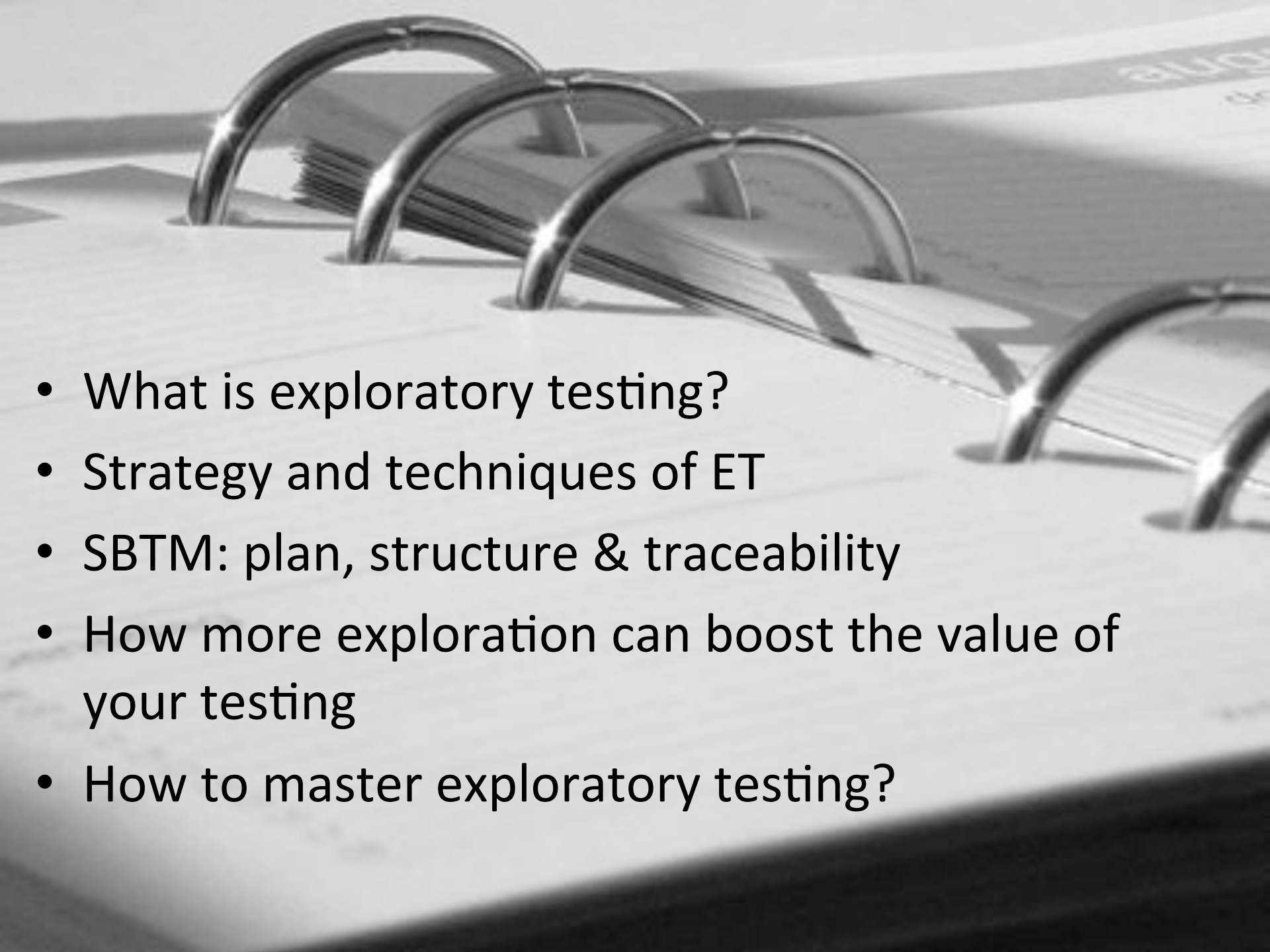
Huib Schoots (codecentric)
Dutch Testing Day - 2012

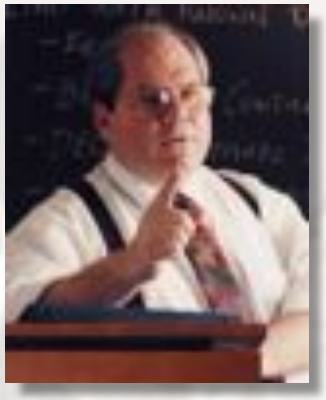
Exploration

Boost your testing power with ~~Exploratory Testing~~!



- What is exploratory testing?
- Strategy and techniques of ET
- SBTM: plan, structure & traceability
- How more exploration can boost the value of your testing
- How to master exploratory testing?

Exploratory Testing

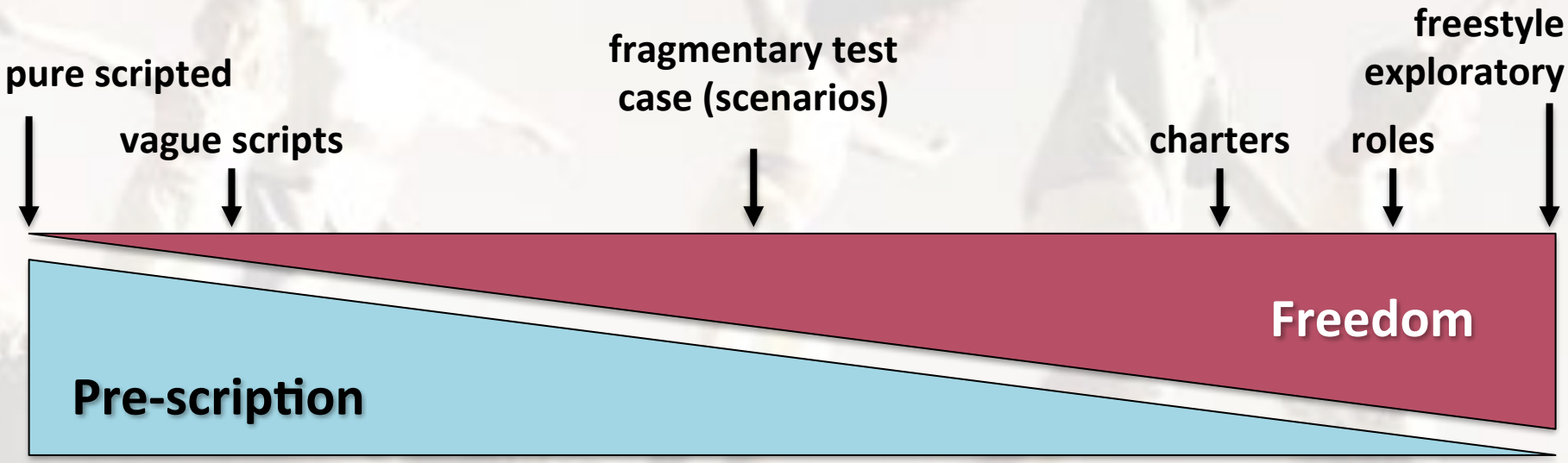


"a style of software testing that emphasizes the personal freedom and responsibility of the individual tester to continually optimize the quality of his/her work by treating test-related learning, test design, test execution, and test result interpretation as mutually supportive activities that run in parallel throughout the project"

Exploratory Testing

“An approach to software testing that emphasizes the personal freedom and responsibility of each tester to continually optimize the value of his work by treating learning, test design and test execution as mutually supportive activities that run in parallel throughout the project."

"Simultaneously designing and executing tests to learn about the system, using your insights from the last experiment to inform the next."



Source: Telling Your Exploratory Story by Jon Bach

Fairy tales

Do not apply ET if:

- Higher demands on demonstrability / reporting (eg. imposed standards)
- Critical functionality (failure = big damage)
- Inexperienced testers
- Test cases must be able to be executed by a different tester
- Test cases have to be re-usable
- No direct feedback from test execution (the results are not immediately available)
- Tests require a lot of preparation
- Testing needs to be on the critical path as short as possible

Source: TMap Next, chapter 14

Fairy tales

4 WHAT ARE THE DISADVANTAGES?

- Tests are performed on the fly, no review in advance.
- Test execution difficult to repeat in detail.
- Difficult to report test coverage.
- Difficult to maintain constant test quality.
- Process 'mystical' ('white coat authority').

