IMPROVE
Quality Services

EUROPE'S GREATEST AGILE SOFTWARE TESTING EVENT!
November 10–13, 2014, Potsdam, Germany

Workshop

Practical agile test strategy using heuristics

Huib Schoots
@huibschoots
huib.schoots@improveqs.nl

AGILE TESTING DAYS

Acknowledgements

Thanks to:

- Ruud Cox for the many discussions on this topic
- Jean-Paul Varwijk for helping me making it awesome
- Fiona Charles and Rikard Edgren for inspiration
- Obviously James Bach and Michael Bolton for sharing their knowledge about Rapid Software Testing and the Heuristic Test Strategy Model

Many slides are taken from Rapid Software Testing and are used with permission. Rapid Software Testing was developed by James Bach and Michael Bolton. Also see: http://www.satisfice.com/info_rst.shtml

Agile testing

What is agile testing?

I think agile testing is just testing… in an agile context!

Some context factors to deal with:

- Short sprints
- Iterative and incremental
- Team work
- Less certainty: change is common
- Continuous critical thinking

Source: http://www.huibschoots.nl/wordpress/?p=1072

What is test strategy?

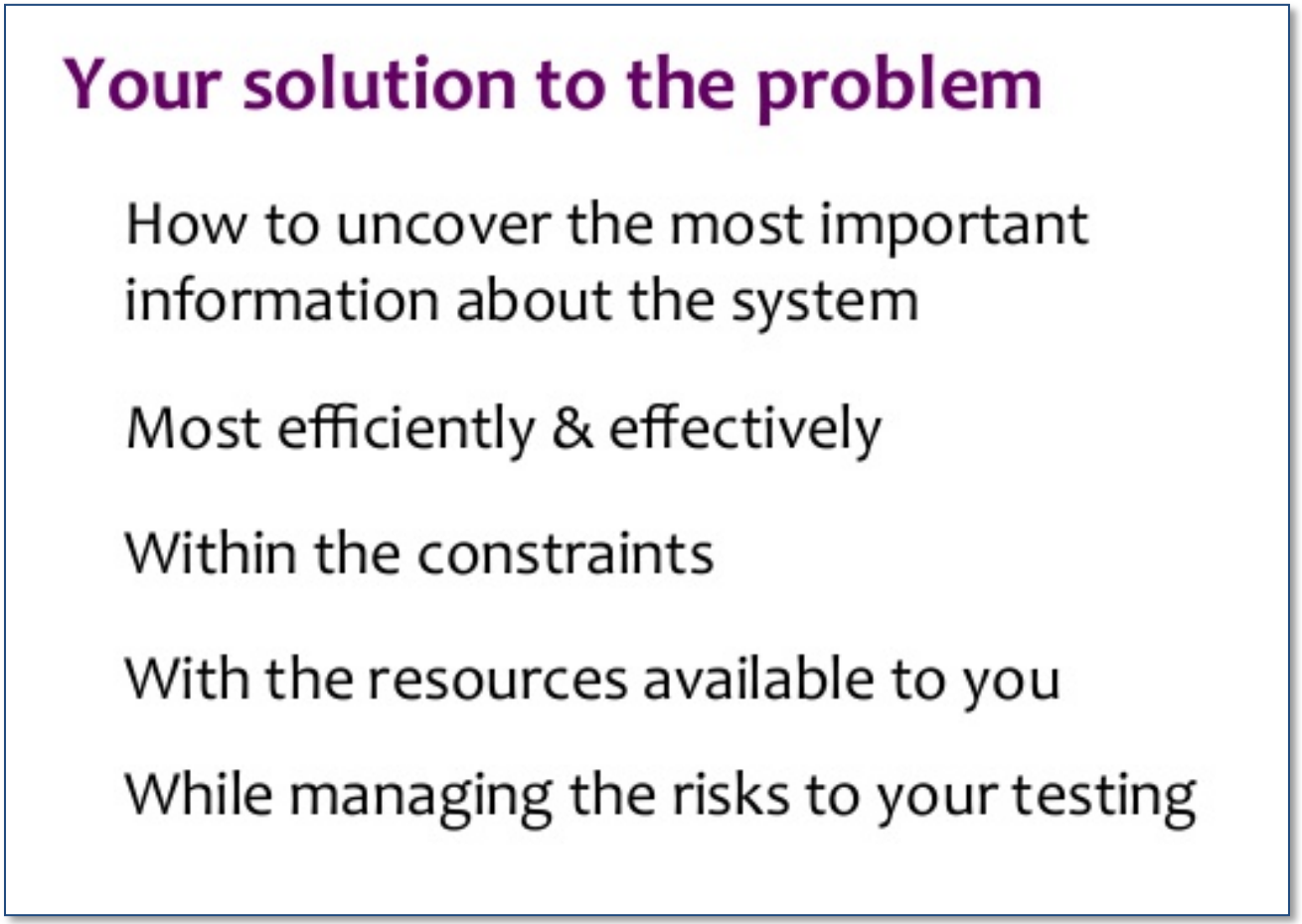
1. What is test strategy to you?

2. Why do you make your test strategy?

3. What does your test strategy look like?

Test Strategy (according to ISTQB Glossary definition)
A high-level description of the test levels to be performed and the testing within those levels for an organization or programme (one or more projects).

Test strategy to me (and Fiona too ☺)



Source: EuroStar Webinar Thinking Strategically About Testing by Fiona Charles

Creating a test strategy: problem solving!

1. Define the testing problems (or test missions)
2. Define solutions to these problems
3. Communicate / capture / execute

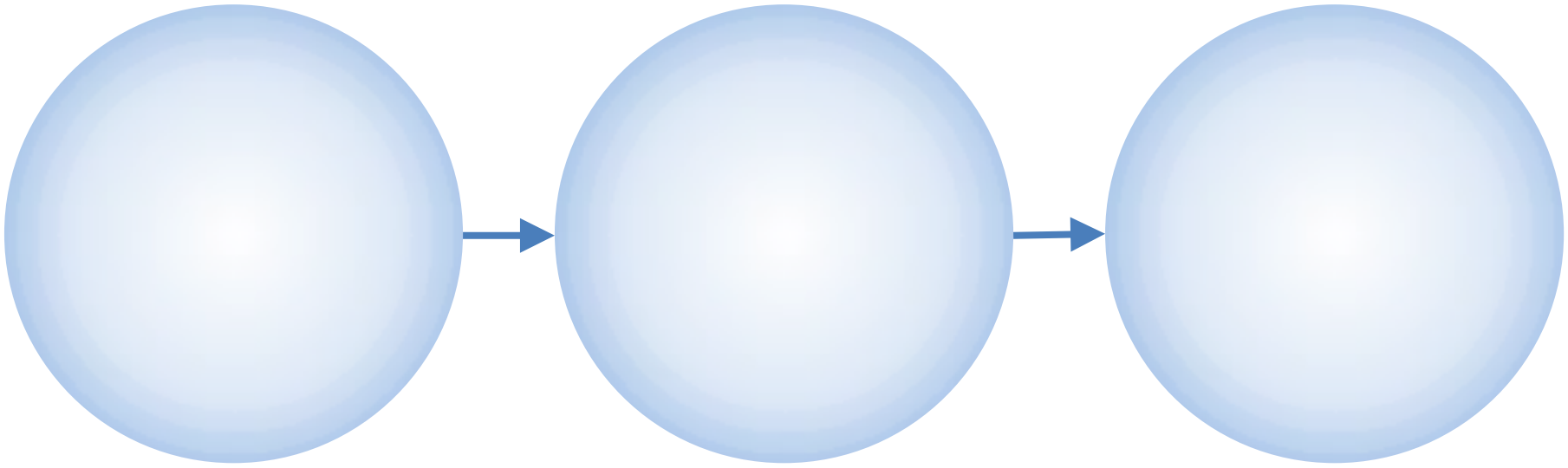
A test strategy is a solution to a complex problem: How do we meet the information needs of the stakeholders in the most efficient way possible?

Test strategy

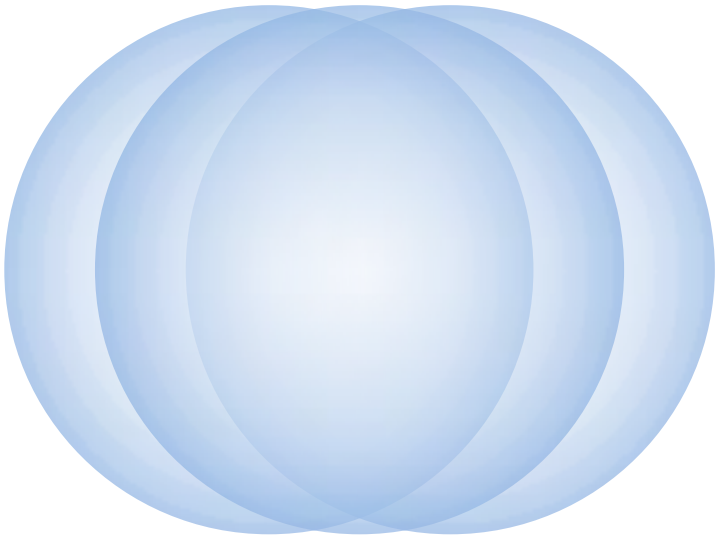
- Strategy: The set of ideas that guide your **test design**
- Logistics: The set of ideas that guide your **application of resources** to fulfilling the test strategy
- Plan: The set of ideas that guide your **test project**

