

Test Automation Blunders

Prepared and presented by

Dorothy Graham

email: info@dorothygraham.co.uk

Twitter: [@DorothyGraham](https://twitter.com/DorothyGraham)

www.DorothyGraham.co.uk
www.TestAutomationPatterns.org
© Dorothy Graham 2018

DG

1

Blunder

- from old Norse word “*blundra*”
 - meaning “to shut one’s eyes”
- now means
 - mistake caused by ignorance, carelessness
 - or not thinking things through
- people blunder when they don’t see or understand

DG

2

Contents

- Test automation blunders
 - Testing-Tools-Test
 - Who needs GPS?
 - Silver Bullet
 - The Wrong Thing
 - How hard can it be?
 - Stable Application Myth
 - Project / Not Project
 - Inside the Box
 - Isolationism
- Conclusion

Twitter: @DorothyGraham

Testing-Tools-Test

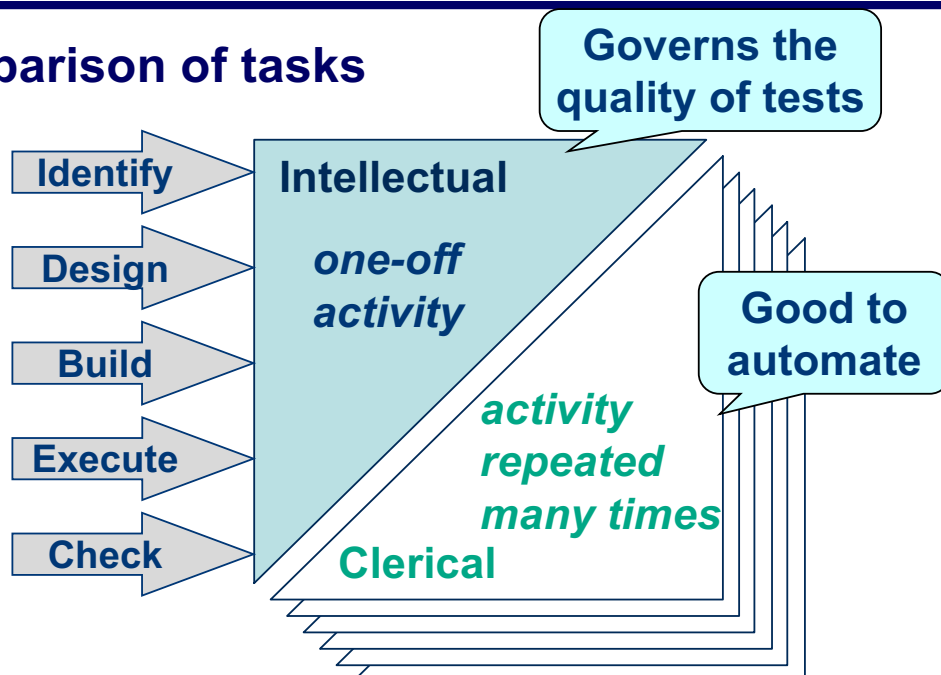
- blunder: thinking that tools actually do testing
 - most important, at the root of other blunders
- shouldn't have been called "testing tools"
- better names:
 - "tester assistance tools"
 - "check-running tools"
 - "test execution tools"
 - "test running tools"
 - "partial-support-for-some-aspects-of-test-execution-and-comparison tools"

People (testers) vs tools

- what do people do?
 - think, evaluate, assess, decide, observe, interpret
 - recognize patterns, have new ideas, find bugs
 - make mistakes
- what do tools do?
 - they just run stuff - whatever they've been programmed to execute (including bad tests)
 - intelligence level: zero

*Get tools to do what computers do best,
get testers to do what people do best*

Comparison of tasks



Contents

- Test automation blunders
 - Testing-Tools-Test
 - ➔ – Who needs GPS?
 - Silver Bullet
 - The Wrong Thing
 - How hard can it be?
 - Stable Application Myth
 - Project / Not Project
 - Inside the Box
 - Isolationism
- Conclusion