Source: Fan booklet Exploratory testing (Microsoft)

Fairy tales

ISTQB
International Software Testing Qualifications Board

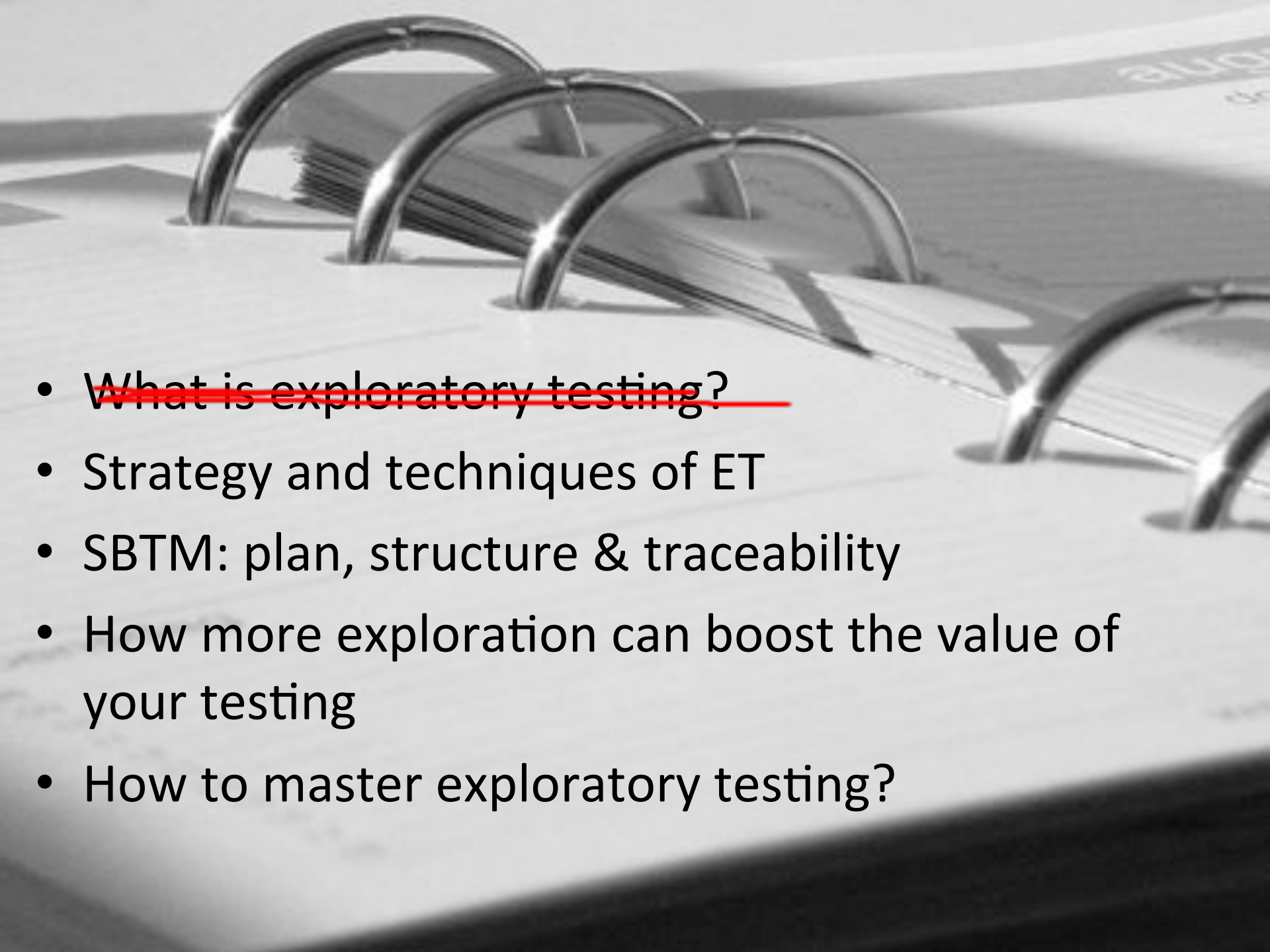
2.5.1 Test Implementation

Testing without scripts should not be ad hoc or aimless as this can be unpredictable in duration unless time boxed (see SBTM). Over the years, testers have developed a variety of experience-based techniques, such as attacks (see section 4.4 and [Whittaker03]), error guessing [Myers79], and exploratory testing. Test analysis, test design, and test implementation still occur, but they occur primarily during test execution. When following such dynamic test strategies, the results of each test influence the analysis, design, and implementation of the subsequent tests. While these strategies are lightweight and often effective at finding bugs, they require expert testers, can be unpredictable in duration, often do not provide good coverage information, and may be difficult to repeat without specific tool assistance for regression testing.

4.4.2 *Experienced-based techniques*

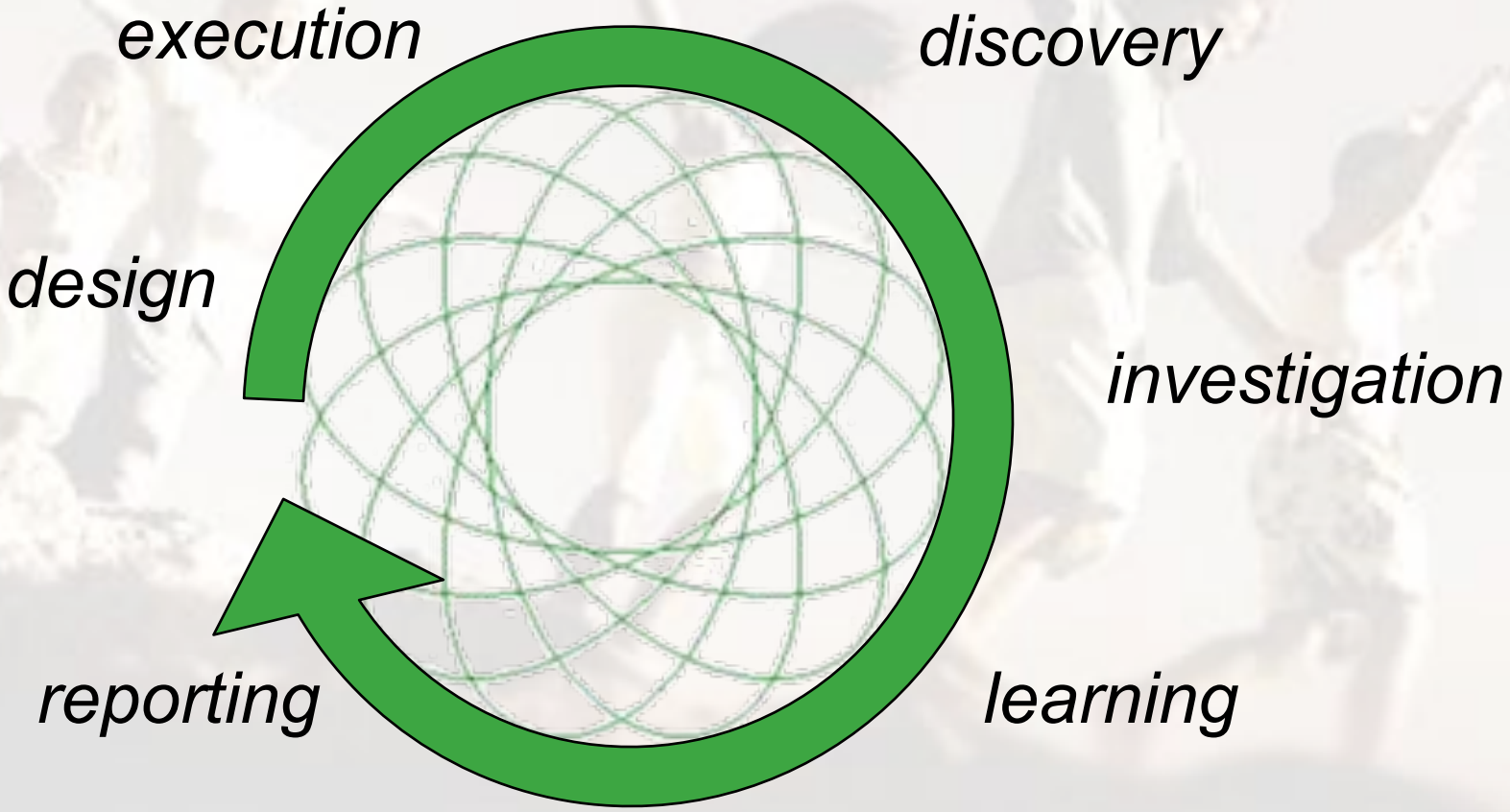
There are other test design techniques which consider defect history but do not necessarily have systematic coverage criteria. These are categorized as experienced-based test techniques

Experience-based tests utilize testers' skills and intuition, along with their experience with similar applications or technologies. These tests are effective at finding defects but not as appropriate as other techniques to achieve specific test coverage levels or producing reusable test procedures.



- ~~What is exploratory testing?~~
- Strategy and techniques of ET
- SBTM: plan, structure & traceability
- How more exploration can boost the value of your testing
- How to master exploratory testing?