plan = strategy + logistics

Strategy – Execution – Report

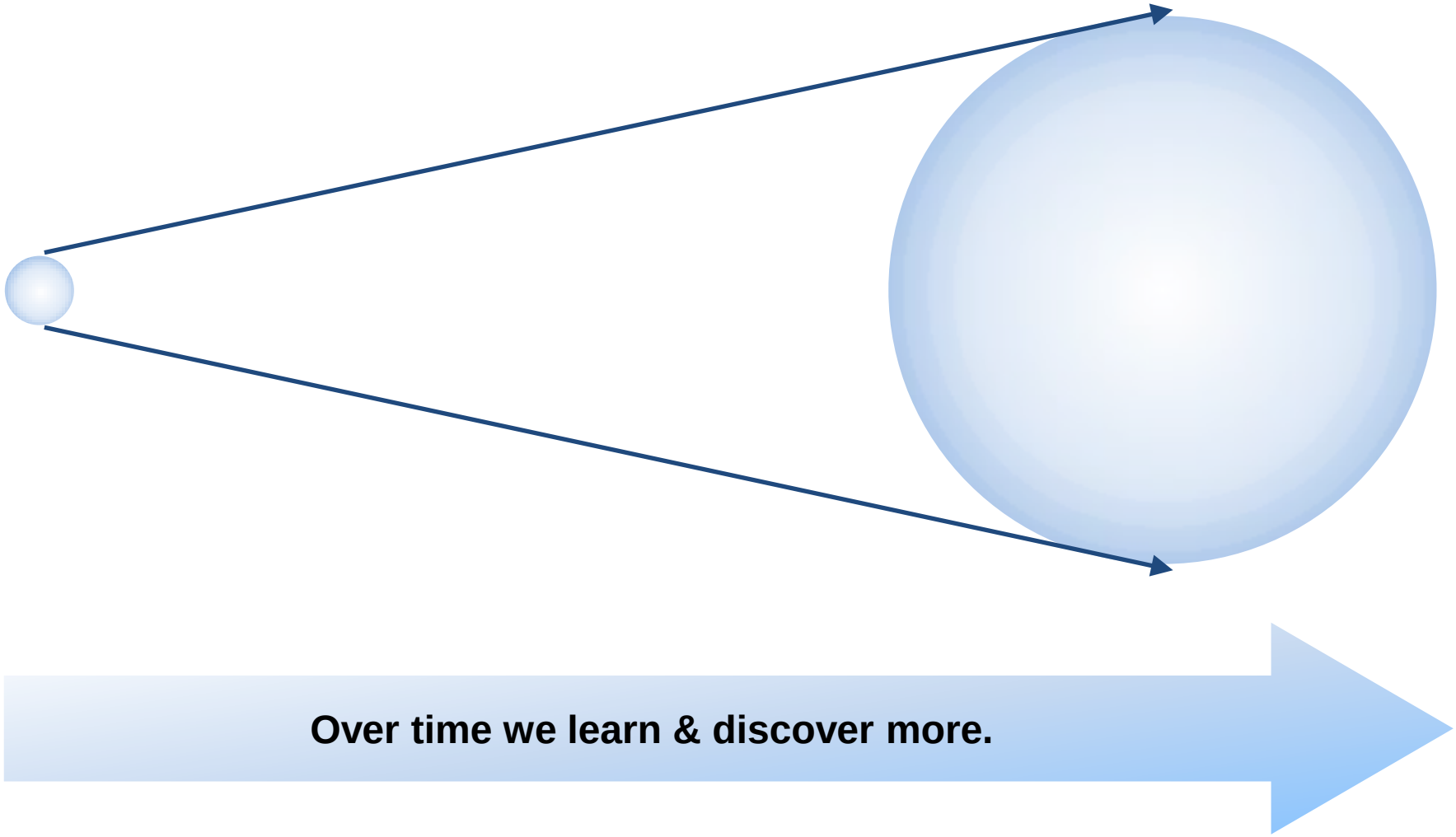


Strategy – Execution – Report



Testing

Evolving test strategy



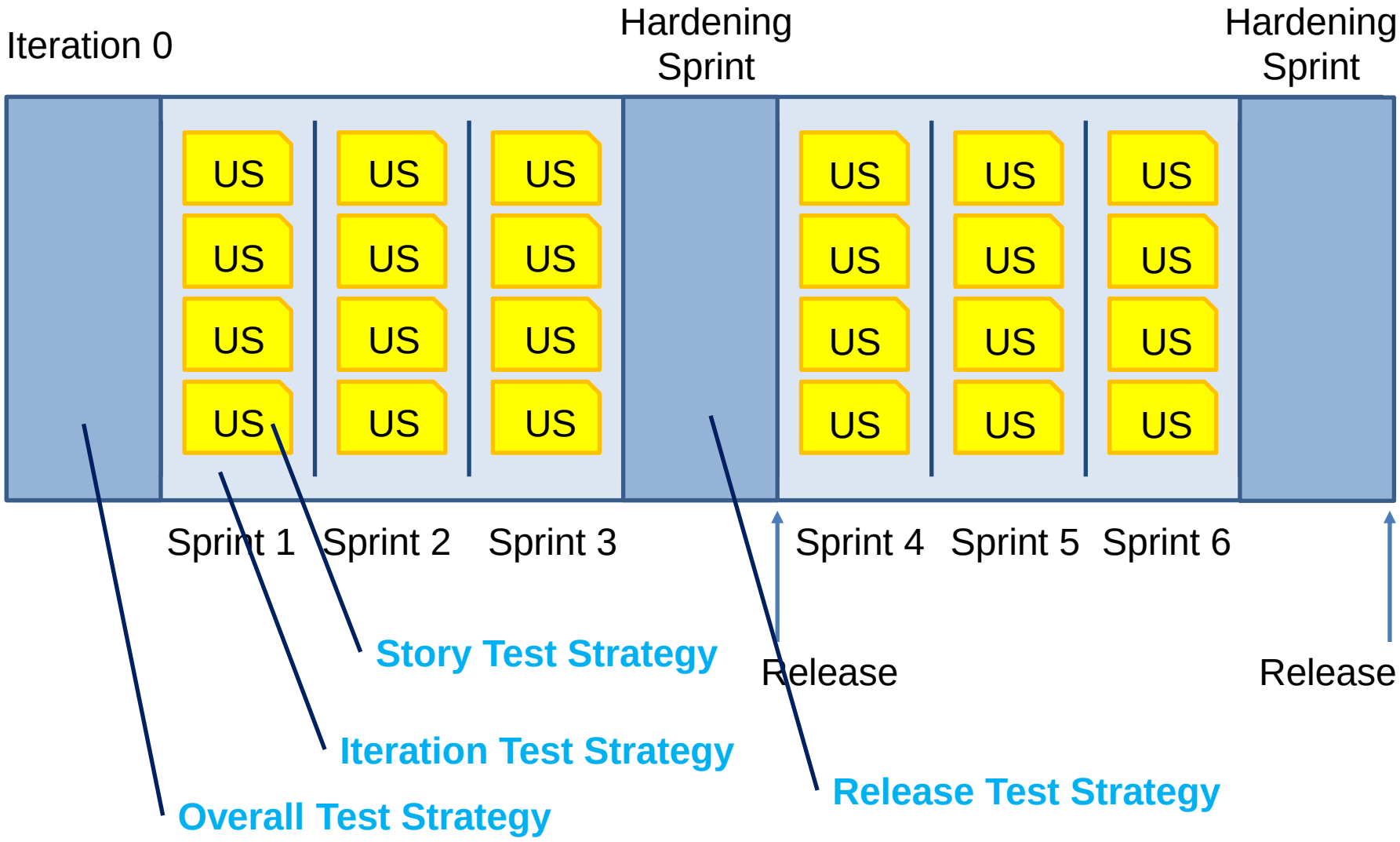
Things to consider…

Aspects of test strategies

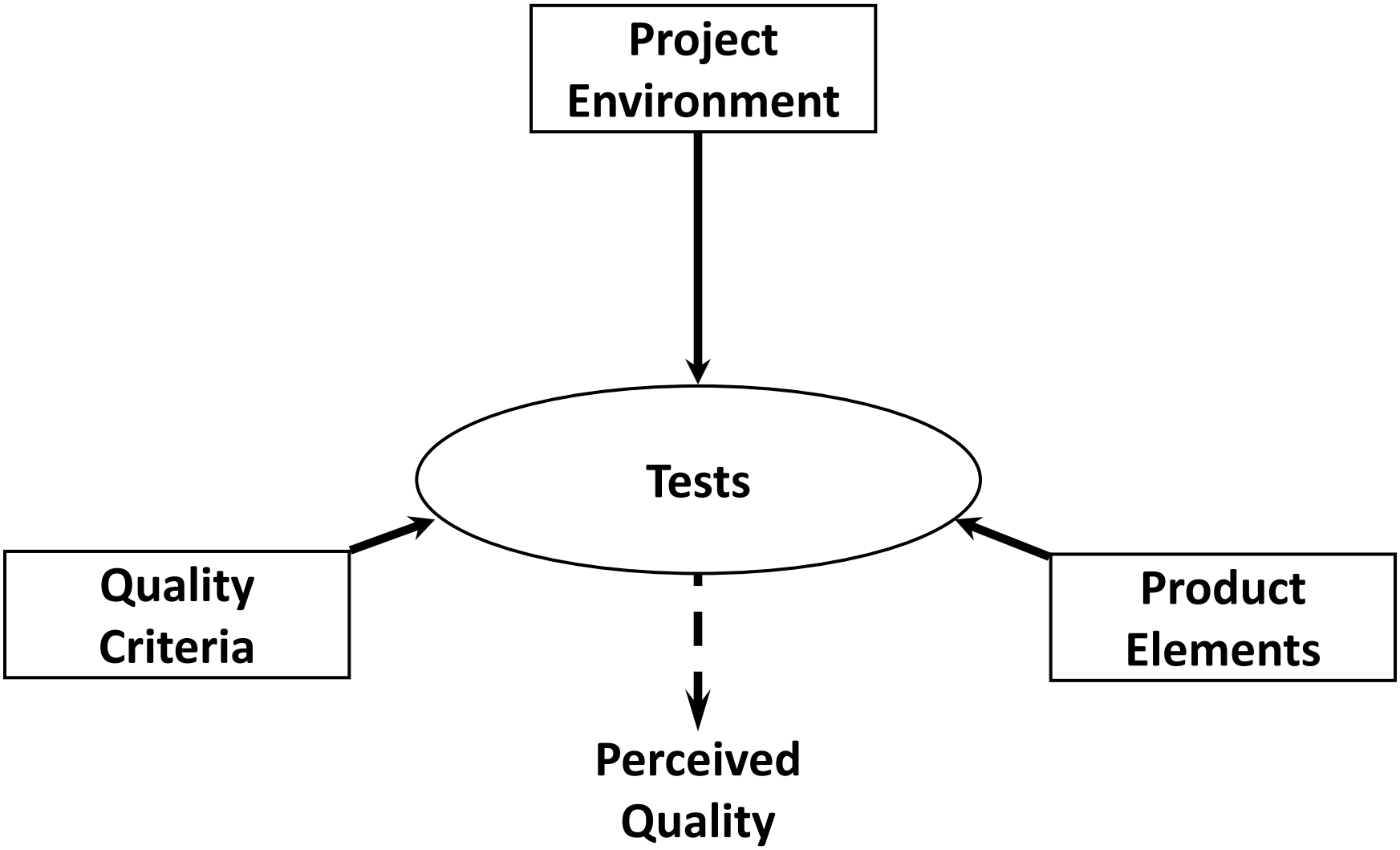
- What is important?
- Goals
- Test techniques
- Test ideas (worth mentioning)
- Information sources
- Oracles
- Models
- Quality objectives
- How testers think
- Trade-offs
- Risks
- Marketing