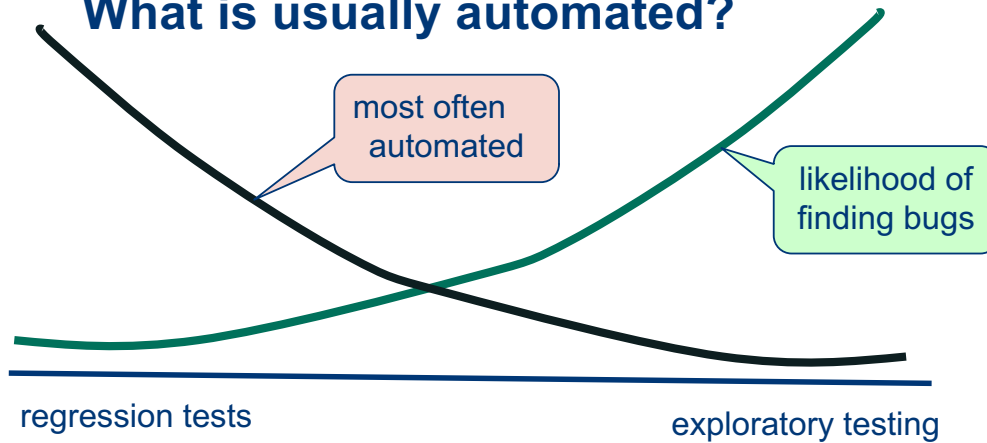
Twitter: @DorothyGraham

Who needs GPS?

- if you don't know where you are going, any road will do (Lewis Carroll)
- where are you going with your automation?
 - testing and automation are different activities
 - different activities require different objectives
- good objectives for testing?
 - find bugs, gain confidence, investigate
- good objectives for automation?
 - Hint: they shouldn't be the same!

What finds most bugs?

What is usually automated?



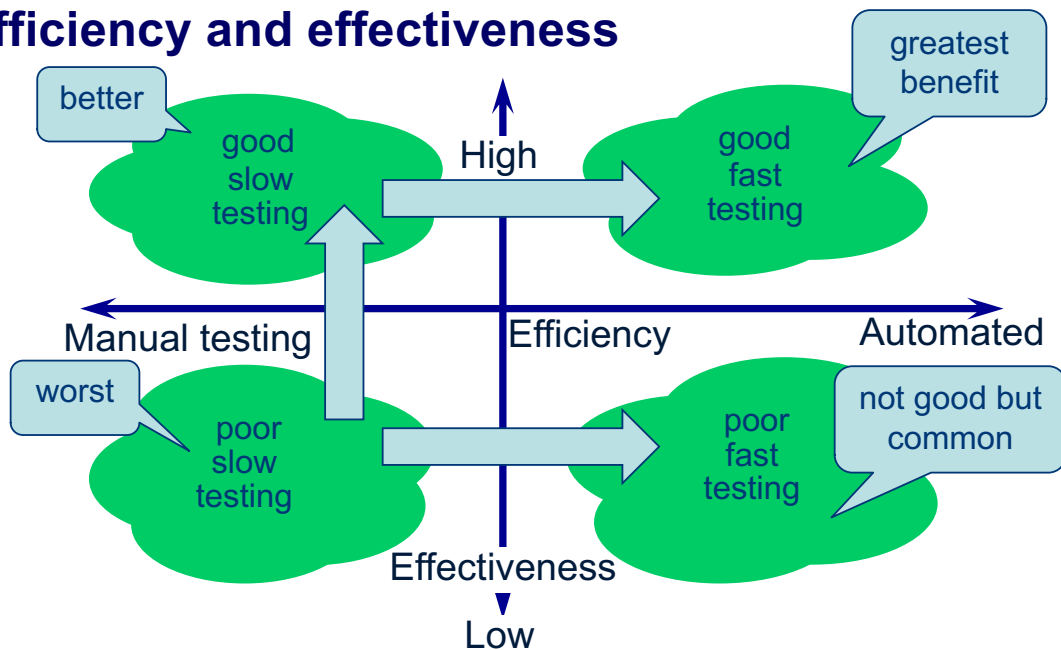
Automation success = find lots of bugs?

- tests find bugs, not automation
- automation is a mechanism for running tests
- the bug-finding ability of a single test is not affected by the manner in which it is executed
- “find bugs” can be a dangerous objective
 - especially for regression automation!

