IMPROVE
Quality Services

TESTNET

Workshop TestNet
Najaarsevemenent 2014

Exploratory Testing

huib.schoots@improveqs.nl

06-24641033

Acknowledgements

Met dank aan:

- Keri Smith voor inspiratie
- Ruud Cox voor de vele discussies en hulp bij voorbereiding
- Uiteraard James Bach en Michael Bolton voor het delen van hun kennis over Rapid Software Testing, Exploratory Testing en Session Based Test Management

Sommige slides zijn met toestemming overgenomen uit Rapid Software Testing (http://www.satisfice.com/info_rst.shtml).

Rapid Software Testing is ontwikkeld door James Bach en Michael Bolton.

Debrief Exercises

Observe

Compare (analyze/evaluate)

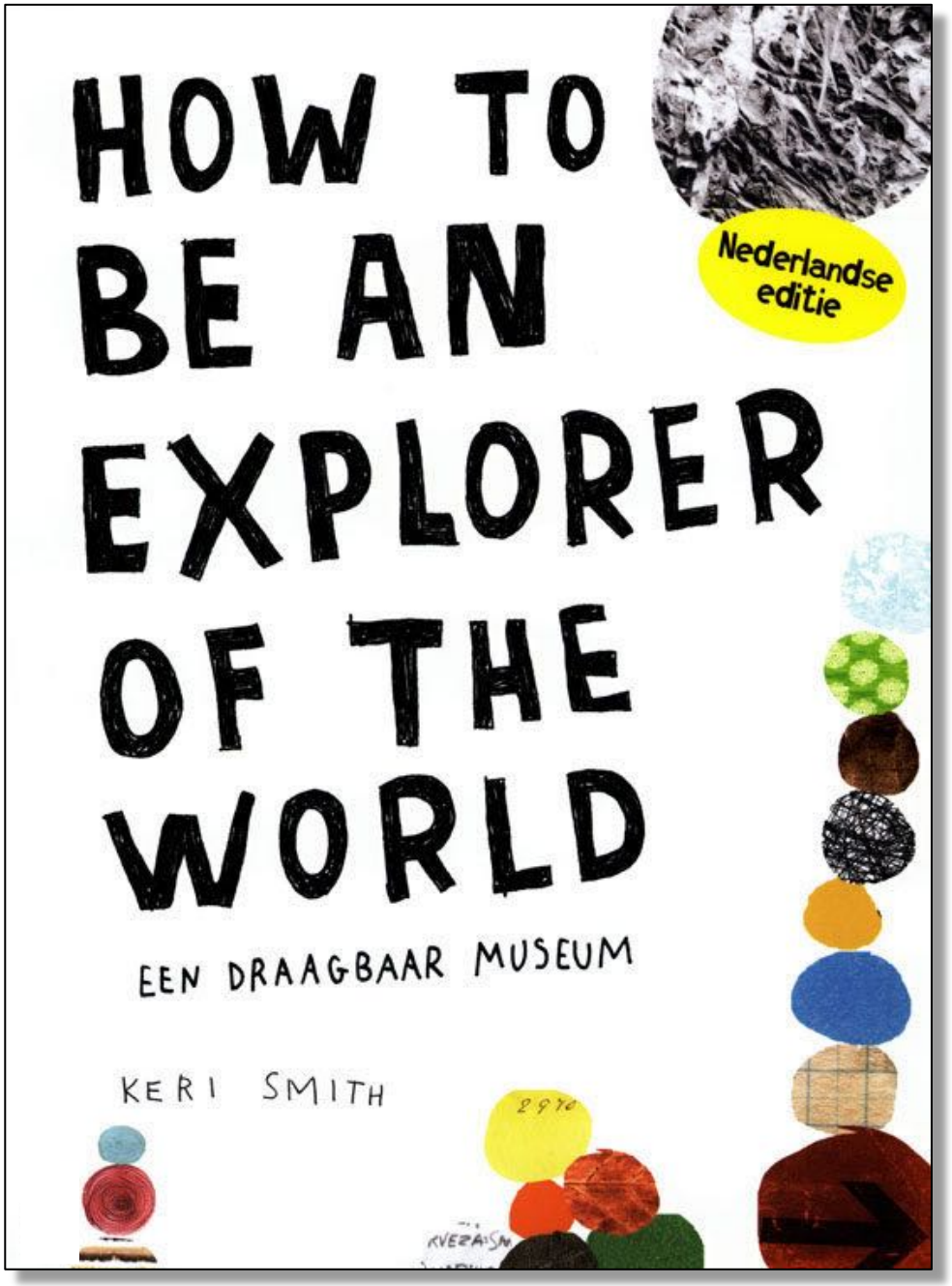
Test: Procedure *has four elements*

Addresses a motivating question

- **Configure**
  - *(if necessary)* Obtain product for testing
  - *(if necessary)* Install on a platform.
  - *(if necessary)* Prepare test data and tools for test execution.
  - Assure that the product is in a "clean enough" starting state.
- **Operate**
  - Control the product and platform with inputs to exercise the product.
  - Cause the product to exercise the right functions/states in the right sequence with the right data.
- **Observe**
  - Collect information about how the product behaves (collect both direct and indirect output) so that it can be evaluated.
- **Evaluate**
  - Apply oracles to detect bugs.

Provides a clear answer to the question

art by & inspired by KERI SMITH


HOW TO
BE AN
EXPLORER
OF THE
WORLD
Nederlandse editie
EEN DRAAGBAAR MUSEUM
KERI SMITH

Keri Smith is een Canadees conceptueel kunstenaar en auteur van diverse bestsellers over creativiteit. De belangrijkste focus van haar werk/onderzoek ligt op het creeren van wat de schrijver Umberto Eco genaamd "Open werken", stukken die worden ingevuld door de lezer/gebruiker.

Kunstenaars en wetenschappers analyseren de wereld om hen heen in verrassend gelijkaardige manieren: door te observeren, verzamelen, documenteren, analyseren en vergelijken. In haar boeiende "guided journals" worden lezers aangemoedigd om hun wereld verkennen als kunstenaars en wetenschappers.

Wat is haar missie? Om de wereld om je heen observeren en te documenteren. Alsof je nog het voor het eerst ziet... Maak notities. Verzamel dingen die je vindt op je reizen. Documenteer je bevindingen. Herken patronen. Kopieer. Speur. Focus op een ding tegelijk. Volg waardoor je wordt aangetrokken?

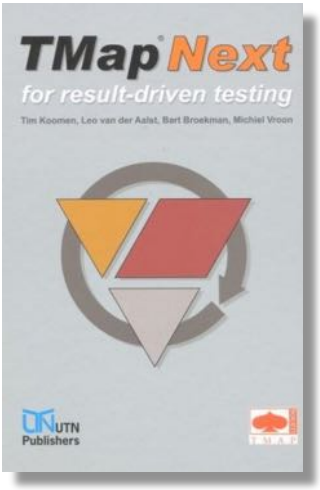
OBSERVEER
VERZAMEL
ANALYSEER
VERGELIJK
HEB OOG VOOR
PATRONEN

Inleidende discussie…

Discussie

- Wat is ET?
- Wat is het doel van ET?
- Wie bedienen we met ET?