Exploration: searching for value and risk



Source original picture: Two futures of testing by Michael Bolton

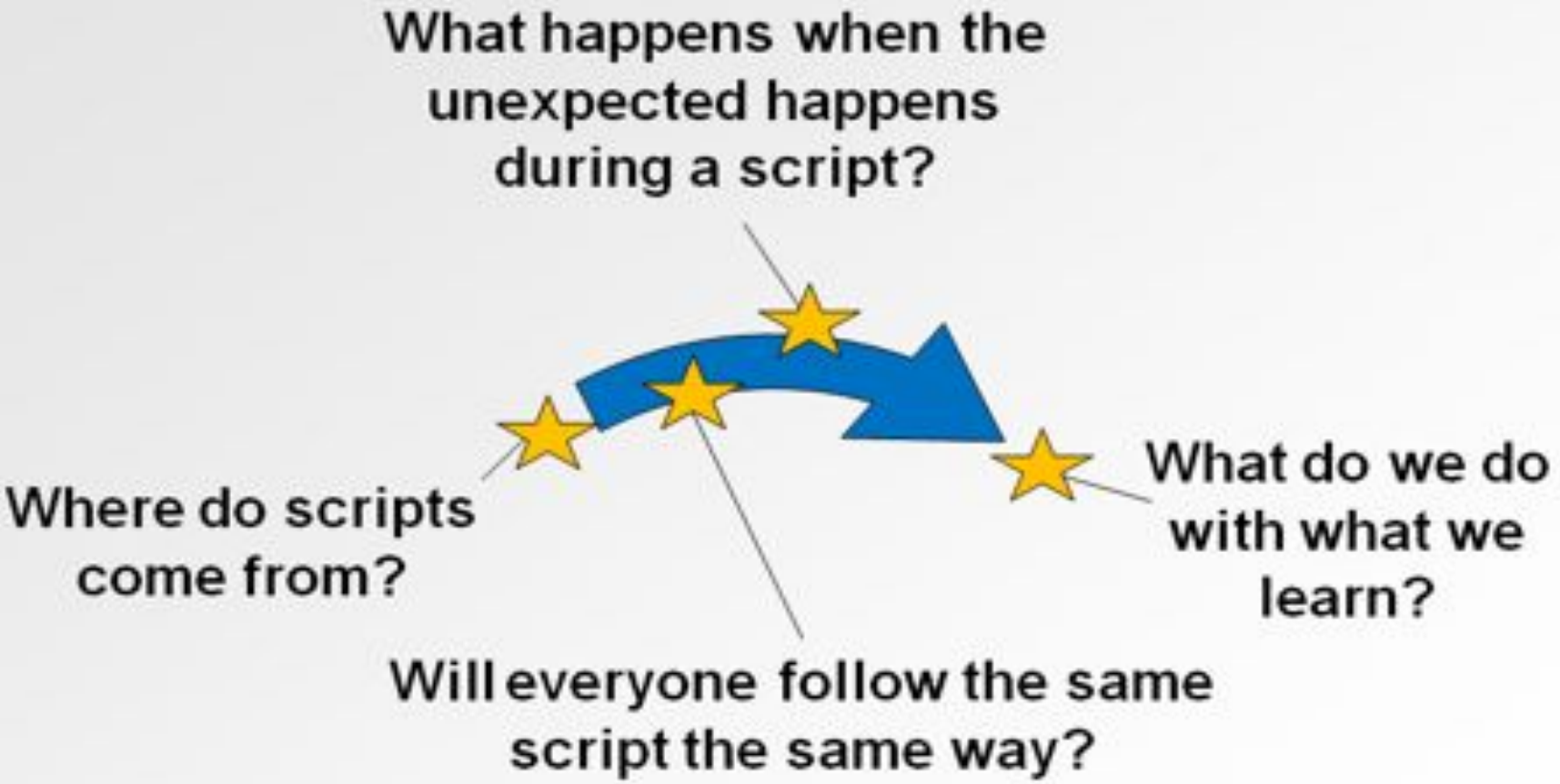
Questions About Exploration...

arrows and cycles

Source: Rapid Software Testing by James Bach & Michael Bolton
This slide is used with kind permission

Questions About Scripts...

arrows and cycles



Source: Rapid Software Testing by James Bach & Michael Bolton
This slide is used with kind permission

Answers About Scripts...

arrows and cycles

Source: Rapid Software Testing by James Bach & Michael Bolton
This slide is used with kind permission

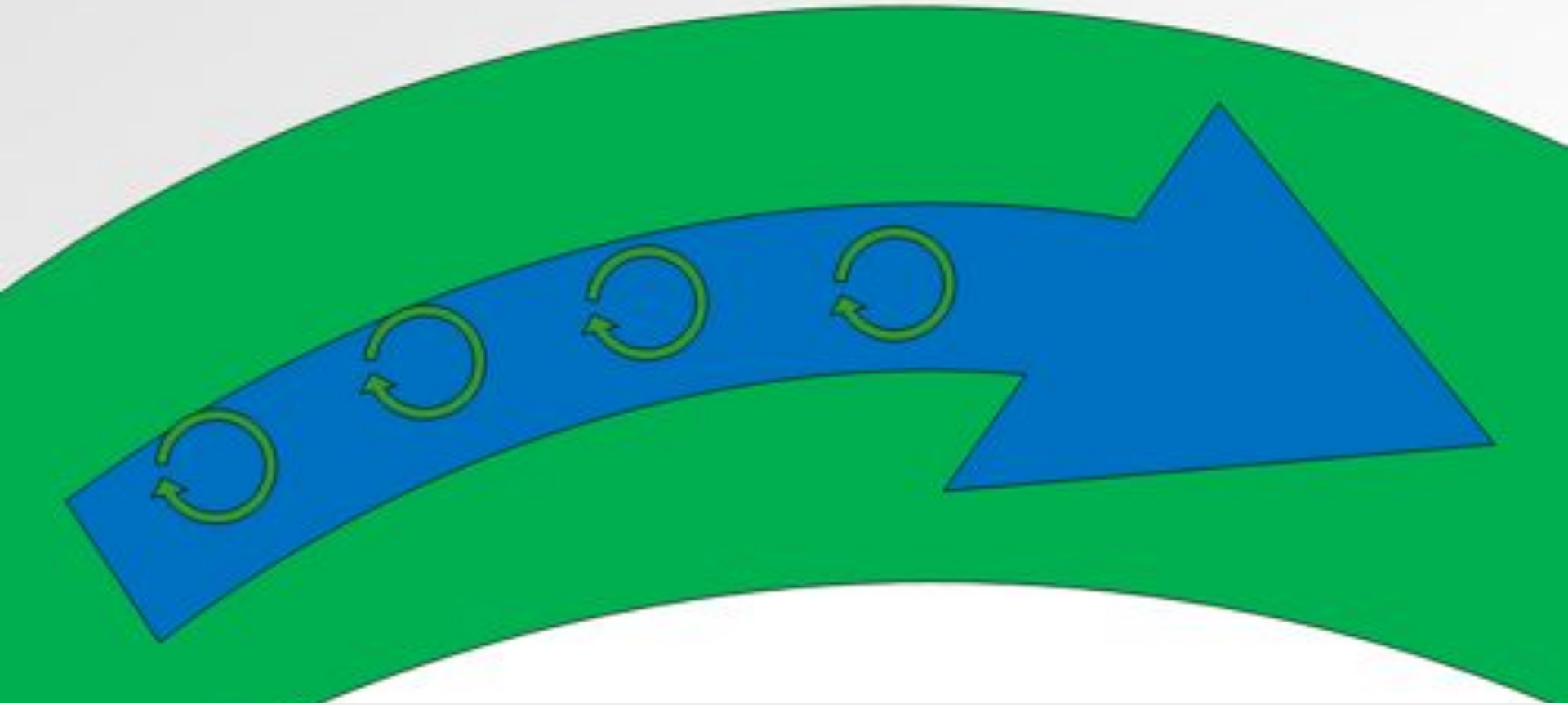
You can put them together!

arrows and cycles

Source: Rapid Software Testing by James Bach & Michael Bolton
This slide is used with kind permission

You can put them together!

arrows and cycles



Source: Rapid Software Testing by James Bach & Michael Bolton
This slide is used with kind permission