Automated tests	Manual Scripted	Exploratory	Fix Verification
9.3%	24.0%	58.2%	8.4%



Efficiency and effectiveness



DG

11

Contents

- Test automation blunders
 - Testing-Tools-Test
 - Who needs GPS?
 - – Silver Bullet
 - The Wrong Thing
 - How hard can it be?
 - Stable Application Myth
 - Project / Not Project
 - Inside the Box
 - Isolationism
- Conclusion

Twitter: @DorothyGraham

DG

12

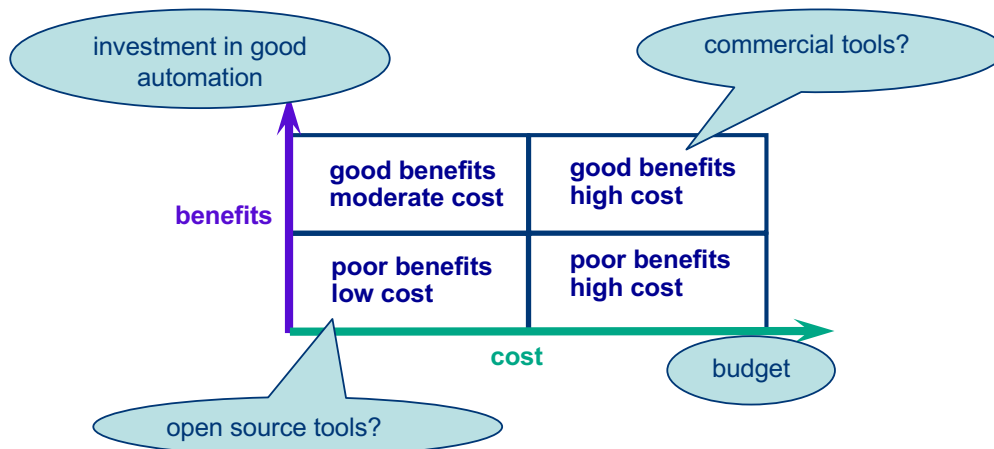
Silver bullet solution: get the right tool

- no such thing as “the right tool” or “best tool”
– what’s “the best car”?



Silver bullet solution: get the right tool


- no such thing as “the right tool” or “best tool”
– what’s “the best car”?



Silver bullet: success is automatic

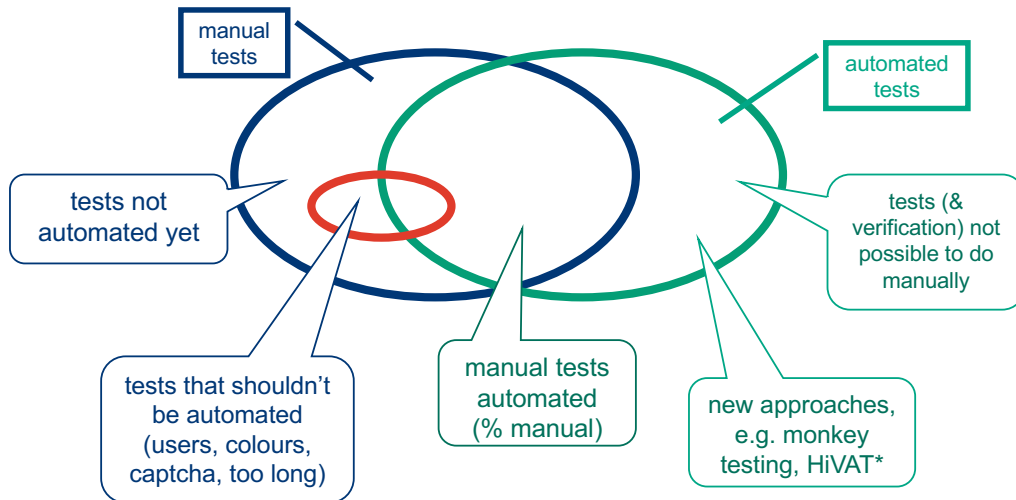
- automation is (much) more than just a tool
- it takes time and effort to succeed
 - building good automation is a learning process
- management support is critical
 - high level managers need to understand automation capability & limitations, and have realistic expectations and budget
 - “people issues” – people use the automation, people develop the automation