TMap Next



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

Microsoft

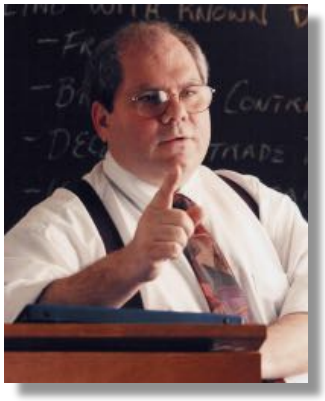
4 WHAT ARE THE DISADVANTAGES?

- 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)

Wat is Exploratory Testing?

Exploratory Testing is …



"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 is …

“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."

Exploratory Testing is …

“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."

How does it work?

Call this “Checking” not Testing

operating a product to check specific facts about it…

means

Interact with the product in specific ways to collect specific observations.

Apply algorithmic decision rules to those observations.

Report any failed checks.

A Check Has Three Elements

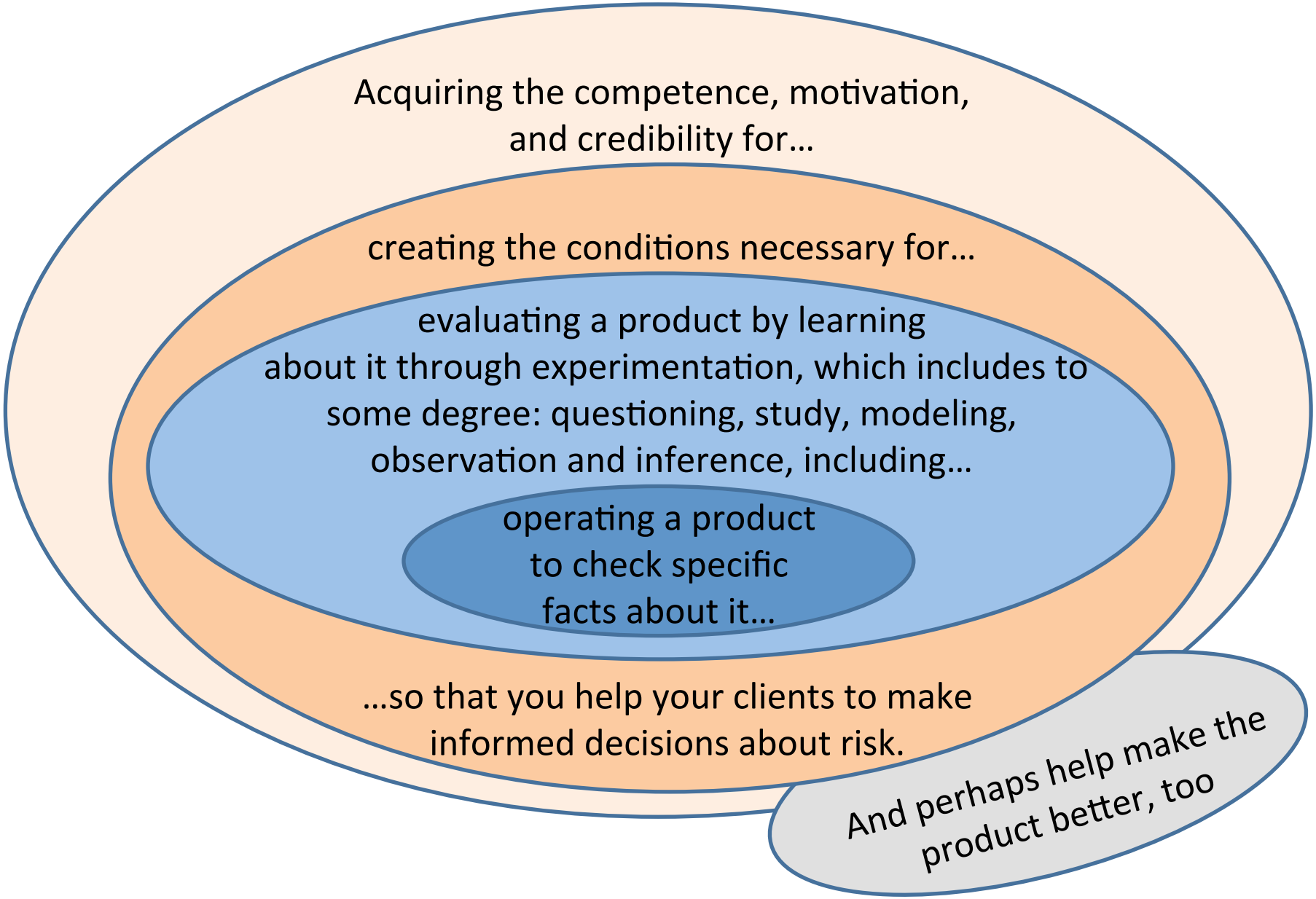
1. An *observation* linked to…
2. A *decision rule* such that…
3. both observation and decision rule can be applied algorithmically.

A ***check*** can be performed

by a machine
that *can't* think
(but that is quick and precise)

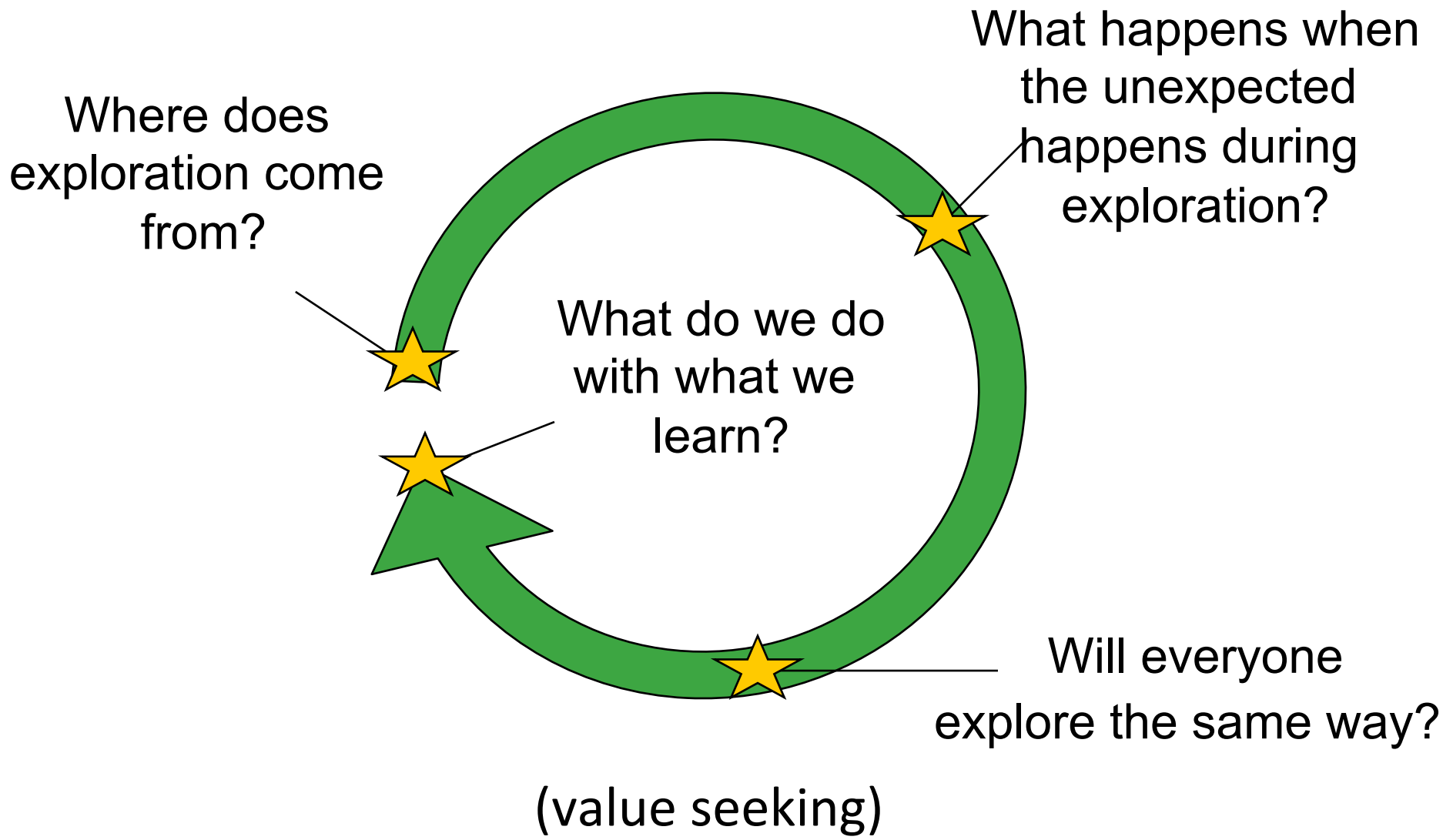
by a human who has been
instructed *not* to think
(and who is slow and variable)