Source: Nordic Testing Days Workshop Test Strategy the next level by Rikard Edgren

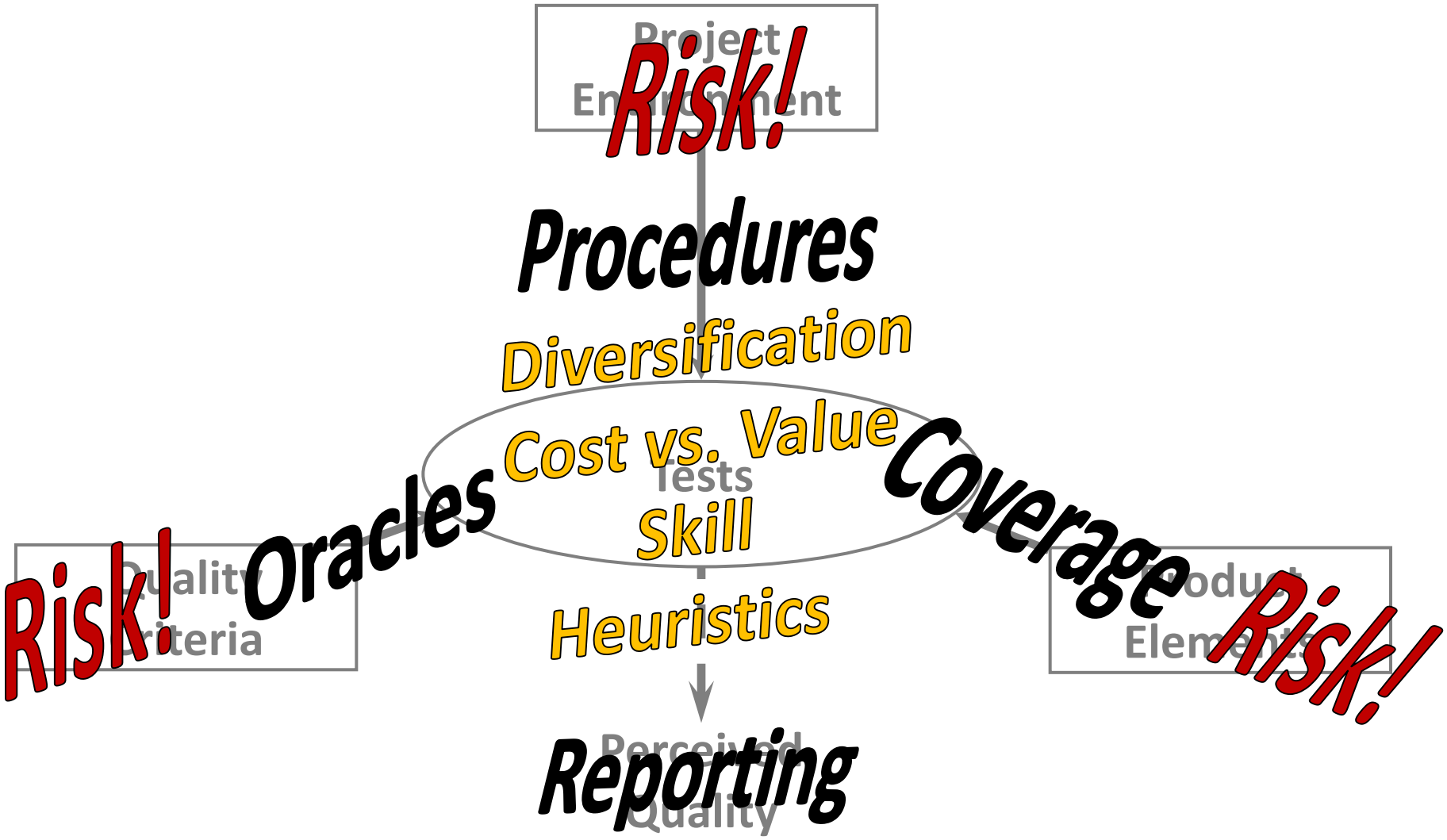
Test strategy in agile…



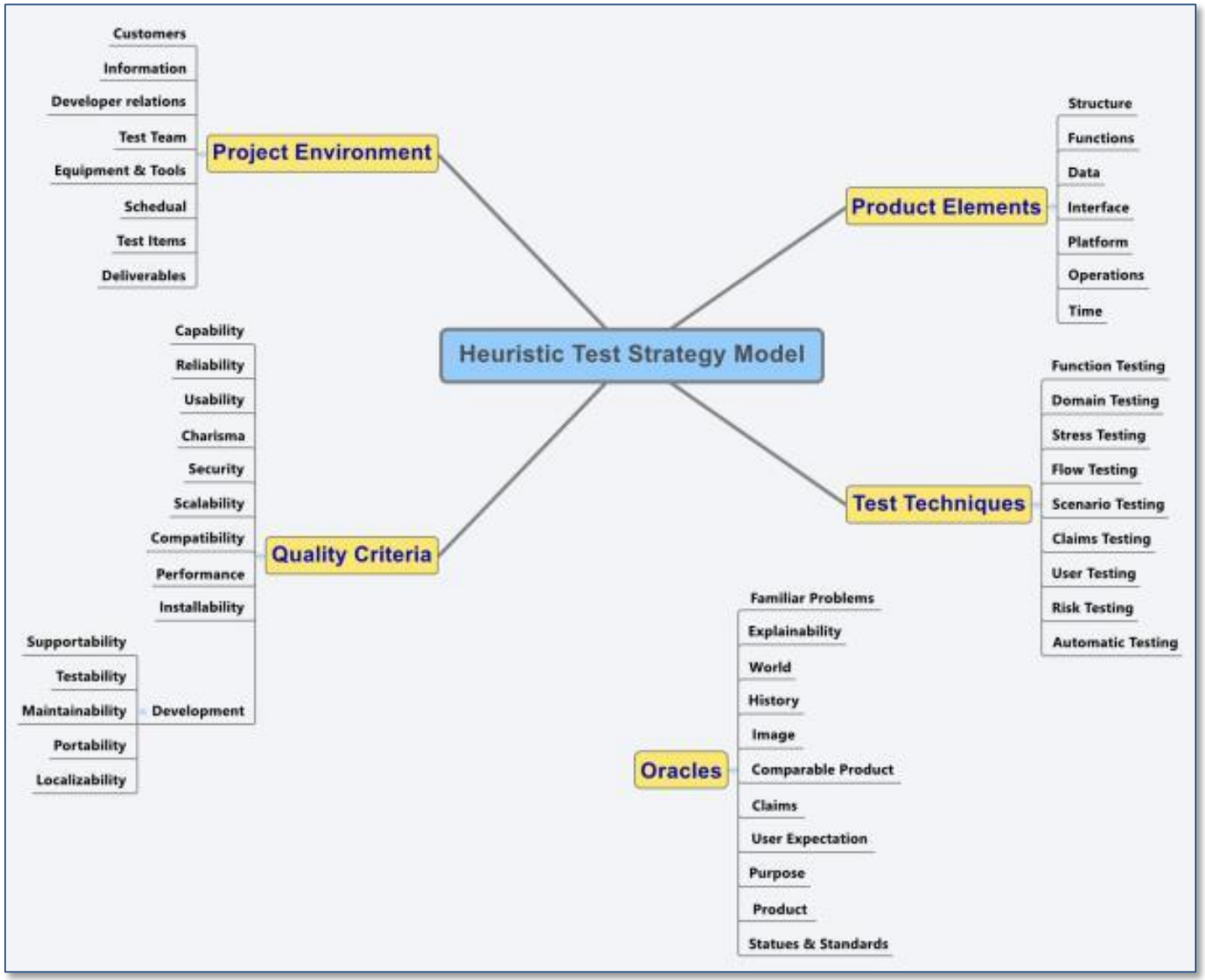
Heuristic Test Strategy Model



Heuristic Test Strategy Model



Heuristic Test Strategy Model



Project Environment

Ways to understand our context

MIDTESTD

- Mission
  - *The set of things we must do in order to satisfy our clients.*
- Information
  - *Information about the product or project that is needed for testing.*
- Developer relations
  - *How you get along with the programmers.*
- Test team
  - *Anyone who will perform or support testing.*
- Equipment & tools
  - *Hardware, software, or documents required to administer testing.*
- Schedule
  - *The sequence, duration, and synchronization of project events.*
- Test Items
  - *The product to be tested.*
- Deliverables
  - *The observable products of the test project.*

General Test Techniques

"Ways to test…"?

