productivity, and flexibility
  - Maintainable automated regression testing contains the regression risks associated with Agile methodologies
  - Reactive testing allows testers to explore areas that risk-based testing and automated regression testing might miss
- Good test strategies alone cannot fully resolve the Agile challenges, so we need to continue to work with Agile practitioners to fit testing properly

…Contact RBCS

For almost 20 years, RBCS has delivered consulting, outsourcing and training services to clients, helping them with software and hardware testing. Employing the industry's most experienced and recognized consultants, RBCS advises its clients, trains their employees, conducts product testing, builds and improves testing groups, and hires testing staff for hundreds of clients worldwide. Ranging from Fortune 20 companies to start-ups, RBCS clients save time and money through improved product development, decreased tech support calls, improved corporate reputation and more. To learn more about RBCS, visit www.rbcs-us.com.

Address: RBCS, Inc.
31520 Beck Road
Bulverde, TX 78163-3911
USA
Phone: +1 (830) 438-4830
Fax: +1 (830) 438-4831
E-mail: info@rbcs-us.com
Web: www.rbcs-us.com