Testing is…


Acquiring the competence, motivation, and credibility for…
creating the conditions necessary for…
evaluating a product by learning about it through experimentation, which includes to some degree: questioning, study, modeling, observation and inference, including…
operating a product to check specific facts about it…
…so that you help your clients to make informed decisions about risk.
And perhaps help make the product better, too

Questions About Exploration…

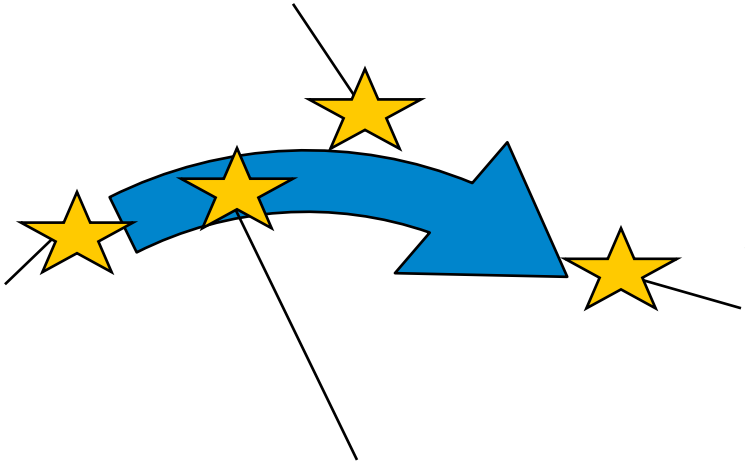
arrows and cycles



Questions About Scripts…

arrows and cycles

What happens when the unexpected happens during a script?



Where do scripts come from?

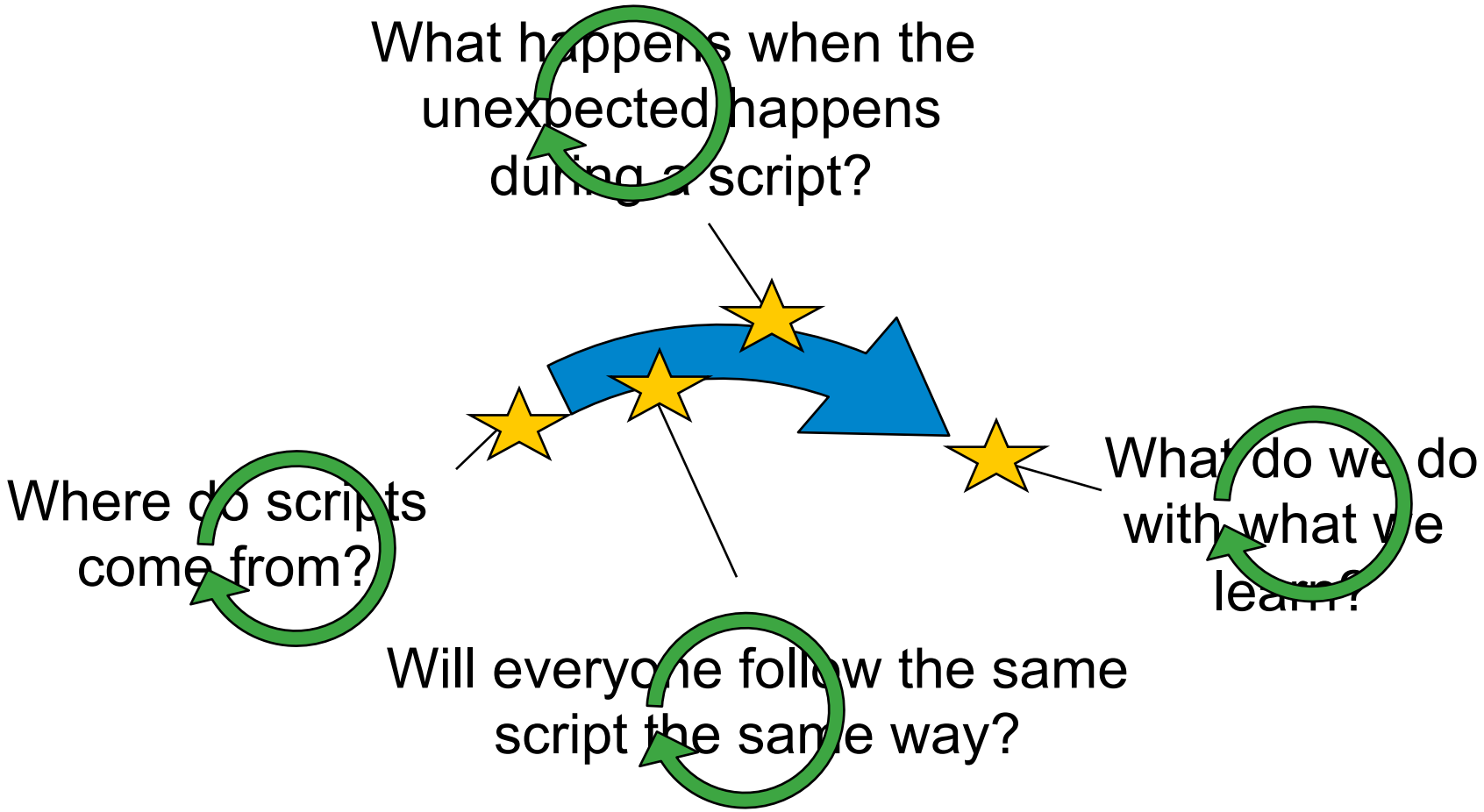
What do we do with what we learn?

Will everyone follow the same script the same way?