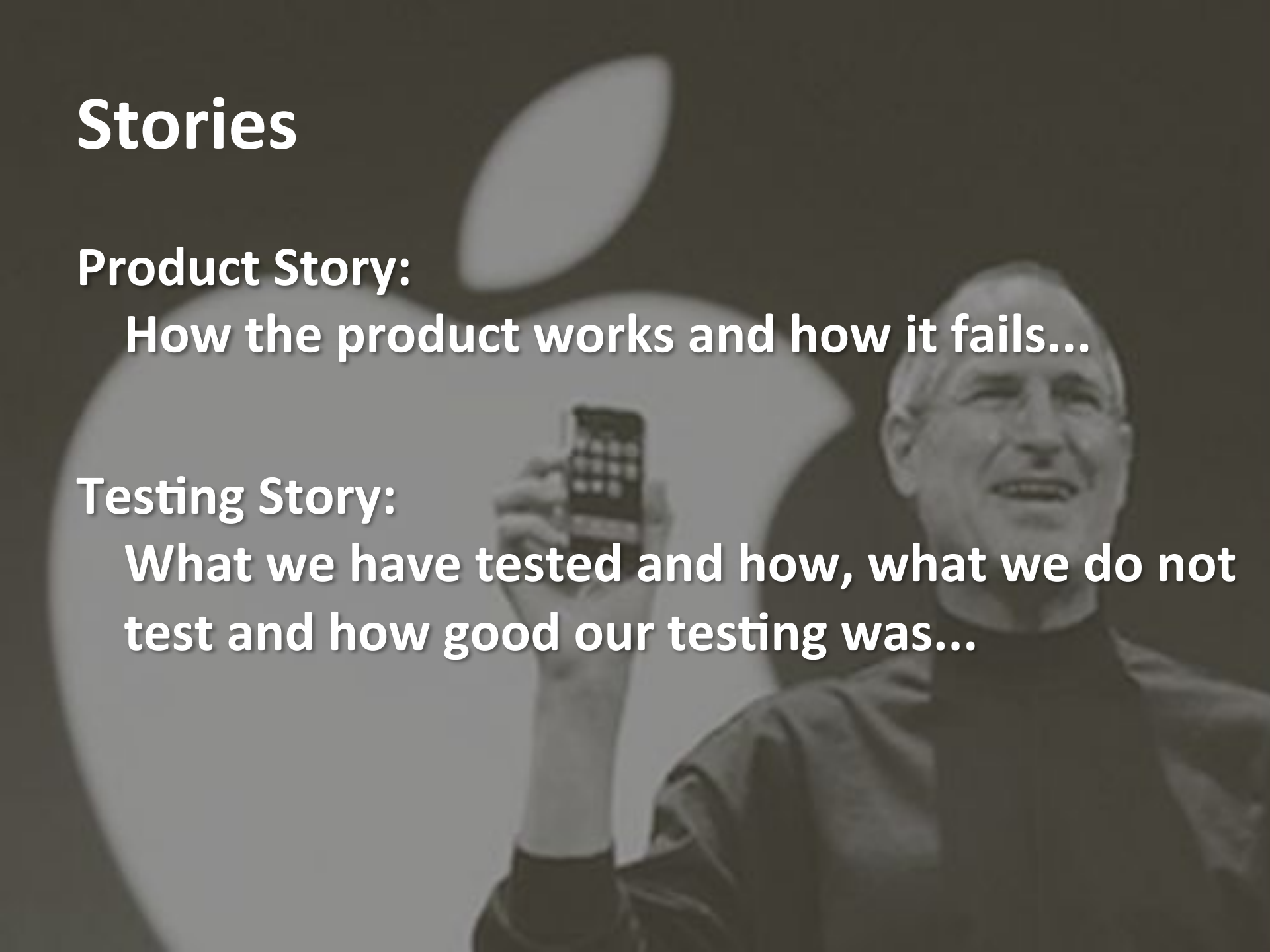
Stories

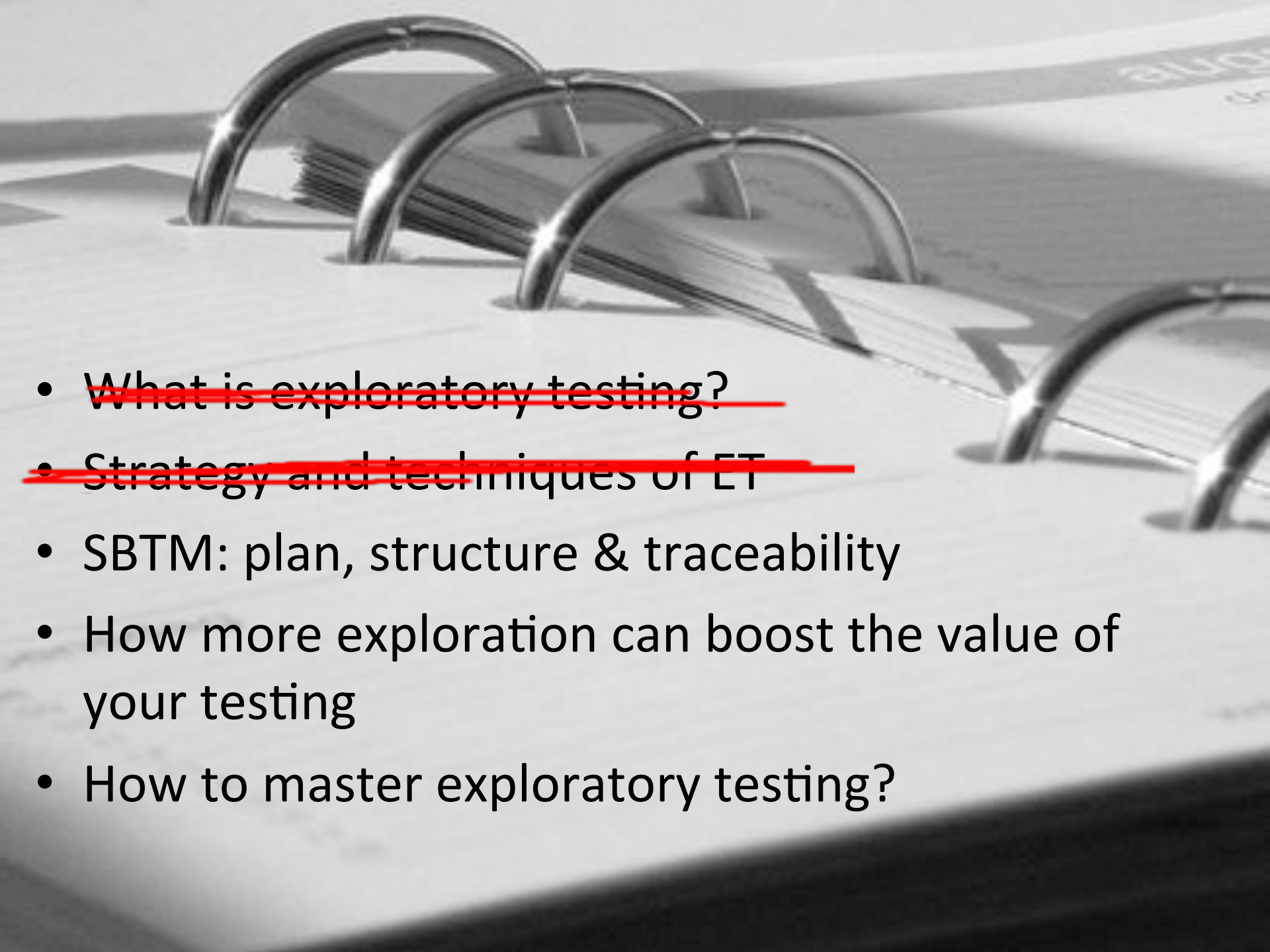
Product Story:

How the product works and how it fails...