FDSFSCURA

- Function testing: test what it can do
- Domain testing: divide and conquer the data
- Stress testing: overwhelm or starve the system
- Flow testing: do one thing after another after another
- Scenario testing: test to a compelling story
- Claims testing: test what people have written or said
- User testing: involve (or systematically simulate) the users
- Risk testing: think of a problem, then test for it
- Automatic checking: check a million different facts

General Test Techniques

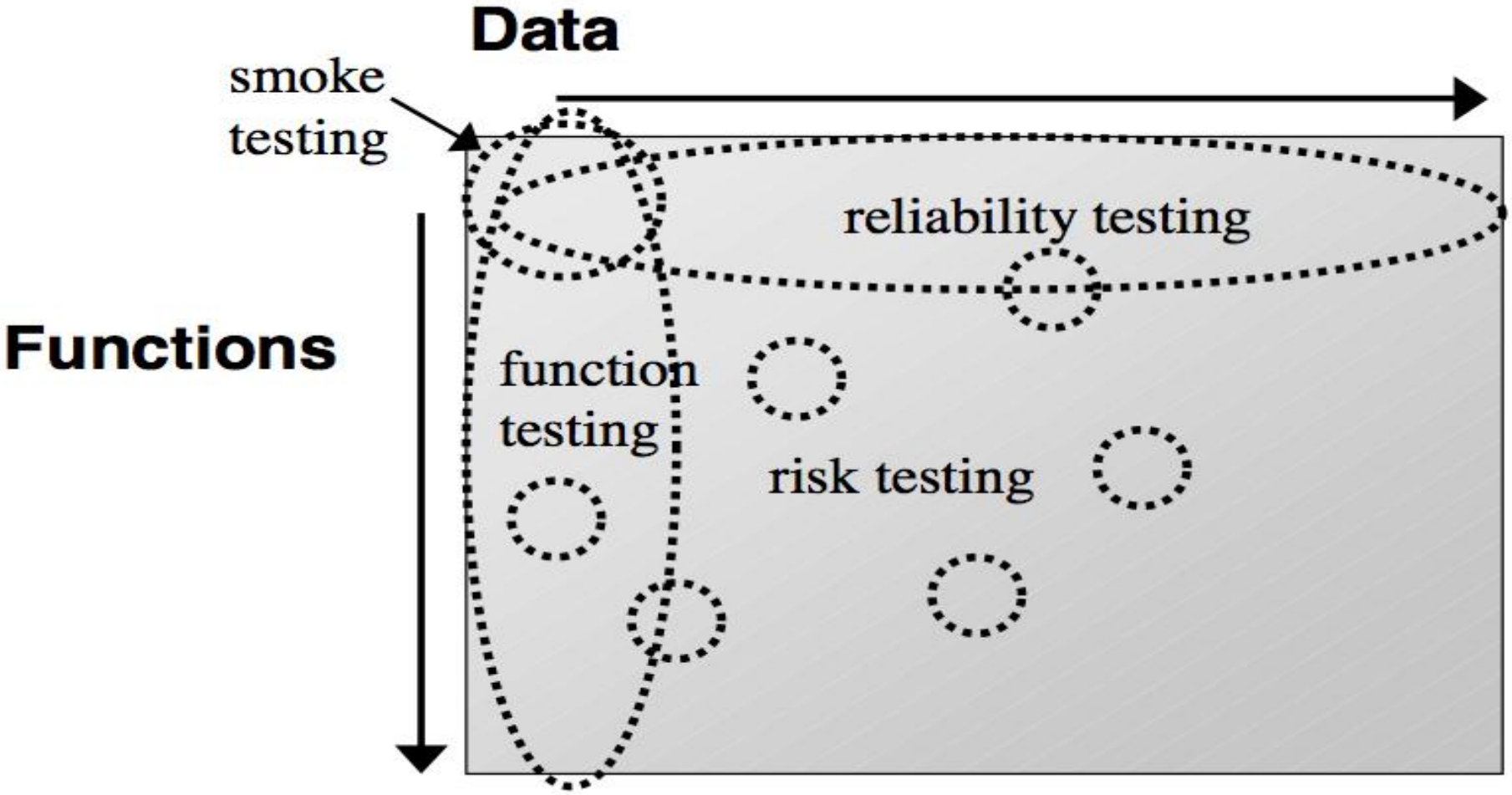
General testing techniques

- **Function testing** – test that each function does with it's supposed to
- **Risk-based testing** – try to provoke important risks (deal with probablility afterwards)
- **Specification-based testing** – use product claims (not necessarily a specification) and see if they hold.
- **Scenario testing** – test longer sequences, with complexity for sequence order, users, data and/or environment.
- **Model-based testning** – test from states, architecture, flows or custom models.
- **Quality objective-based testing** – Each quality characteristic can be used as a testing method, e.g. performance, security, usability, compatibility (plus sub-categories.)
- **High volume testing** – Run an awful amount of tests to evaluate stability, use of "all" data, see patterns etc.
- **Domain testing** – Choose data from equivalence groups, boundary values, or best representatives.
- **User testing** – Let (simulated) users perform tasks.
- **Testing without flourishes**– You know what to test, and do it.

- Manual/Automated/Exploratory/Scripted are orthogonal.

Source: Nordic Testing Days Workshop Test Strategy the next level by Rikard Edgren

Using Test Techniques



Source: http://www.satisfice.com/presentations/etta.pdf

Quality Criteria

Identifying value and threats to it…

CRUCSS CPID

- Capabililty
- Reliability
- Usability
- Charisma
- Security
- Scalability
- Compatibility
- Performance
- Installability
- Development

Many test approaches focus on Capability (functionality) and underemphasize the other criteria

What IS Coverage?

_____ coverage is "how much testing we've done with respect to some model of _____"

It's the extent to which we have traveled over *some map* of the product.

But what does it mean to "map" a product?

Talking about coverage means talking about

MODELS

There are as many kinds of test coverage as there are ways to model the system.

- Structure
- Function
- Data
- Interfaces
- Platform
- Operations
- Time
- Technical Risk
- Business Risk
- Features / stories

…and each kind of coverage can be obtained *intentionally*, *incidentally*, or *accidentally*.

See “Got You Covered”, “Cover or Discover”, and “A Map By Any Other Name”

http://www.developsense.com/publications.html

Product elements

SFDIPOT modeling

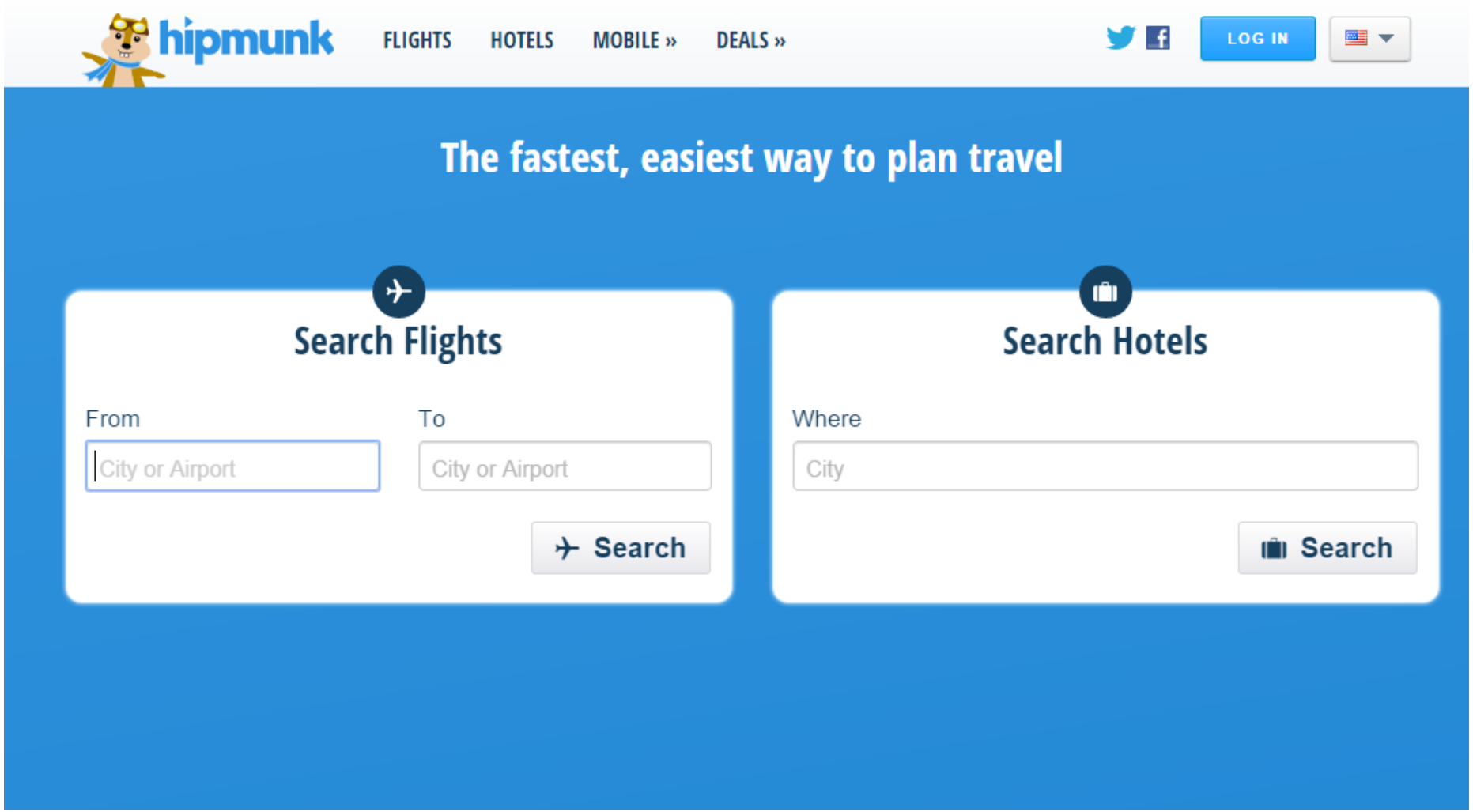
- A great framework for getting structure is to use SFDIPOT from James Bach's Heuristic Test Strategy Model.