(task performing)

Answers About Scripts…

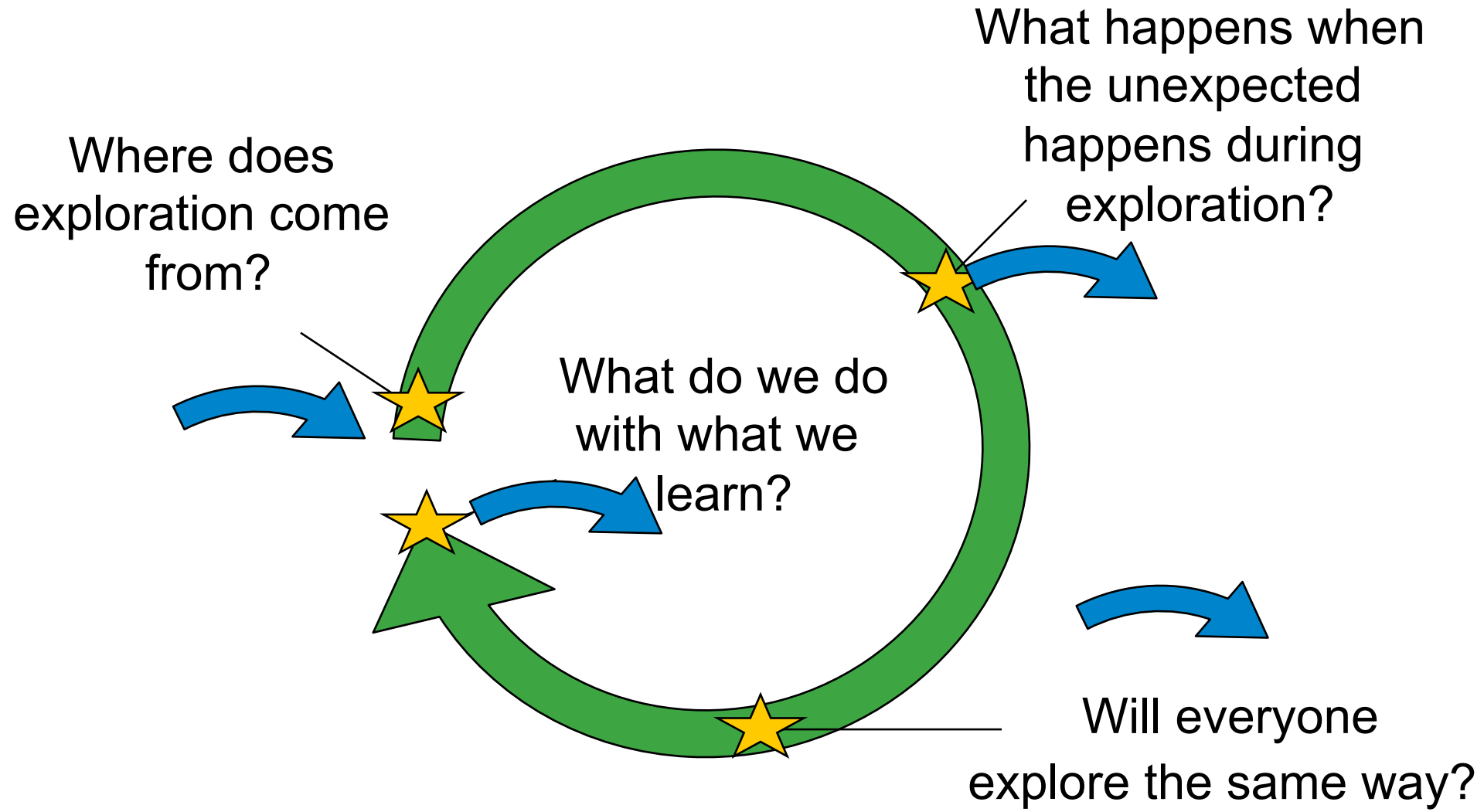
arrows and cycles



(task performing)

Questions About Exploration…

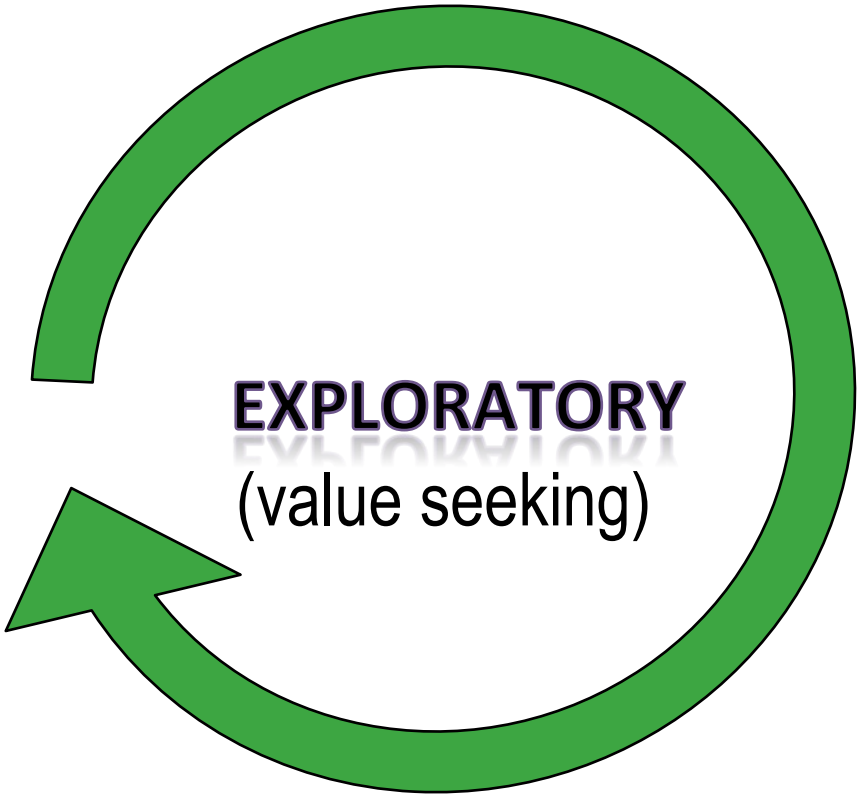
arrows and cycles

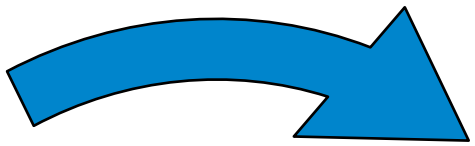


(value seeking)

You can put them together!