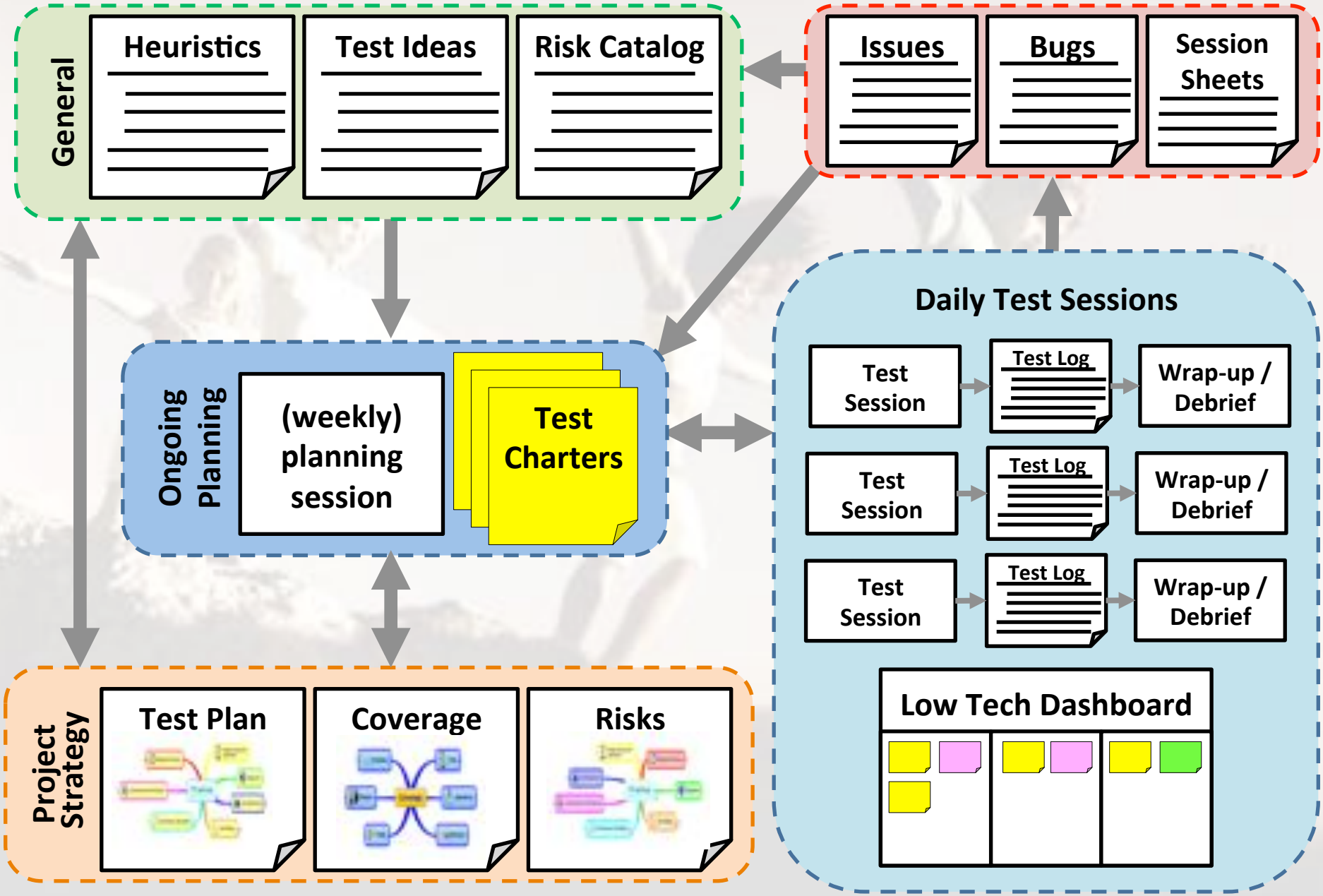
Testing Story:

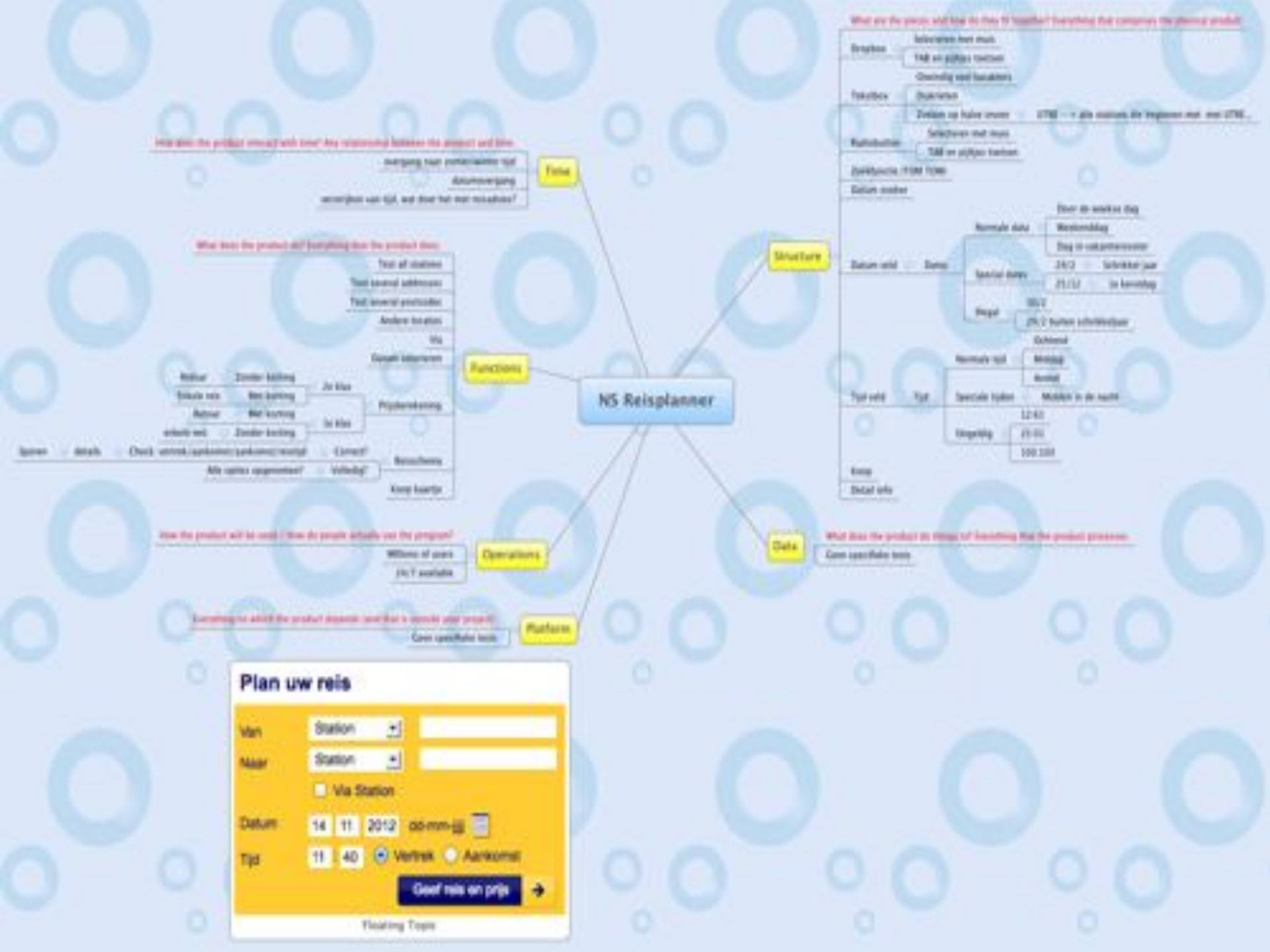
What we have tested and how, what we do not test and how good our testing was...