- **Structure** – what the product is
- **Functions** – what the product does
- **Data** – what the product operates on
- **Interfaces** – how you interact with the product
- **Platform** – the environment the product depends on
- **Operations** – what the users want to accomplish
- **Time** – relations between the product and time

- These guidewords structure your thinking, and give better breadth.
- But you still have to do all the work yourself...

Source: Nordic Testing Days Workshop Test Strategy the next level by Rikard Edgren

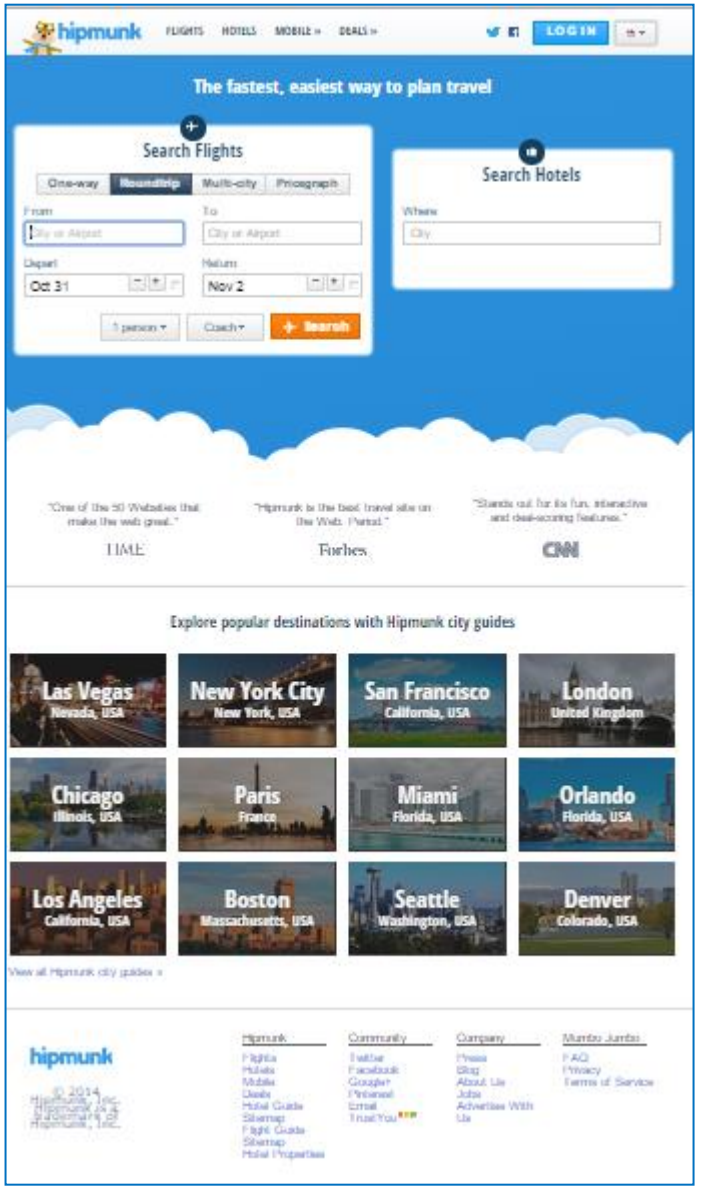
Let's look at Hipmunk…



Let’s look at Hipmunk…

- Hipmunk is a remarkable new travel search site that aims to take the agony out of travel planning. The goal is to help you book travel faster and more efficiently.
- Hipmunk shows all relevant flight or hotel results on a single page, in a visual "timeline" that makes it easy to understand the tradeoffs between options. Hipmunk was designed to help people who are overwhelmed with pages of irrelevant search results.
- How can we look at Hipmunk and find problems in it?

- Learn the product. Using a mind map, begin creating a product coverage outline and a risk list. A map of the product's elements will help to guide future sessions of testing.
- Identify problems that might threaten the value of the product.



http://www.hipmunk.com/

http://www.satisfice.com/tools/htsm.pdf

Thirty-Four Test Strategy Heuristics

midtestdsfdipotcrucsscpidfdsfscura

Mission
Information
Developer relations
Team
Equipment & tools
Schedule
Test Items
Deliverables

Project Environment

Structures
Functions
Data
Interfaces
Platforms
Operations
Time

Product Elements

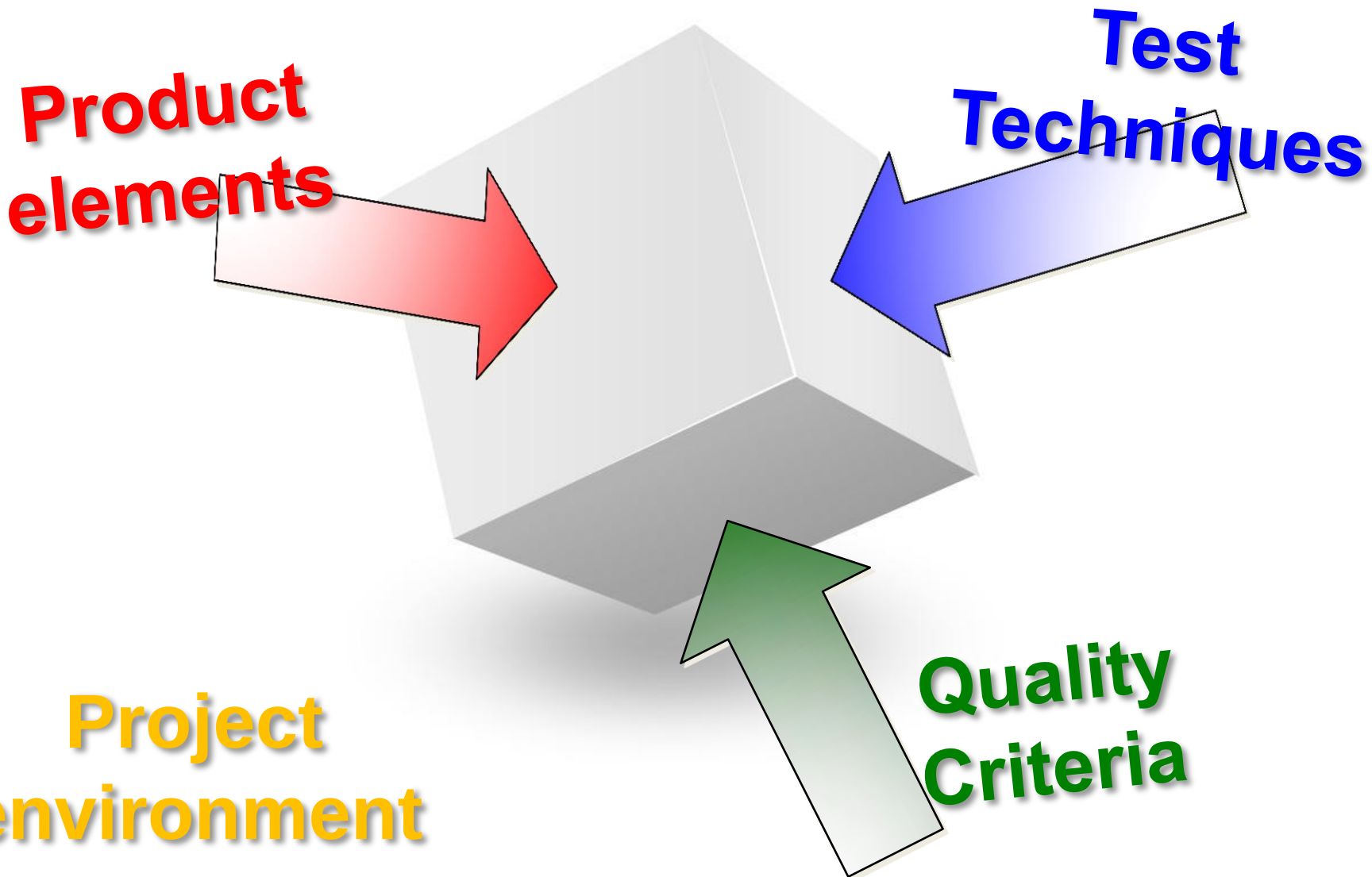
Capability
Reliability
Usability
Charisma
Security
Scalability
Compatibility
Performance
Installability
Development

Quality Criteria

Function testing
Domain testing
Stress testing
Flow testing
Scenario testing
Claims testing
User testing
Risk testing
Automatic testing

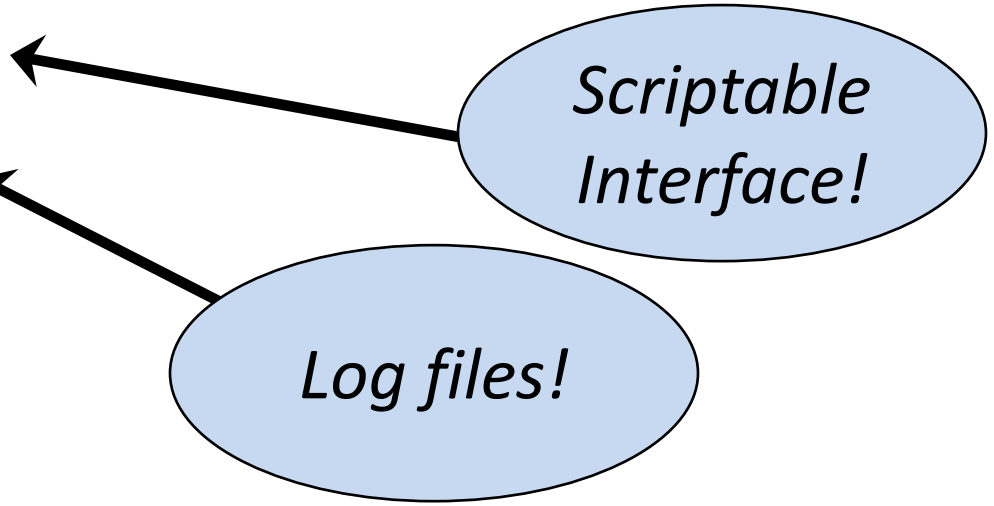
Test Techniques

Balanced test strategy



Sometimes it's really hard to cover…
Ask for testability!