Contents

- Test automation blunders
 - Testing-Tools-Test
 - Who needs GPS?
 - Silver Bullet
 -  – The Wrong Thing
 - How hard can it be?
 - Stable Application Myth
 - Project / Not Project
 - Inside the Box
 - Isolationism
- Conclusion

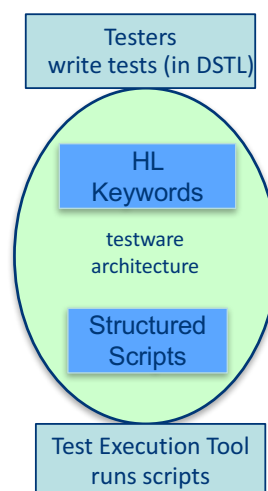
Twitter: @DorothyGraham

Automate x% of tests?



Testware architecture

- poor architecture gives high maintenance cost
 - most frequent cause of abandoned automation / shelfware
- two layers of abstraction
 - technical: for long life
 - human: for wide use
- using the tool's architecture ties you to that tool (version)



Ch 5

Contents

- Test automation blunders
 - Testing-Tools-Test
 - Who needs GPS?
 - Silver Bullet
 - The Wrong Thing
 - – How hard can it be?
 - Stable Application Myth
 - Project / Not Project
 - Inside the Box
 - Isolationism
- Conclusion

Twitter: @DorothyGraham


How hard can it be?

- different activities require different skills
- classic blunder: let the testers automate
 - automating without automation skills?
- newer blunder: let the automators write tests
 - testing without testing skills?
- if testers are automators → a conflict of interest
 - do you run tests or do you automate tests?
 - automation is better long-term, BUT
 - deadline pressure pushes you back into manual testing

Tools will replace testers?

- “we can reduce the number of testers once we have the tool”
 - what are your testers like?
 - mindless morons, or
 - intelligent investigators?
 - need more skills, not fewer
 - automation can free testers to do more test design, exploratory testing
 - and find more bugs
 - tools don’t replace testers, they support them

Contents

- Test automation blunders
 - Testing-Tools-Test
 - Who needs GPS?
 - Silver Bullet
 - The Wrong Thing
 - How hard can it be?
 -  – Stable Application Myth
 - Project / Not Project
 - Inside the Box
 - Isolationism
- Conclusion

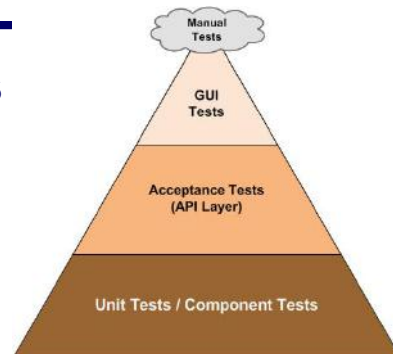
Twitter: @DorothyGraham

Stable application myth

- can't start automating until the application (or the GUI) is stable
 - throw-back to testing attitudes 30 years ago?
 - testing comes at the end?
 - testing is (only) execution?
- testing is more than testing [execution]
- automation is more than execution
- design your automation early, ready to run when anything is ready to test

When to write automated tests

- test automation pyramid
 - Mike Cohn, Lisa Crispin (Ch 1)
 - more unit/component tests
 - fewer GUI tests
- instability an opportunity rather than a problem
 - build flexibility & robustness in your automation against common types of changes
- which aspects are stable? (GUI often late)
 - execute tests for stable parts when ready
 - build (GUI) tests now but with detail abstracted out



Contents

- Test automation blunders
 - Testing-Tools-Test
 - Who needs GPS?
 - Silver Bullet
 - The Wrong Thing
 - How hard can it be?
 - Stable Application Myth
 - – Project / Not Project
 - Inside the Box
 - Isolationism
- Conclusion