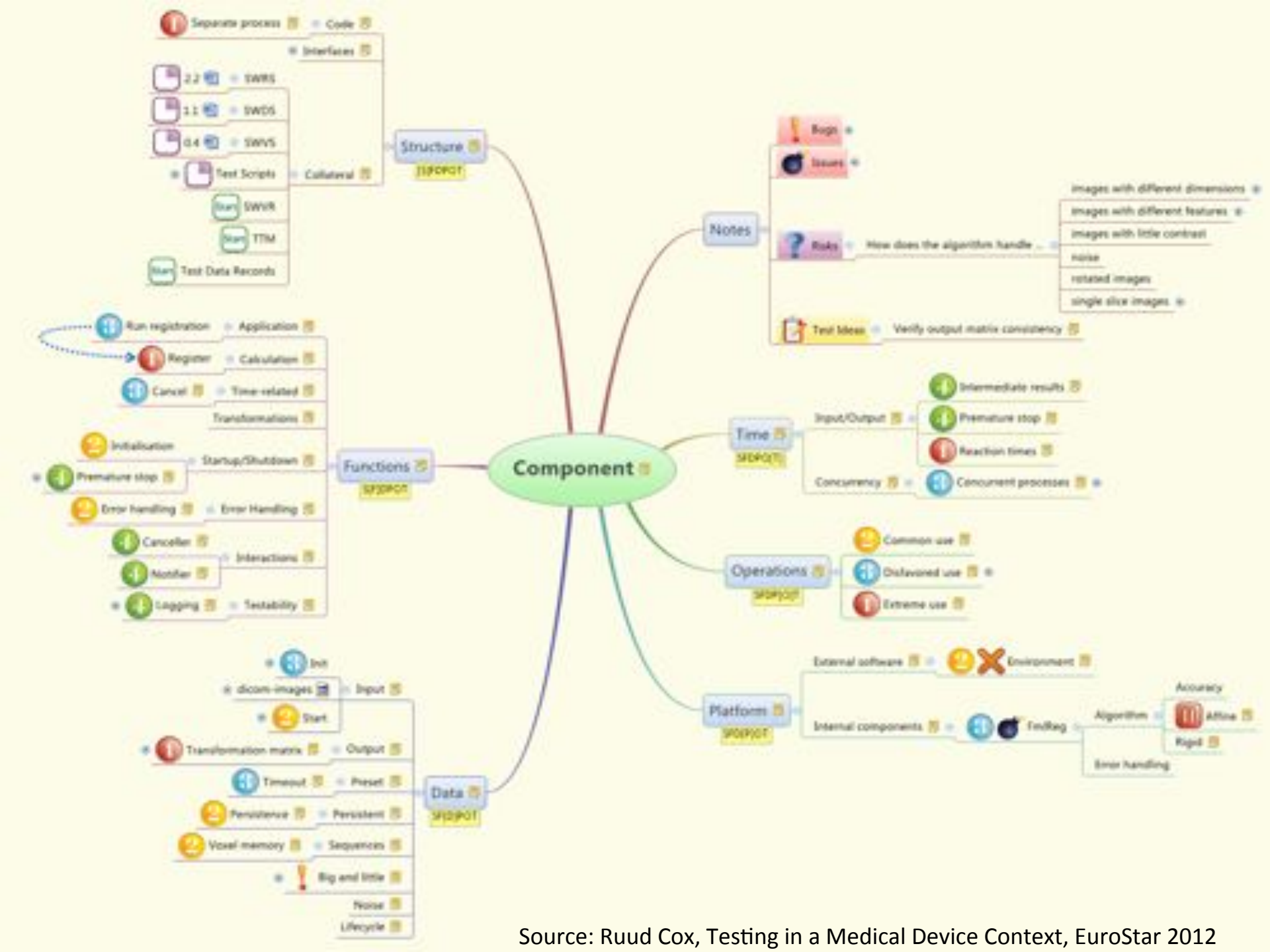




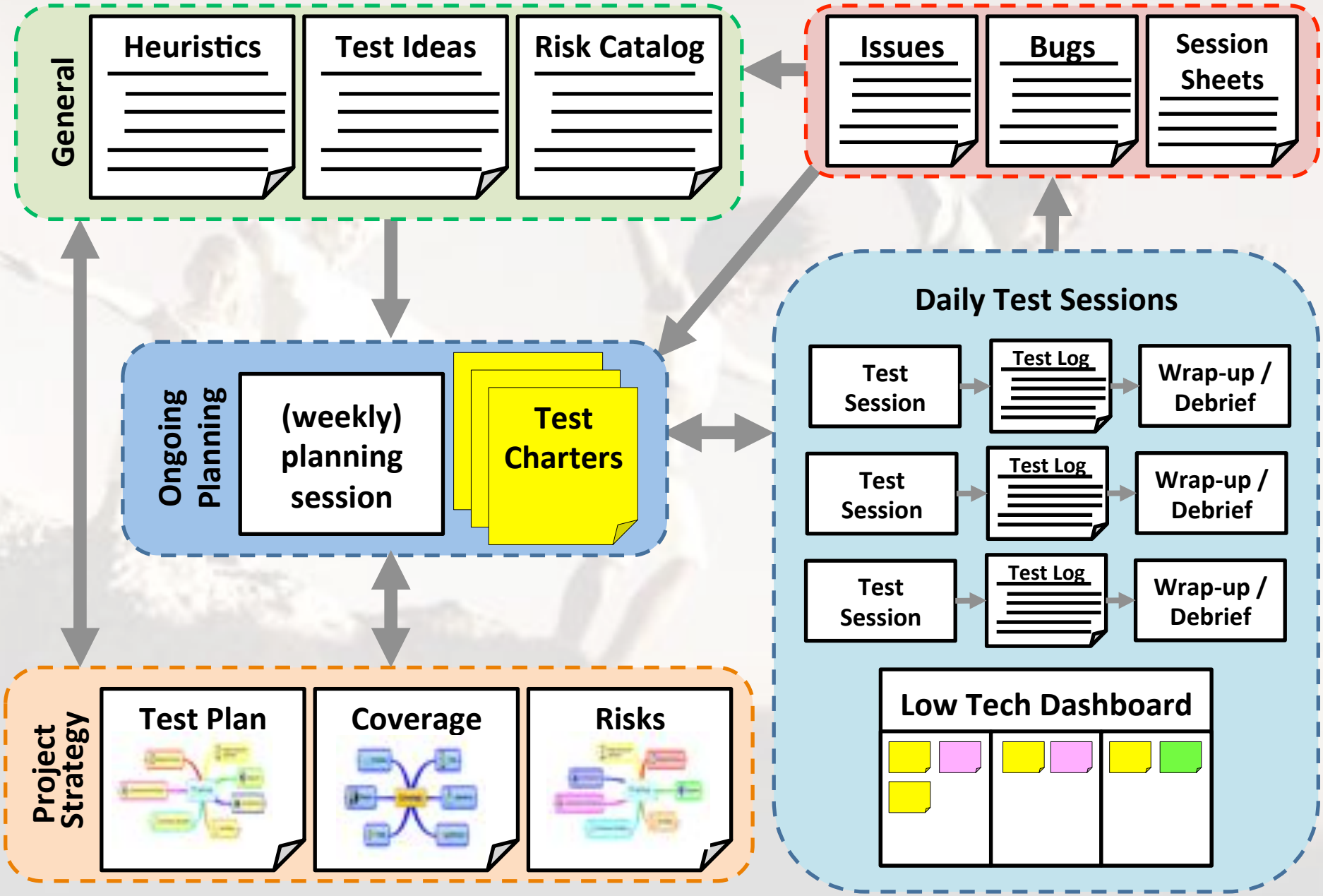
- ~~What is exploratory testing?~~
- ~~Strategy and techniques of ET~~
- SBTM: plan, structure & traceability
- How more exploration can boost the value of your testing
- How to master exploratory testing?


General
Heuristics
Test Ideas
Risk Catalog
Issues
Bugs
Session Sheets
Daily Test Sessions
Ongoing Planning
(weekly) planning session
Test Charters
Test Session
Test Log
Wrap-up / Debrief
Test Session
Test Log
Wrap-up / Debrief
Test Session
Test Log
Wrap-up / Debrief
Low Tech Dashboard
Project Strategy
Test Plan
Coverage
Risks


NS Reisplanner
Time
Functions
Operations
Platform
Structure
Data
Plan uw reis
Van
Station
Naar
Station
Via Station
Datum
14
11
2012
dd-mm-jjjj
Tijd
11
40
Vertrek
Aankomst
Geef reis en prijs
Floating Topic



Source: Ruud Cox, Testing in a Medical Device Context, EuroStar 2012


General
Heuristics
Test Ideas
Risk Catalog
Issues
Bugs
Session Sheets
Daily Test Sessions
Ongoing Planning
(weekly) planning session
Test Charters
Test Session
Test Log
Wrap-up / Debrief
Test Session
Test Log
Wrap-up / Debrief
Test Session
Test Log
Wrap-up / Debrief
Low Tech Dashboard
Project Strategy
Test Plan
Coverage
Risks

Test Charters

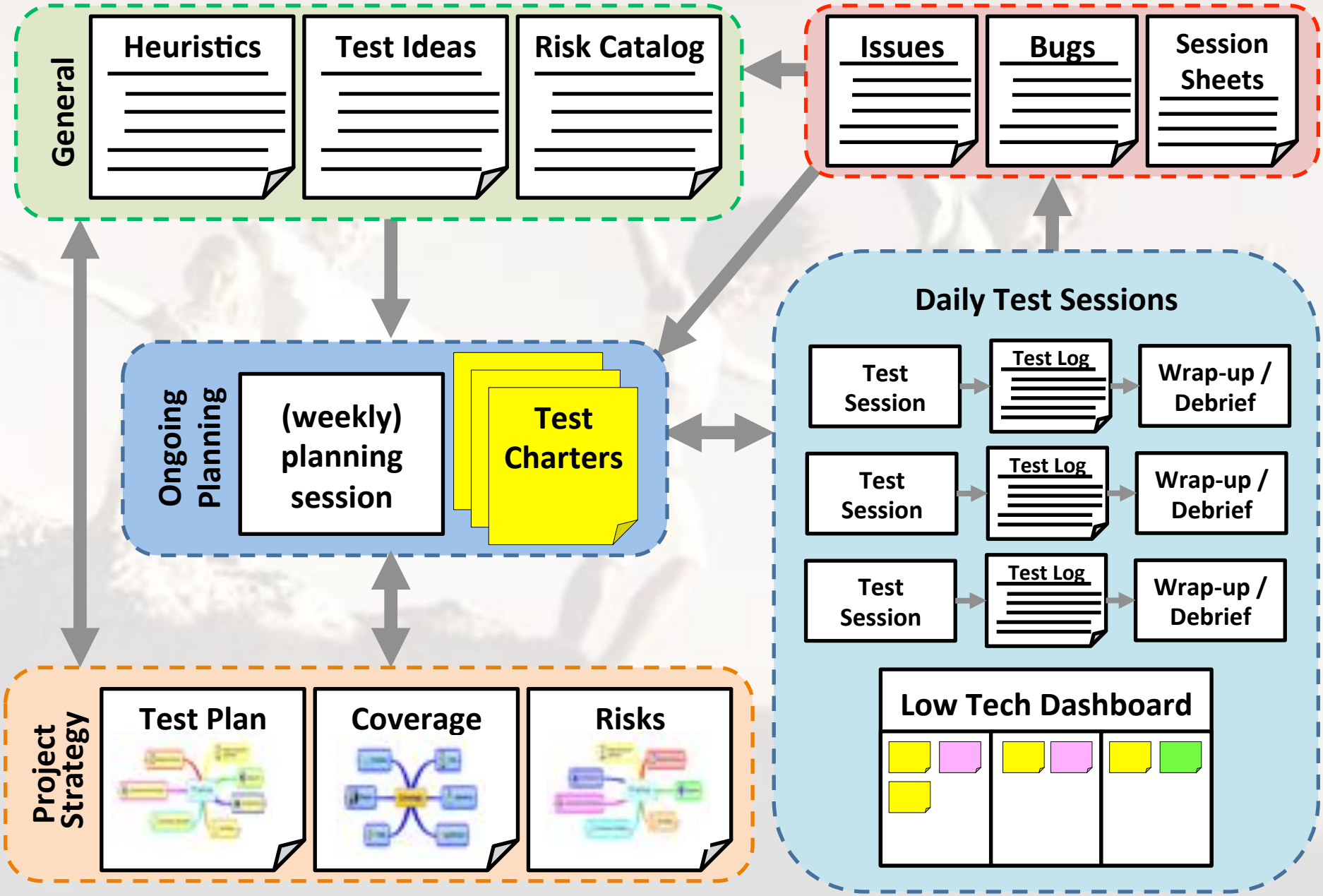
Explore (target)
With (resources)
To discover (information)