arrows and cycles



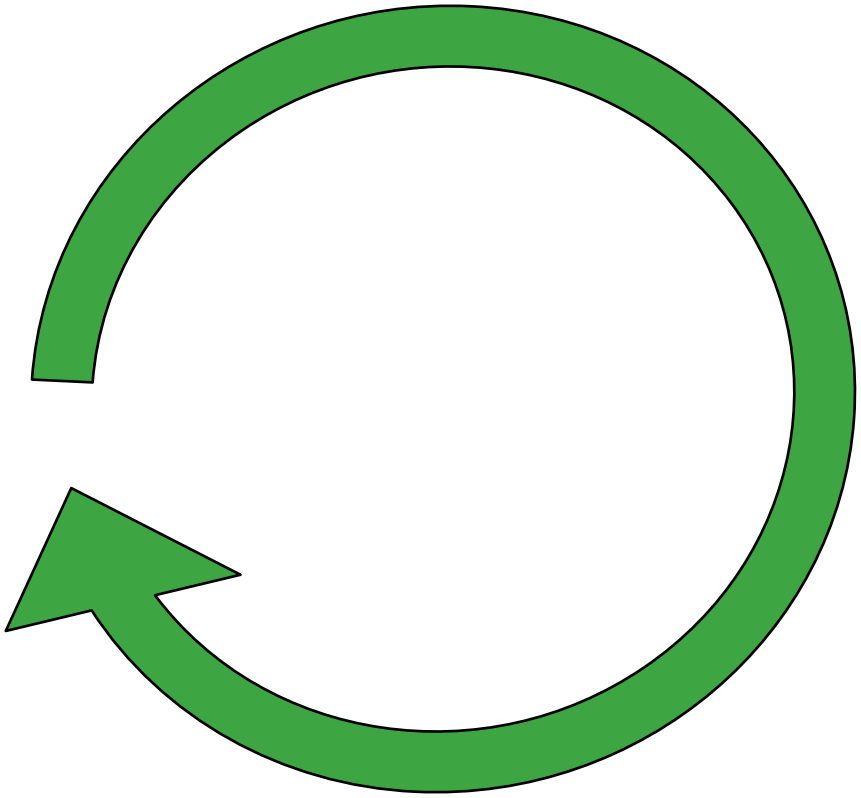


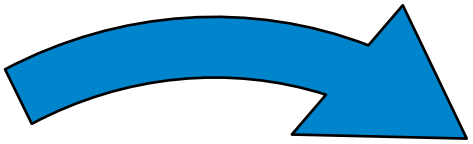
SCRIPTED

(task performing)

You can put them together!