- Controllability
- Observability
- Configurability
- Availability
- Simplicity
- Stability
- Information



Testing is far more rapid
when the product is more testable

Test strategy is …

Your unique test strategy

- Every situation requires a unique test strategy.
- You always have one, even though it isn't documented.

- A good test strategy is
  - specific – details rather than fluff
  - practical – possible to execute with "normal" turbulence
  - justified – reaches the testing missions
  - diverse – important systems needs to be tested in many different ways
  - resource efficient – uses available resources without (too much) waste
  - reviewable – possible to understand and review, so it focus on right things
  - anchored – in management, in testers
  - changeable – to be able to deal with the unevitable unknown
  - erroneous – if it isn't "incorrect", it is too vague, or took too long time to write

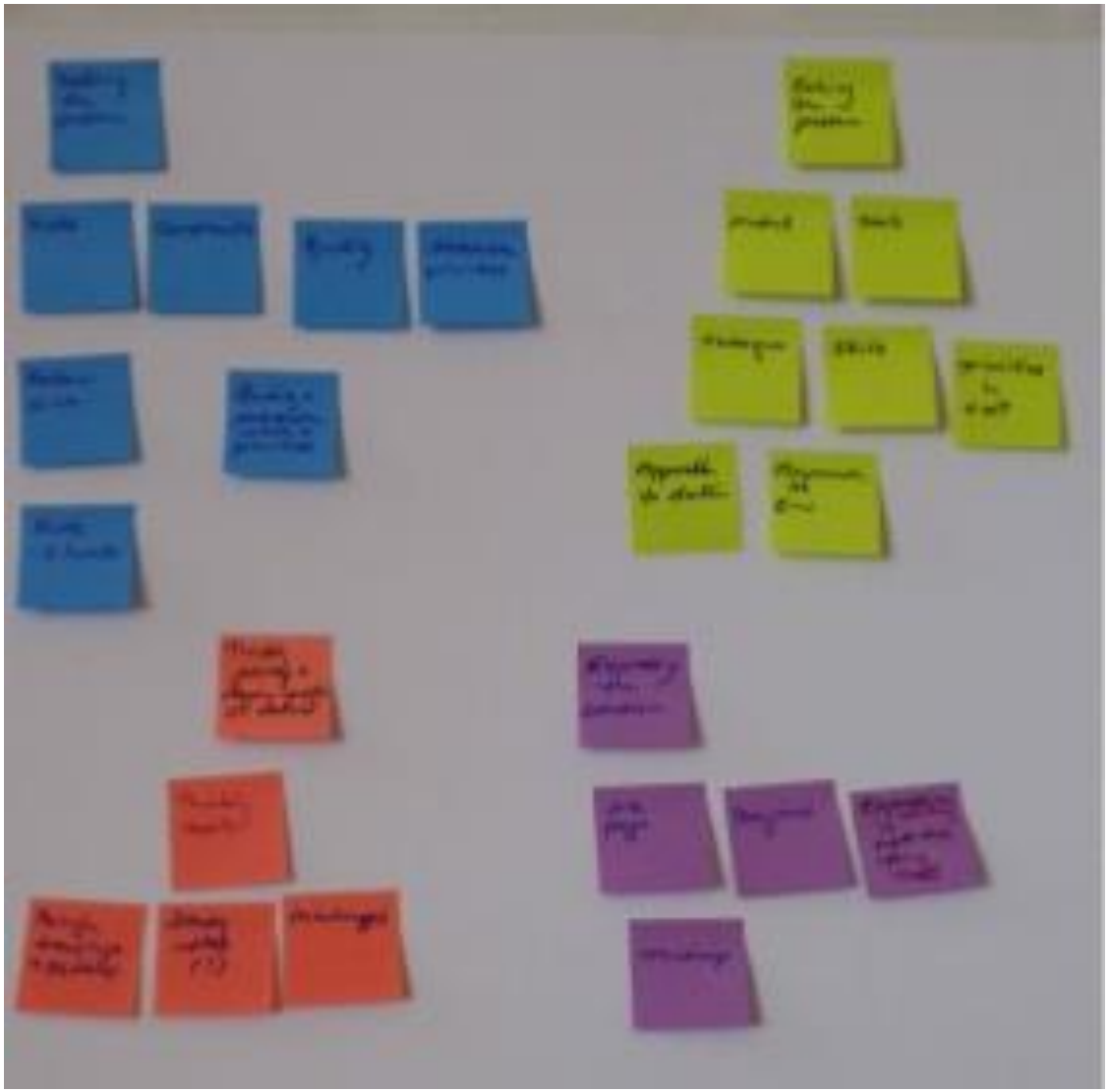
- It is better to test pretty well in many ways, than perfect in one or two. [#283, Lessons Learned in Software Testing]

Source: Nordic Testing Days Workshop Test Strategy the next level by Rikard Edgren

Some examples

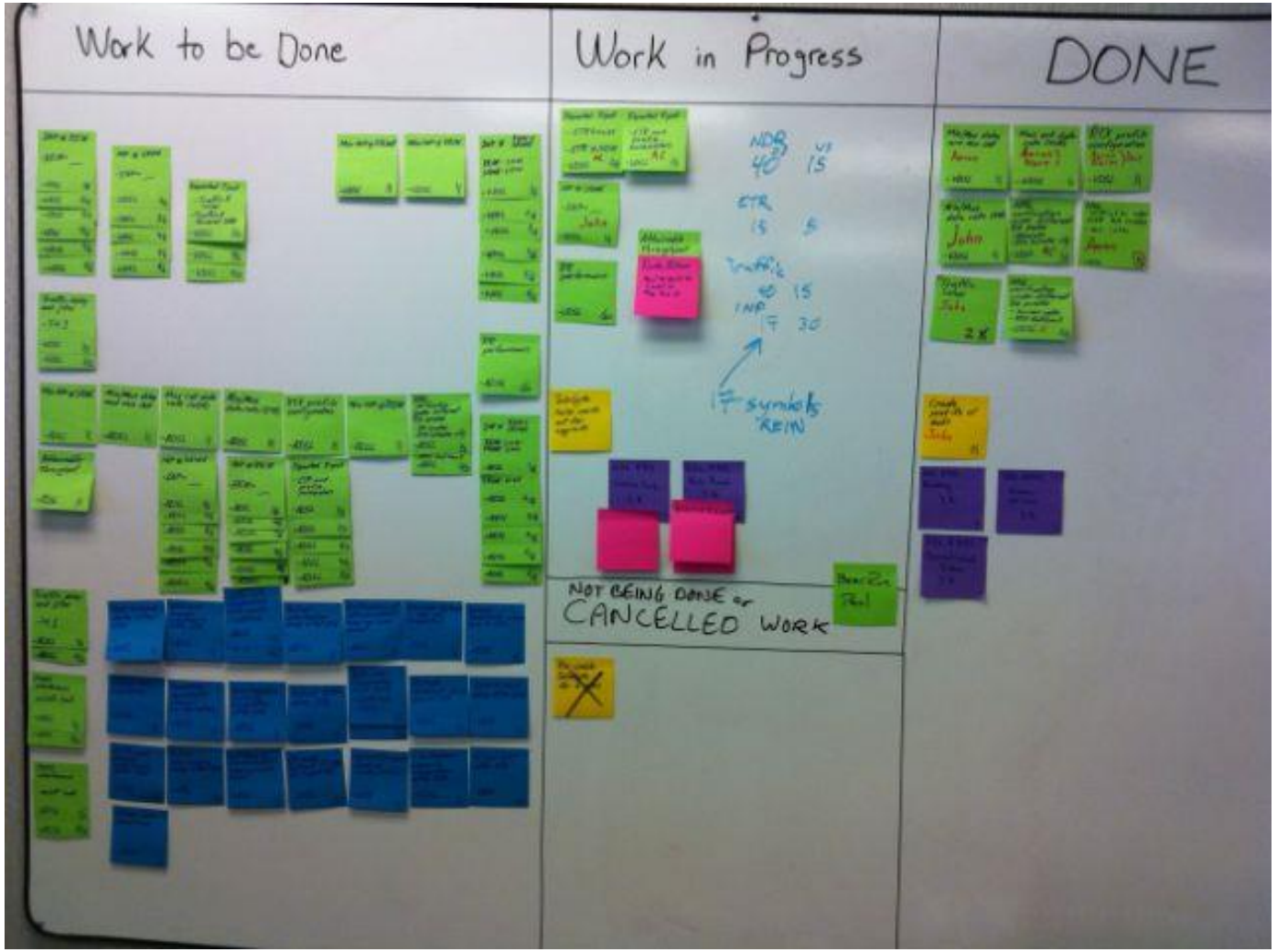
- OWL Quality Plan (p. 107, RST Appendices)
  - Risk and Task Correlation
  - Component Breakdown
- Test Plan (p. 115, RST Appendices)
  - Risk vs. Strategy
- Session Based Test Management
- Visual Test Strategy