Twitter: @DorothyGraham

Project / not project

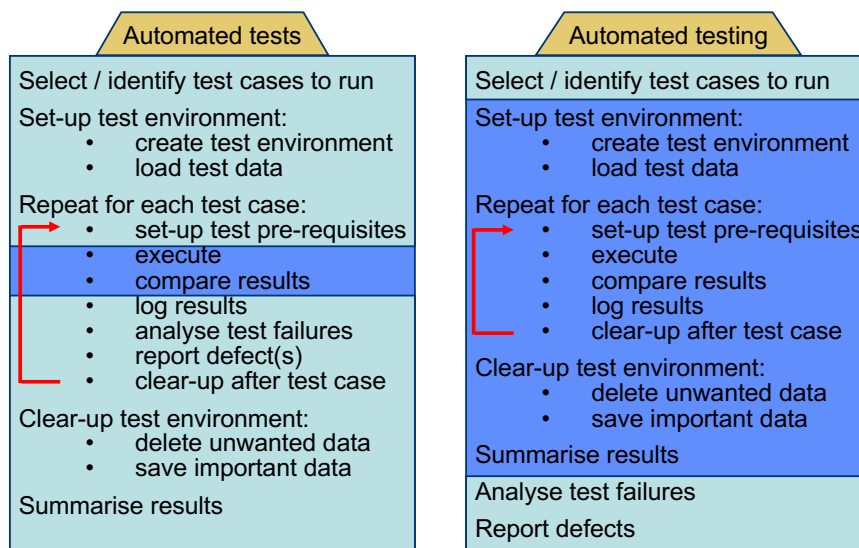
- not thinking of automation as a project
 - doesn't need funding, resourcing
 - just do it in your "spare time"
- thinking that automation is (just) a project
 - when will automation be finished? (wrong question!)
 - needs on-going continuous improvement
 - refactoring at regular intervals
- project to start automation, then continued support

Contents

- Test automation blunders
 - Testing-Tools-Test
 - Who needs GPS?
 - Silver Bullet
 - The Wrong Thing
 - How hard can it be?
 - Stable Application Myth
 - Project / Not Project
 - – Inside the Box
 - Isolationism
- Conclusion

Twitter: @DorothyGraham

Automated tests/automated testing



Contents

- Test automation blunders
 - Testing-Tools-Test
 - Who needs GPS?
 - Silver Bullet
 - The Wrong Thing
 - How hard can it be?
 - Stable Application Myth
 - Project / Not Project
 - Inside the Box
 - Isolationism
- Conclusion



Twitter: @DorothyGraham

Isolationism: isolated from

- change: encasing your first efforts in stone
 - automated tests benefit from review and refactoring
- realism: optimism as a strategy for automation
 - automated tests need to be tested and have bugs
 - realistic expectations
- managers: not making automation benefits visible
 - to the people who matter, in a way that communicates
- developers: not collaborating
 - design for automated testability, help developers
- the wider world: not seeking existing knowledge
 - books, wiki*, articles, blogs, discussions, events

Conclusion: see how these are wrong

- Summary of test automation blunders
 - Testing-Tools- **DON'T** –Test they just run stuff
 - Who needs GPS? automation needs direction
 - **NO** Silver Bullet needs time and effort
 - The Wrong Thing no testware architecture, just manual tests
 - How hard can it be? needs different skills
 - Stable Application Myth opportunity for flexibility
 - Project at times / Not just a Project ongoing, refactor
 - Inside the Box automation is more than automation
 - Isolationism collaborate, experiment, read, learn

DG

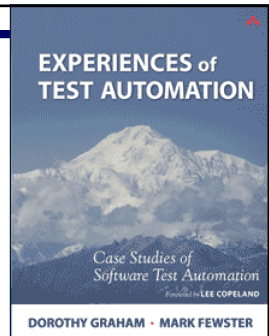
Twitter: @DorothyGraham

31

Thank you!

- www.DorothyGraham.co.uk
 - articles and papers
- info@DorothyGraham.co.uk
- twitter: @DorothyGraham
- Book: Experiences of Test Automation
 - 35% discount on ebook (informit.com/swtesting)
 - with code SWTESTING
- TestAutomationPatterns.org
 - free wiki of automation advice with Seretta Gamba
 - new book: A Journey through Test Automation Patterns
 - available on amazon from November

DG



A Journey through Test Automation Patterns

One team's adventures with the Test Automation Patterns wiki



Seretta Gamba and Dorothy Graham