Explore Application X import menu. Identify important features with the goal of developing a coverage outline and a risk list.

My mission is to test <insert risk here> to <insert coverage here>

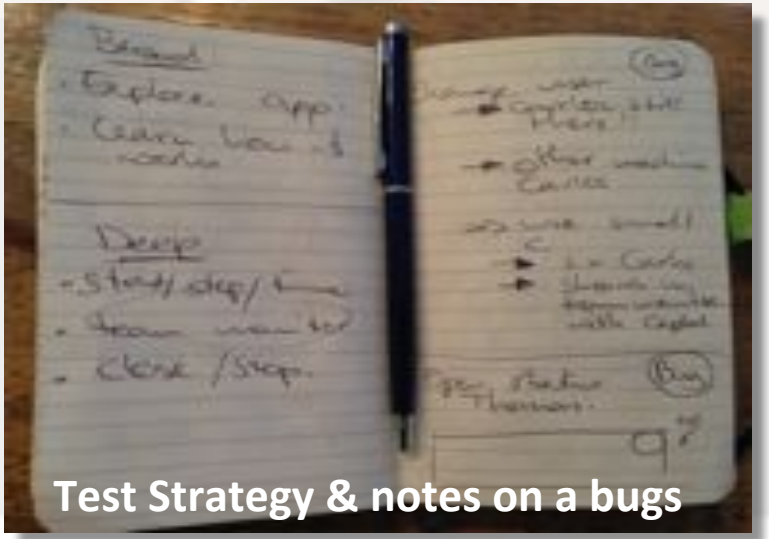
Read Chapter 4 of the product specification. Prepare a mind map, and discuss it with Peter (programmer) and David (architect).

The *charter is a one- to three-sentence mission for a testing session*

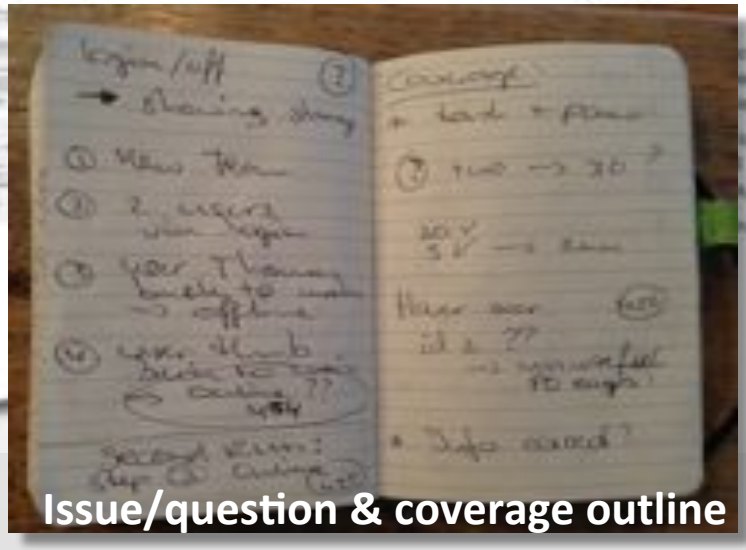

General
Heuristics
Test Ideas
Risk Catalog
Issues
Bugs
Session Sheets
Daily Test Sessions
Ongoing Planning
(weekly) planning session
Test Charters
Test Session
Test Log
Wrap-up / Debrief
Test Session
Test Log
Wrap-up / Debrief
Test Session
Test Log
Wrap-up / Debrief
Low Tech Dashboard
Project Strategy
Test Plan
Coverage
Risks