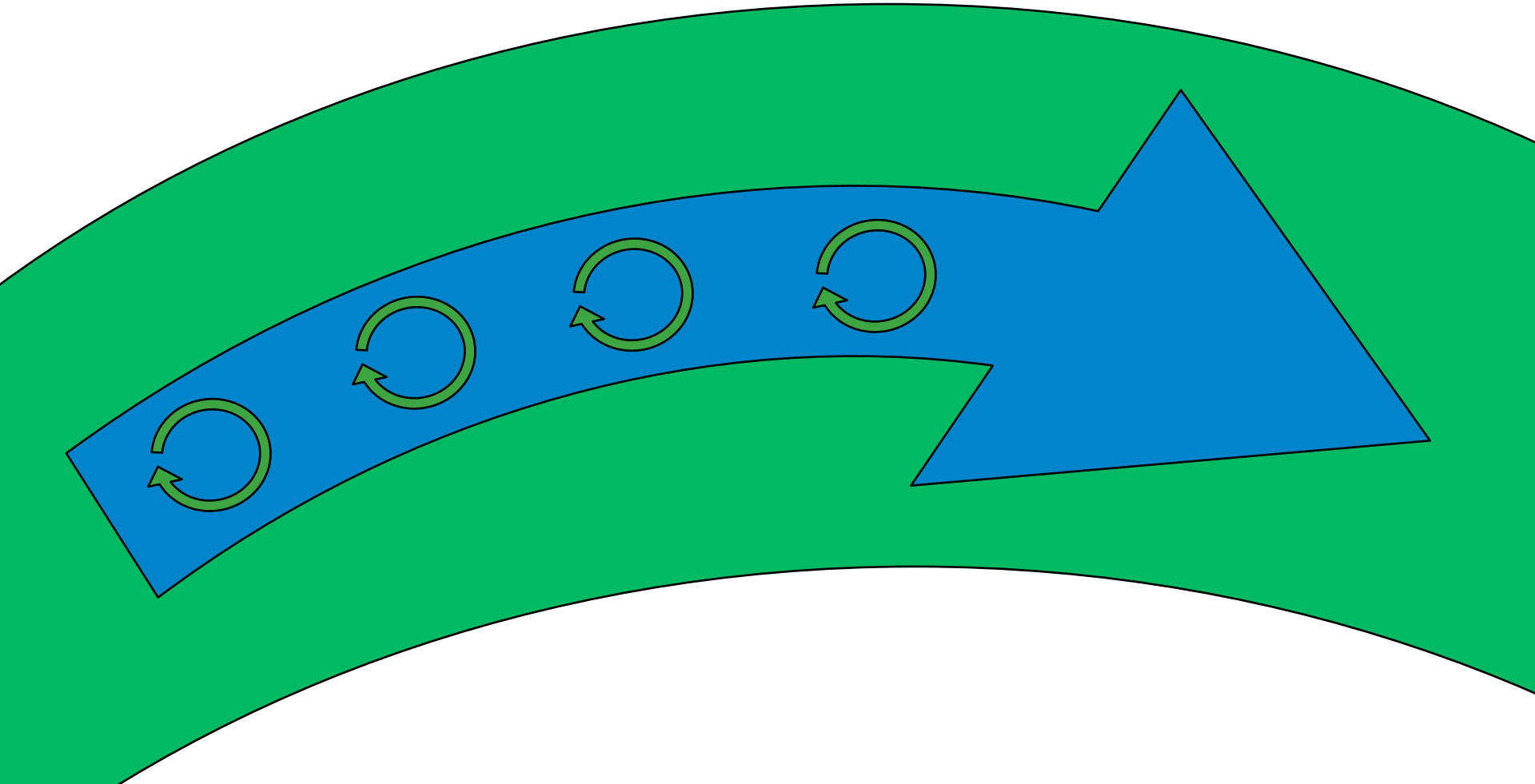
arrows and cycles



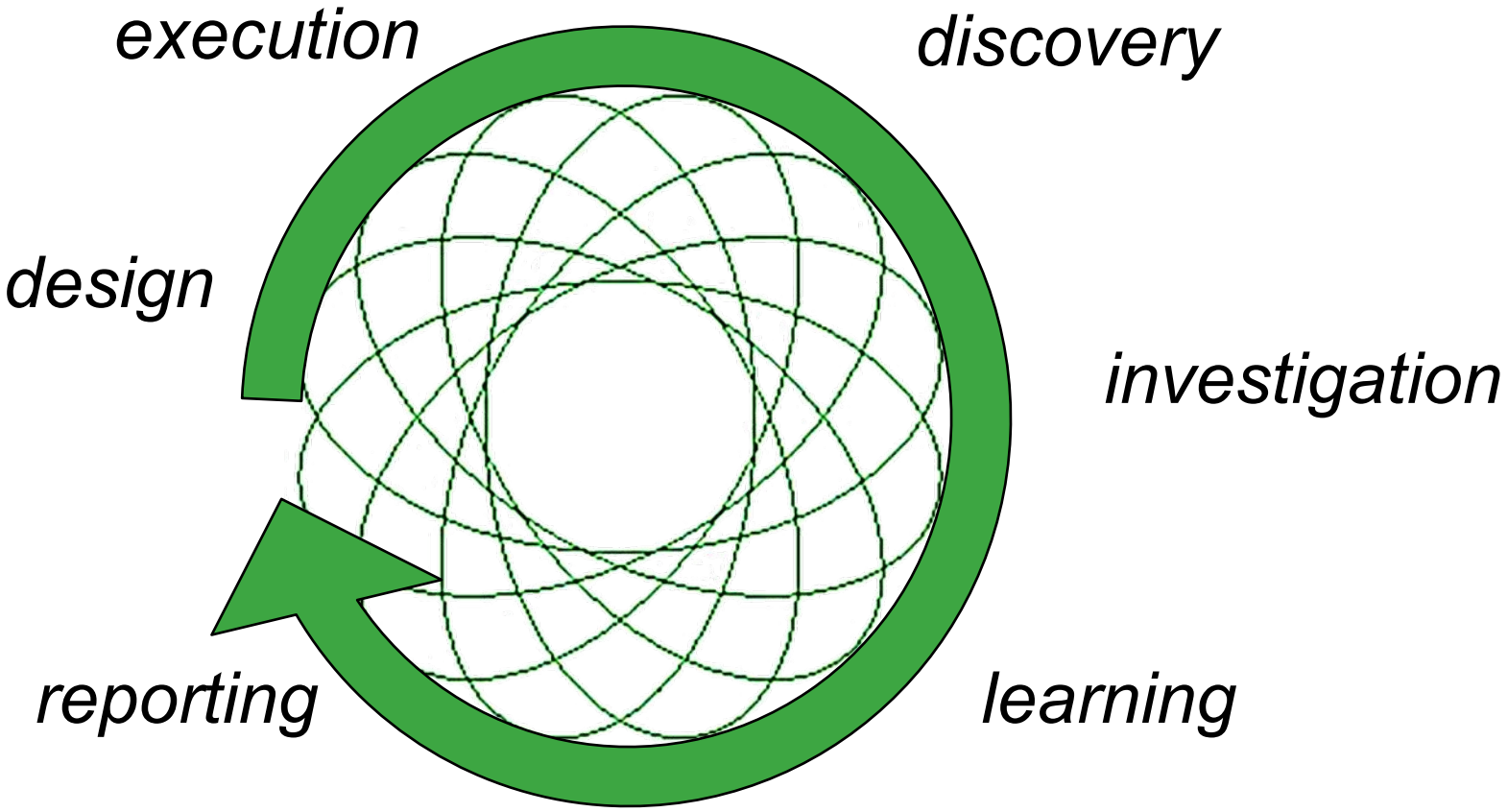


You can put them together!

arrows and cycles



Exploration: searching for value and risk



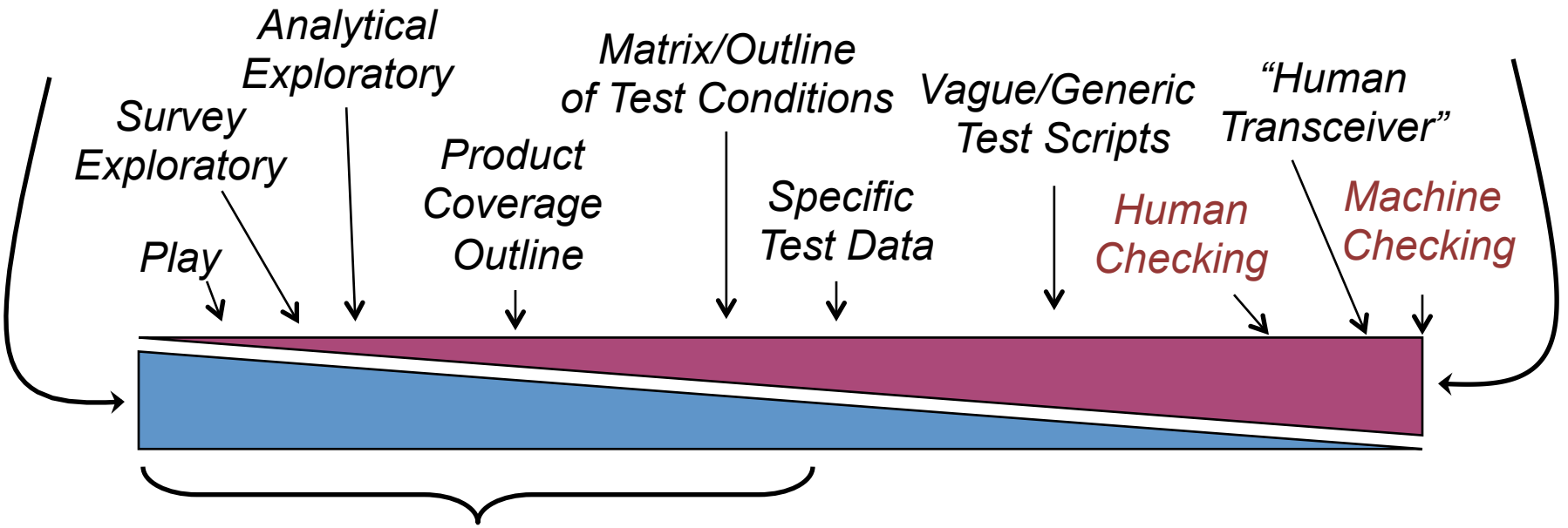
The Testing Formality Continuum
Mixing Scripting and Exploration

INFORMAL

Not done in any specific way, nor to verify specific facts.

FORMAL

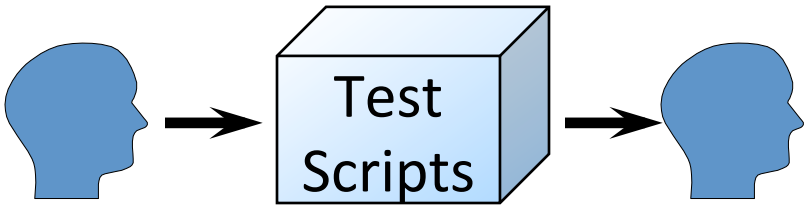
Done in a specific way, or to verify specific facts.



When I say "exploratory testing" and don't qualify it, I mean anything on the exploratory side of this continuum.

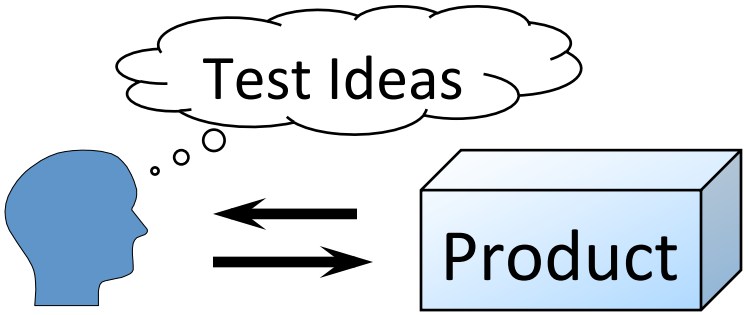
Contrasting Approaches

In *scripted* testing, tests are first designed and recorded. Then they may be executed at some later time or by a different tester.



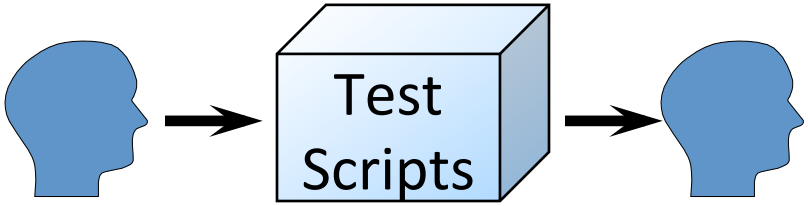
VS.