Visualise your model



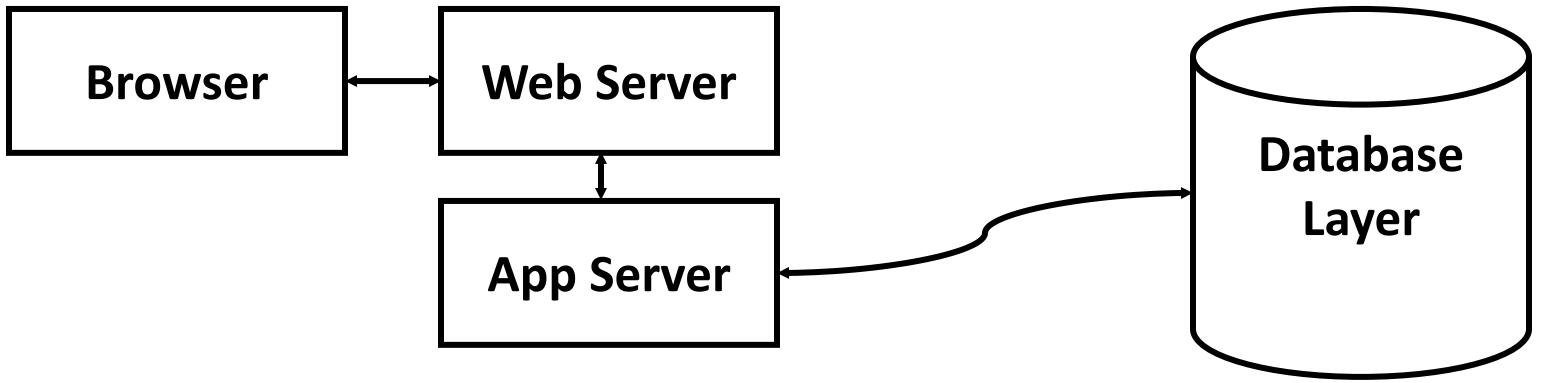
Source: EuroStar Webinar Thinking Strategically About Testing by Fiona Charles

Visualizing Testing & Progress

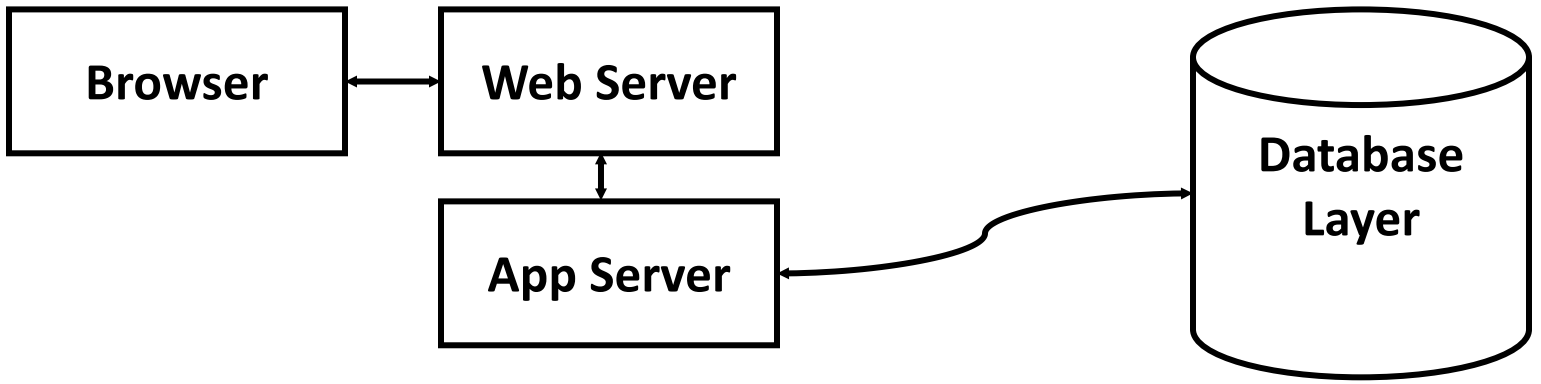


Visual Strategy: *Analysis*

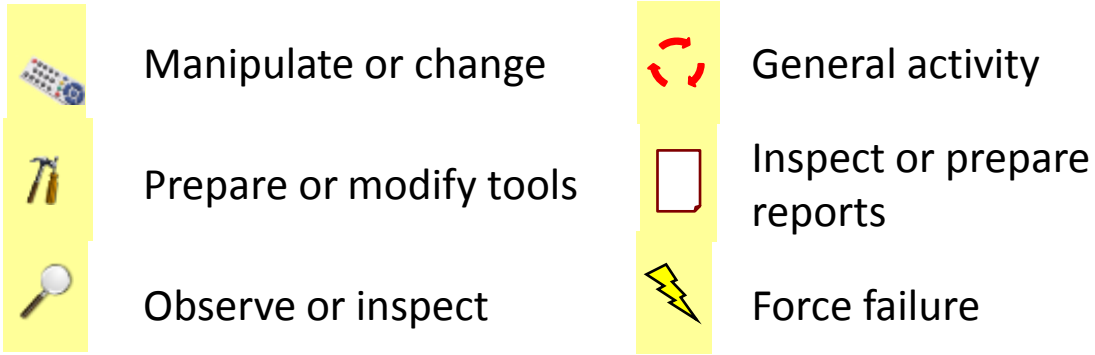
- [pointing at a box] *What if the function in this box fails?*
- *Can this function ever be invoked at the wrong time?*
- [pointing at any part of the diagram] *What error checking do you do here?*
- [pointing at an arrow] *What exactly does this arrow mean? What would happen if it was broken?*
- [pointing at a box] *What actually happens inside this box? What would happen if this box were updated or replaced?*
- [pointing at a box] *Are there other ways for things to get into or out of this box? Are there any missing lines?*