Test log: my notes

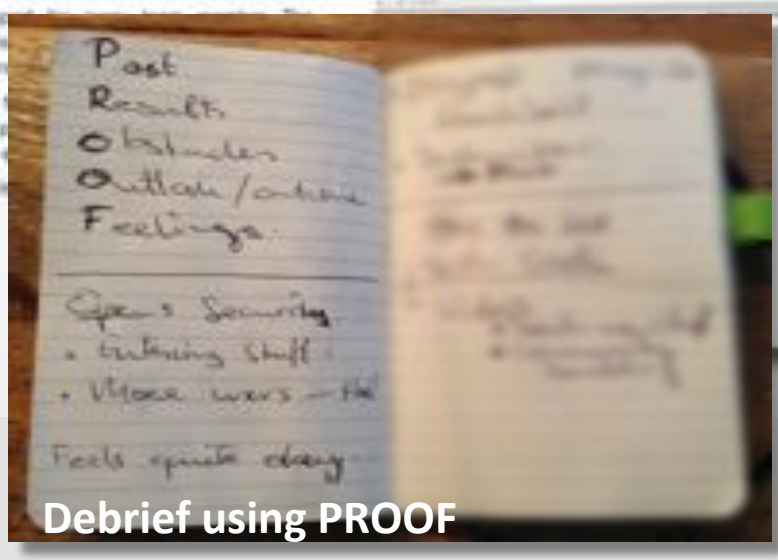
Preparation: talk to product owner



Test Strategy & notes on a bugs



Issue/question & coverage outline

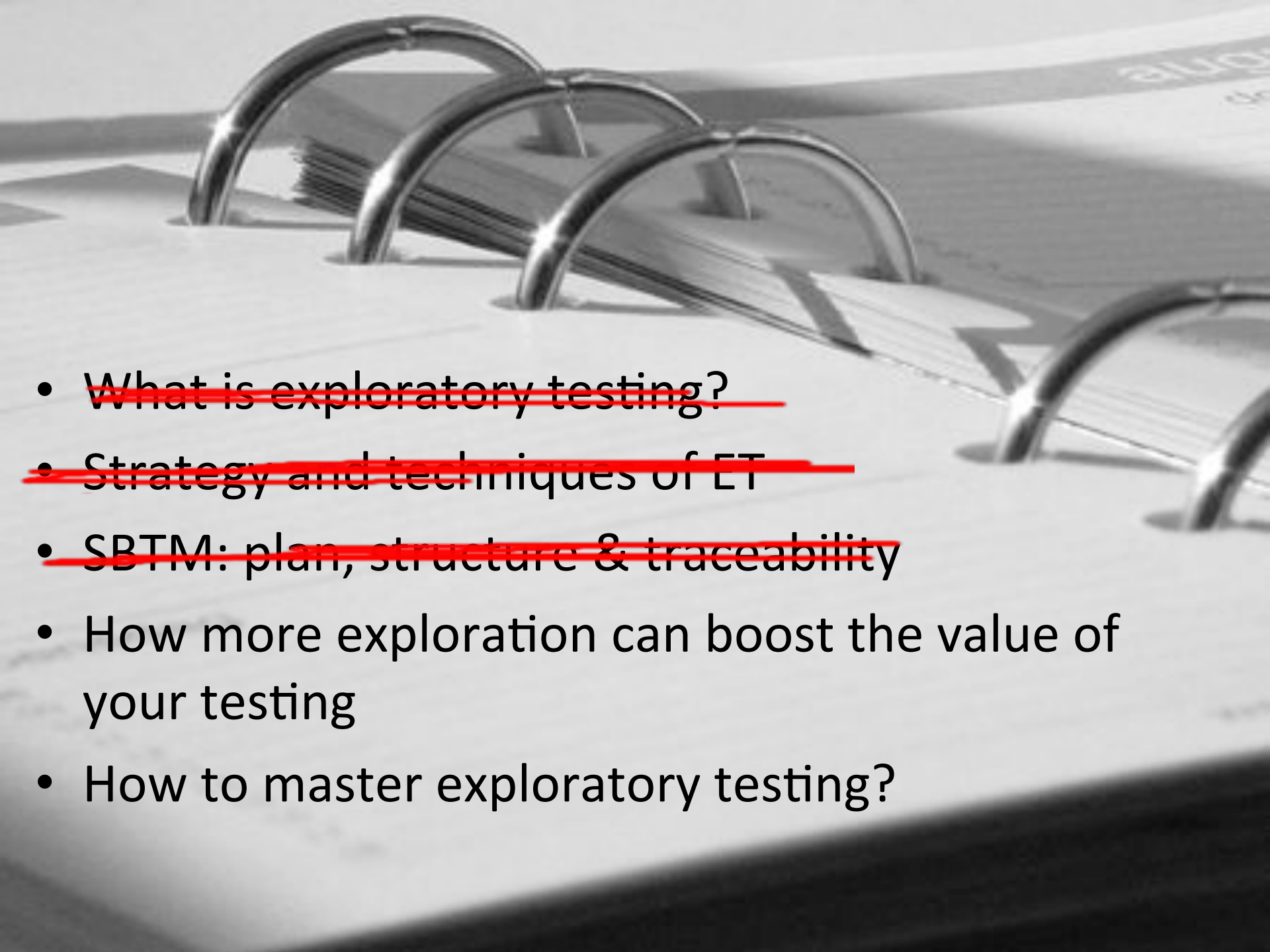


Debrief using PROOF

Session sheet

- Charter
- Coverage areas (product areas, product elements, quality criteria or test techniques)
- Start Time + Session duration (long, normal, or short)
- Tester Name(s)
- Charter/opportunity time , expressed as a time breakdown:
  - Test design and execution (as a percentage of the total on-charter time)
  - Bug investigation and reporting (as a percentage of the total on-charter time)
  - Session setup (as a percentage of the total on-charter time)
- Data Files
- Test Log (notes the tester made during testing)
- Bugs (a problem that threatens the value of the product)
- Issues (a problem that threatens the value of the testing process)

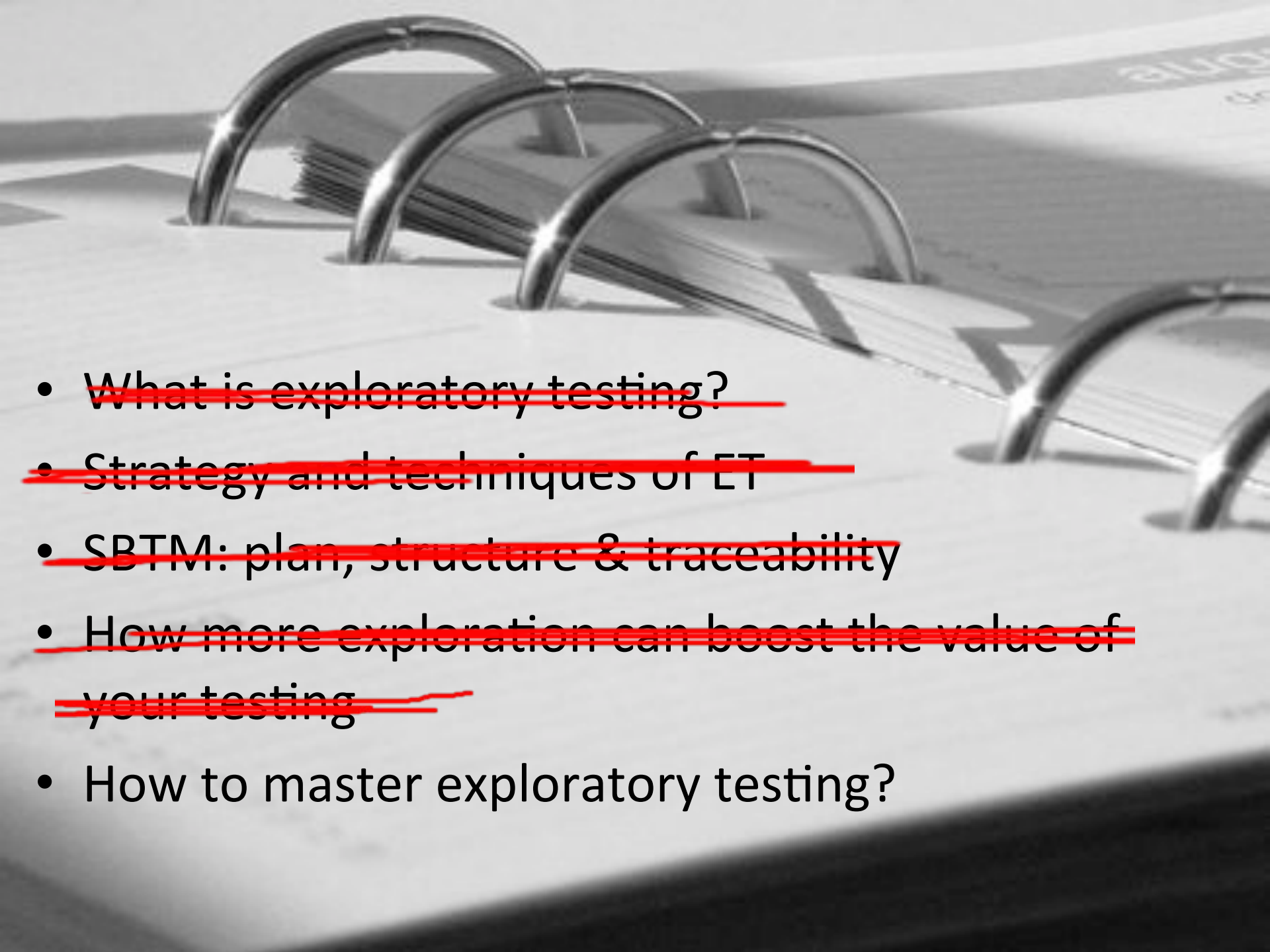
Source: An Exploratory Tester's Notebook by Michael Bolton



- ~~What is exploratory testing?~~
- ~~Strategy and techniques of ET~~
- ~~SBTM: plan, structure & traceability~~
- How more exploration can boost the value of your testing
- How to master exploratory testing?

More exploration can boost the value of testing

- Do what really needs to be done
- Create engagement: help people use their brains
- Take advantage of tacit knowledge and skill
- Use insights from experiments to inform the next
- Beware of the creative power of exploration



- ~~What is exploratory testing?~~
- ~~Strategy and techniques of ET~~
- ~~SBTM: plan, structure & traceability~~
- ~~How more exploration can boost the value of your testing~~
- How to master exploratory testing?

Challenges?

Exploratory Testing = not easy and needs (a lot) training and practice

Often seen as most difficult:

- Note taking
- Test ideas
- Coverage reporting
- Managing ET

How to master exploratory testing?

- Just do it!
- Practice, practice and practice some more…
- Pair with colleagues
- Testing Dojo
- Debrief & retrospect
- Train creating test ideas fast
- Learn more about observation, experiments, biases, social science, ...

Agenda

- Background
- Introducing different "flavors" of exploratory test
- Challenges – and what to do about it
- What we ended up with – our "flavors"
- What's next

Exploratory Flavors

- We tried a number of different flavors:
  - Freestyle exploratory
  - Session based exploratory
  - Testing tours
  - Bug hunts
  - General functionality and stability test procedure (Microsoft)

Source: Gitte Ottosen – Many flavours of exploratory testing (Agile Testing days 2012)

http://trishkhoo.com/

My personal experience

“Testing is about questioning & learning under conditions of fundamental uncertainty.”

RST

“A tester knows that things can be different”

Jerry Weinberg

“I've participated and organized many testing competitions. I've never seen someone win such a competition by writing down tests. Doesn't that tell us something?”

James Bach

If you cannot trust your testers, you do not make them write more detailed test case. But you train them!

Rikard Edgren – EuroStar 2012
Gitte Ottosen – ATD 2012

Image: http://corkboard.me

More information

http://www.huibschoots.nl/links

codecentric
agile software factory

huib.schoots@codecentric.nl
+31 6 - 24641033
@huibschoots
www.magnifiant.com