In *exploratory* testing, tests are designed and executed at the same time, and they are not necessarily recorded, but may be.



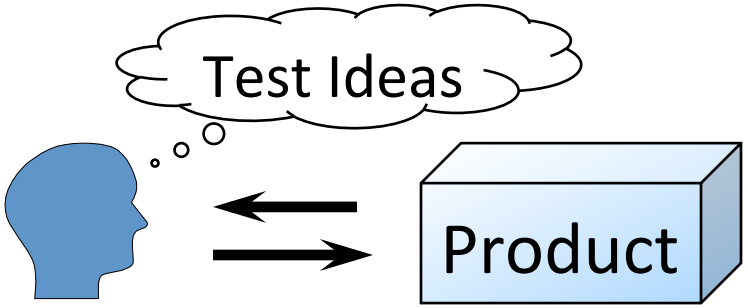
Contrasting Approaches

Scripted testing is about *controlling test execution.*



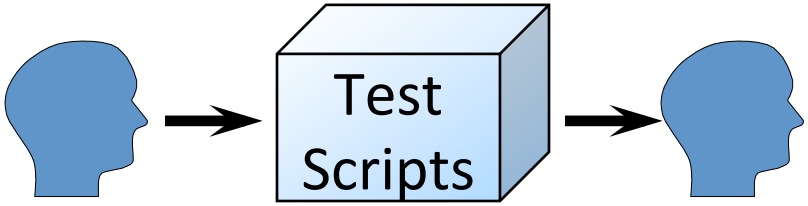
vs.

Exploratory testing is about *improving test design*.



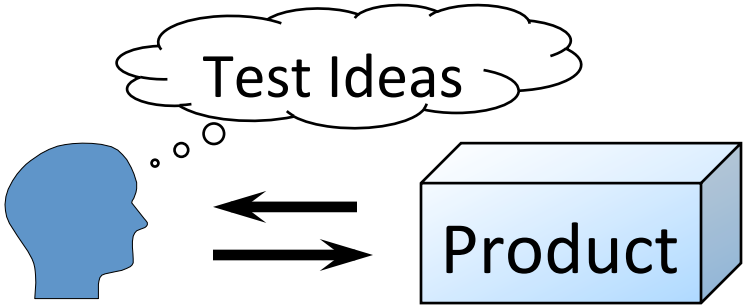
Contrasting Approaches

Scripted testing is like
making a prepared speech, or
reading from sheet music.
It is guided by pre-conceived ideas.



vs.

Exploratory testing is like
having a conversation or
playing improvisational jazz.
It is self-guided.



Blending Scripted & Exploratory

- **Generic scripts:** specify general test procedures and apply them to different parts of a test coverage outline.
- **Vague scripts:** specify a test step-by-step, but leave out any detail that does not absolutely need to be pre-specified.
- **Improvisation:** have scripts, but encourage deviation from them, too.
- **Fragmentary cases:** specify tests as single sentences or phrases.
- **Test Coverage Outline:** use outline of product elements and have tester construct tests from it on the fly.
- **Risk Catalog:** specify types of problems to look for, then construct tests on the fly to find each one.
- **Exploratory Charters:** specify 90 minutes of testing in two sentences or less.
- **Roles:** Give each tester a standing role to test a certain part of the product. Leave the rest up to them.
- **Heuristics:** Train exploratory testers to use standardized test design heuristics.
- **SBTM:** Consider using Session-Based Test Management, a formalized method of exploratory test management.

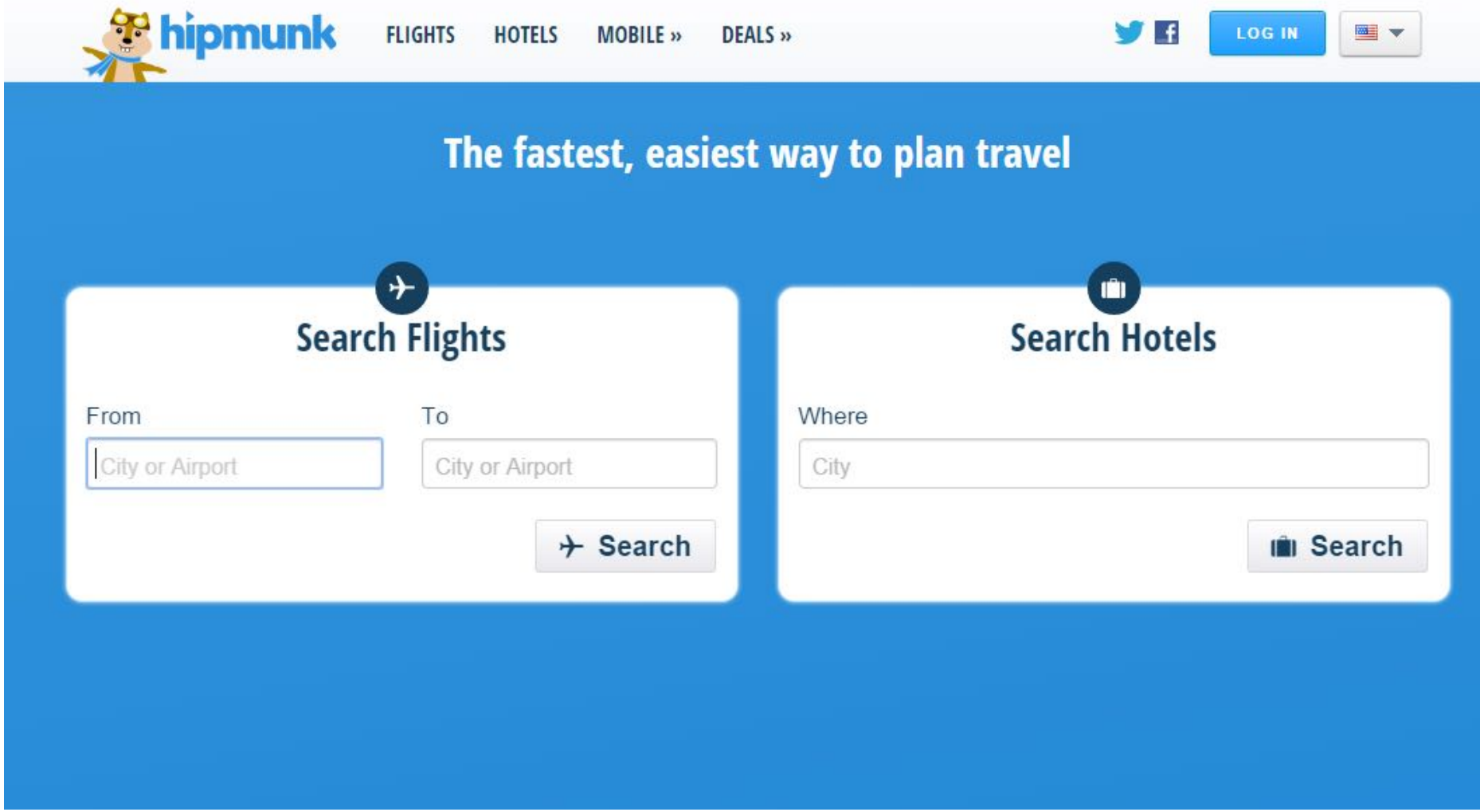
ET is a Structured Process

- Exploratory testing, as we talk about it, is a structured process conducted by a skilled tester, or by lesser skilled testers or users working under supervision.
- The structure of ET comes from many sources:
  - Test design heuristics
  - Chartering
  - Time boxing
  - Perceived product risks
  - The nature of specific tests
  - The structure of the product being tested
  - The process of learning the product
  - Development activities
  - Constraints and resources afforded by the project
  - The skills, talents, and interests of the tester
  - The overall mission of testing

In other words,
it's not "random",
but systematic.