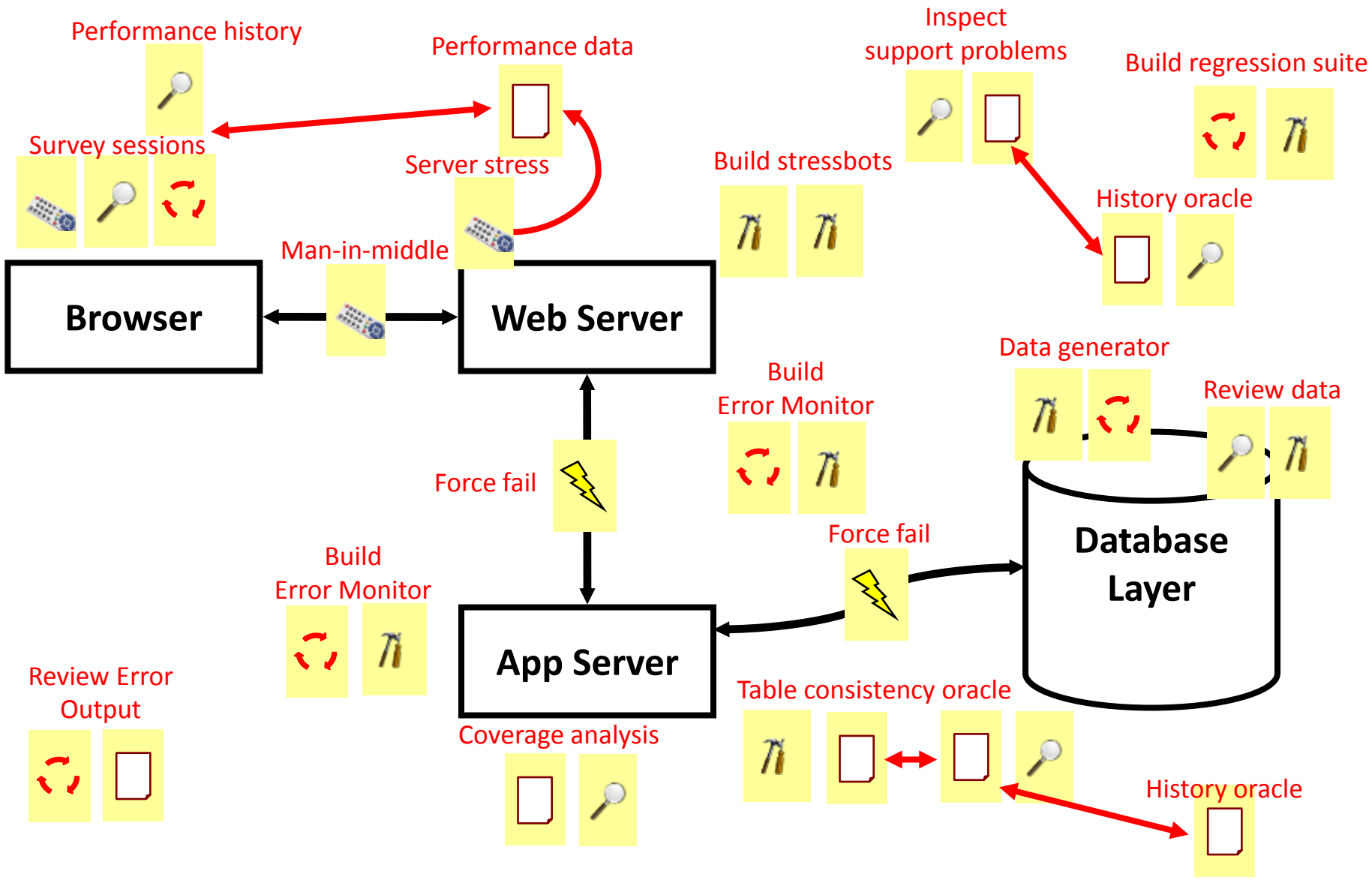
Visual Test Strategy: *Logistics*



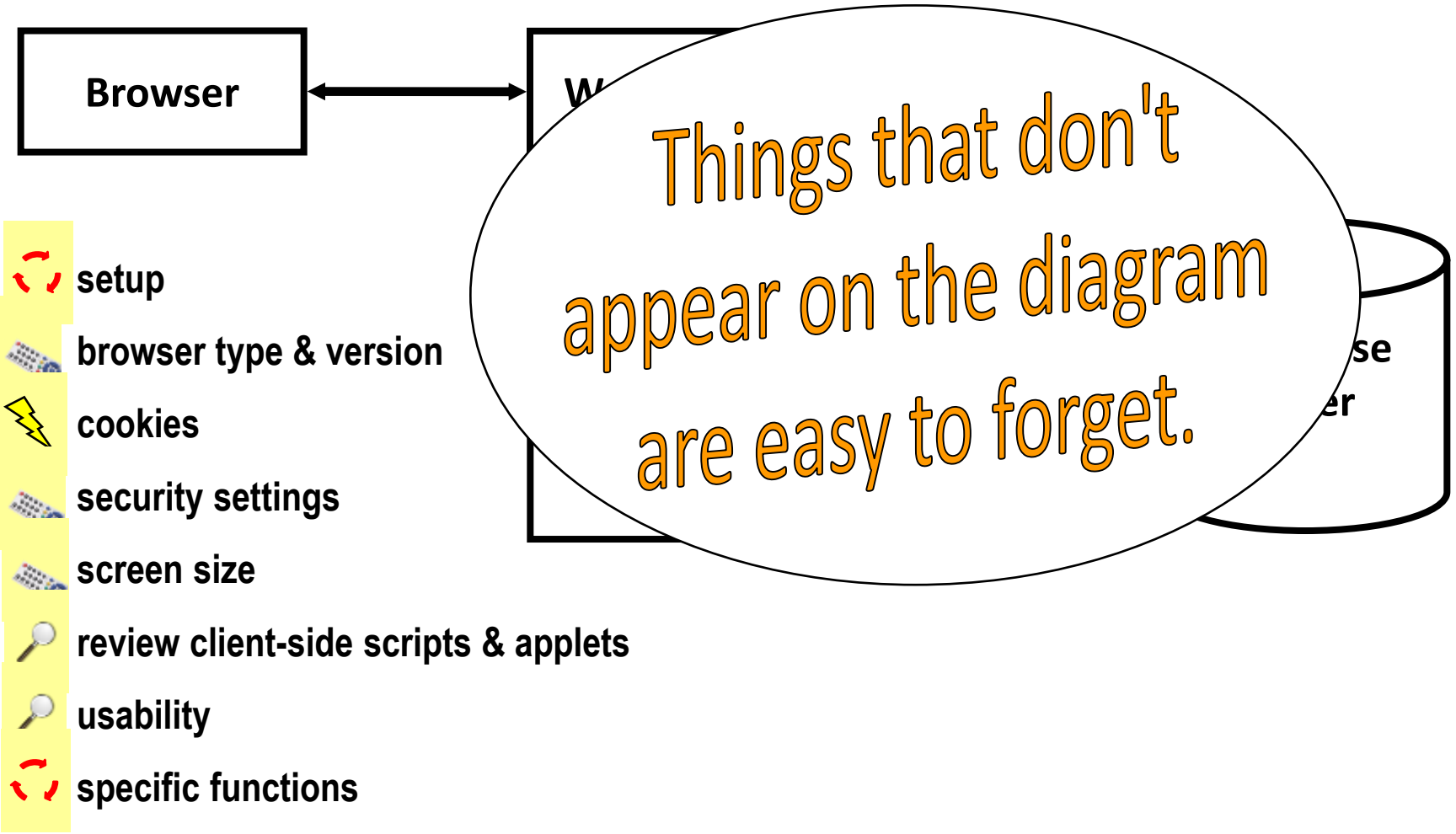
- Example: with a team of four testers, one session per morning/afternoon, five days a week…
- …model time-based activities and coverage with sticky notes



Visualizing Test Coverage: Annotation

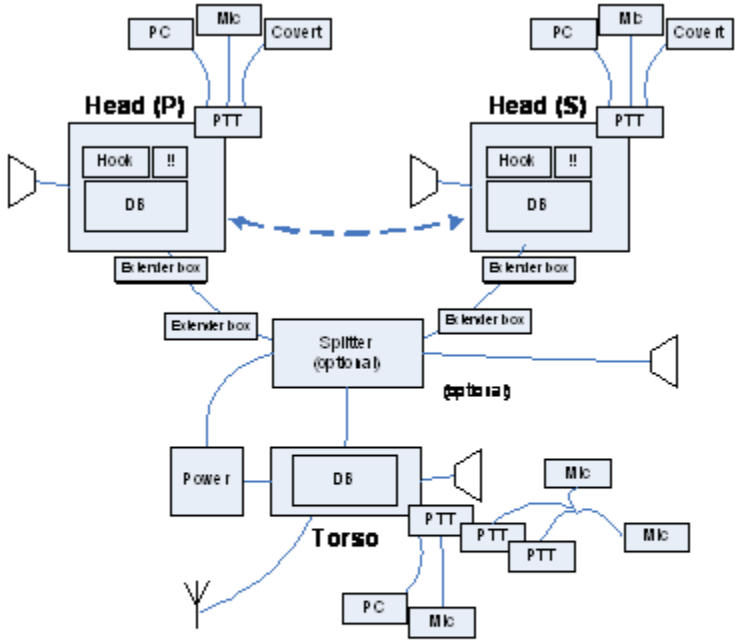


Beware Visual Bias!

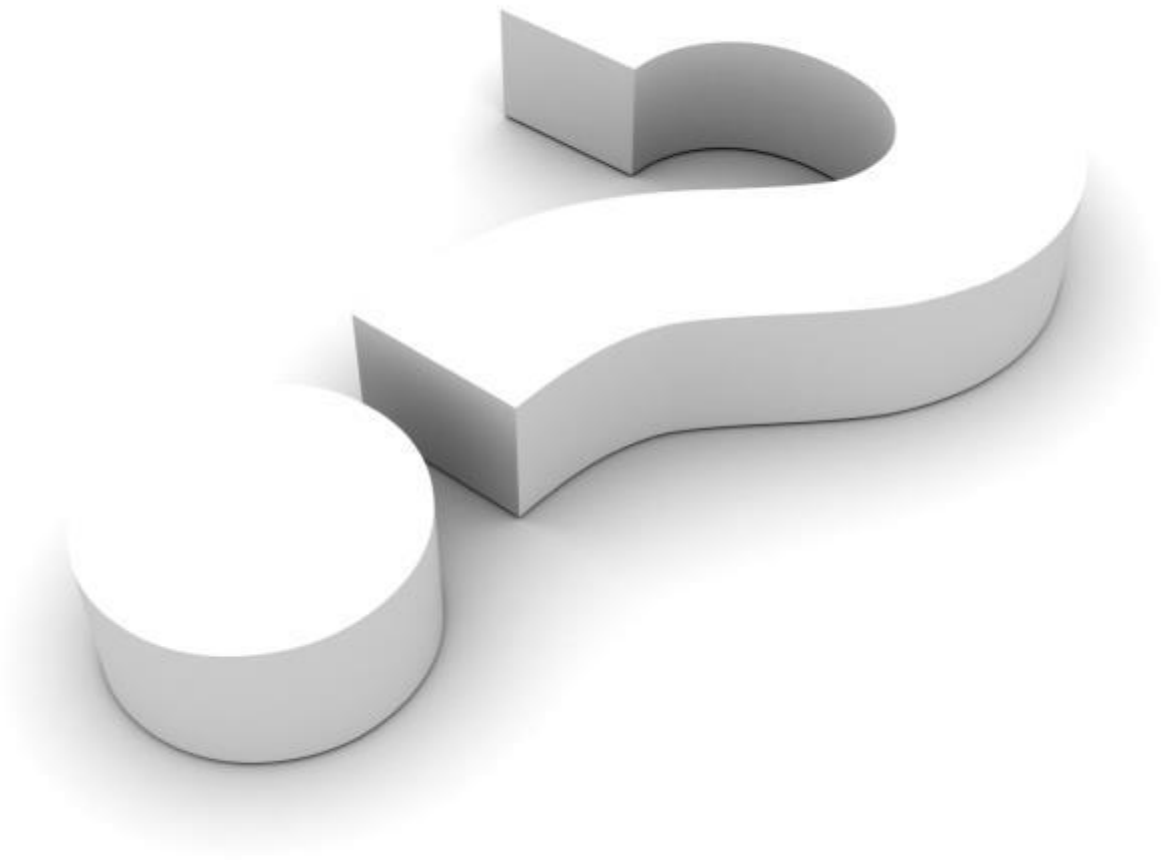


One way to cope with really complex diagrams

- Consider making a special diagram that includes only the things that are worth testing, then put the annotations as bullets on the bottom…



Questions, remarks, discusson, feedback?



Contacts

Huib Schoots

hsc@Improveqs.nl

@huibschoots

www.huibschoots.nl/blog

Laan van Diepenvoorde 1
5582 LA Waalre
The Netherlands
Tel: +31 40 2021803

References & more info

- Rapid Software Testing – James Bach & Michael Bolton
  http://www.satisfice.com/info_rst.shtml
- Heuristic Test Strategy Model – Designed by James Bach
  http://www.satisfice.com/tools/htsm.pdf
- Heuristic Test Planning – James Bach
  http://www.satisfice.com/tools/satisfice-cm.pdf
- Heuristic Risk-Based Testing – James Bach
  http://www.satisfice.com/articles/hrbt.pdf
- Basics Revisited: Test Strategy - Fiona Charles
  http://www.quality-intelligence.com/articles/BasicsRevisited-TestStrategy.pdf
- Webinar: Thinking Strategically About Testing - Fiona Charles
  http://testhuddle.com/resource/thinking-strategically-about-testing-with-fiona-charles/
- What is a good test strategy – Rikard Edgren
  http://thetesteye.com/blog/2013/09/what-is-a-good-test-strategy
- Software Quality Characteristics – Thetesteye.com
  http://thetesteye.com/blog/2011/11/software-quality-characteristics-1-1/
- Workshop Test Strategy the next level – Rikard Edgren
  http://nordictestingdays.eu/sites/default/files/NTD2014%20Presentations/TestStrategyNextLevel_FullDayTutorial.pdf
- Testability heuristics - James Bach
  http://www.satisfice.com/tools/testable.pdf