See "Exploratory Dynamics" in the RST Appendices.

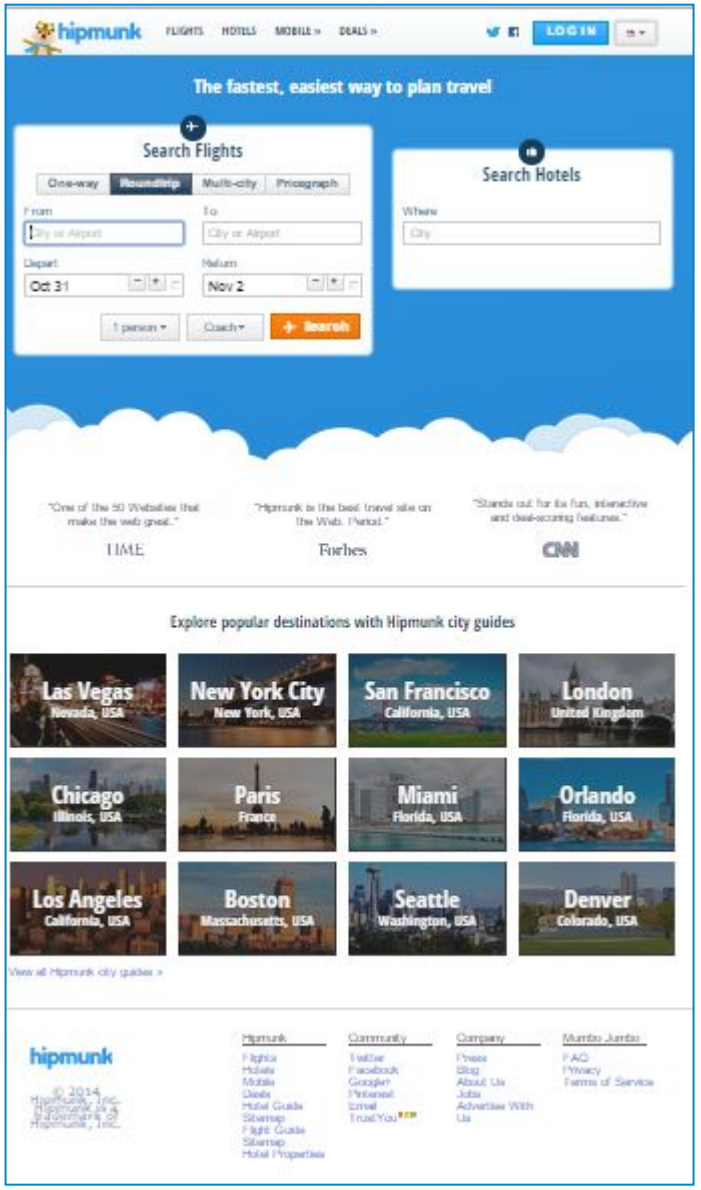
Let’s look at Hipmunk…



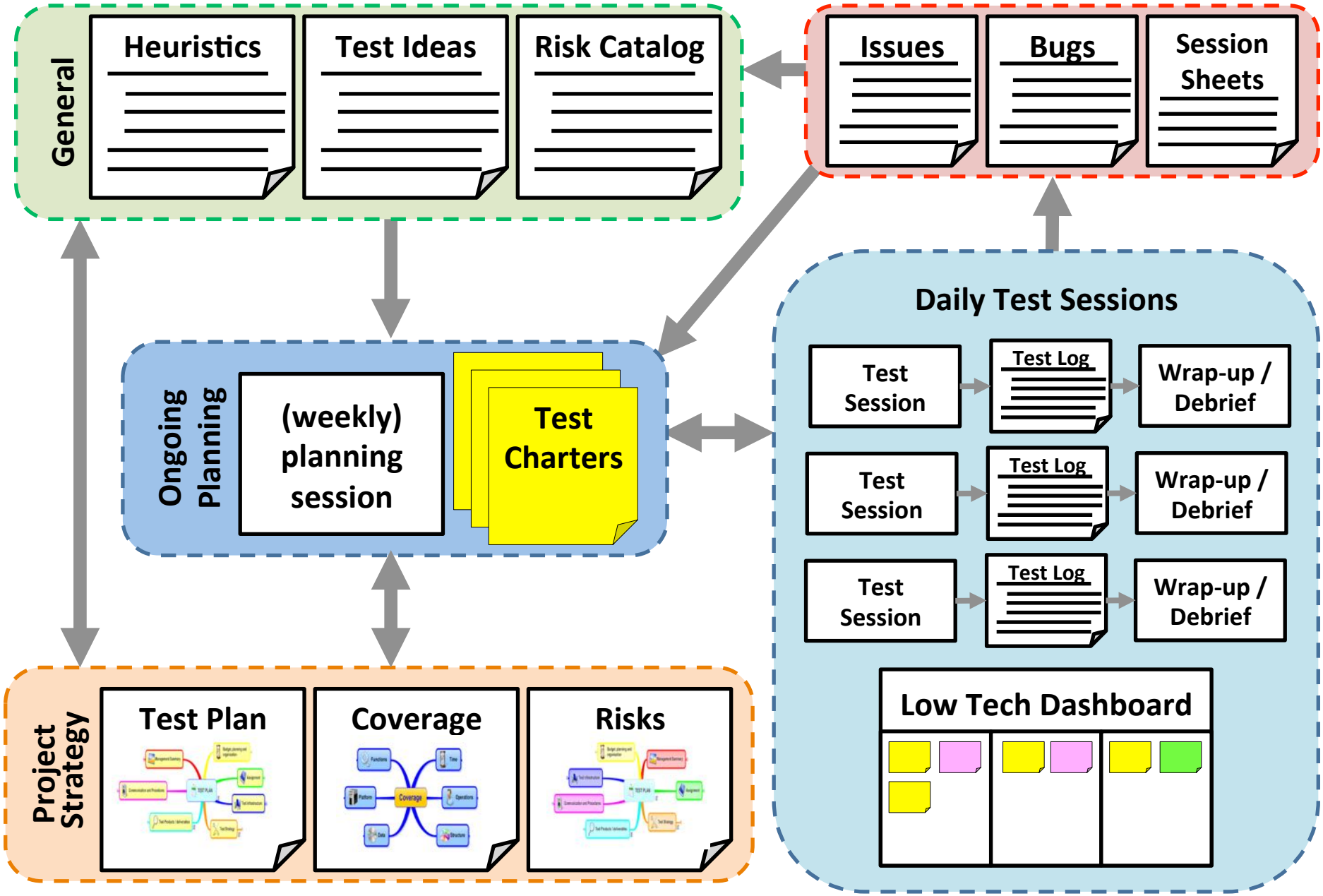
Let's look at Hipmunk…

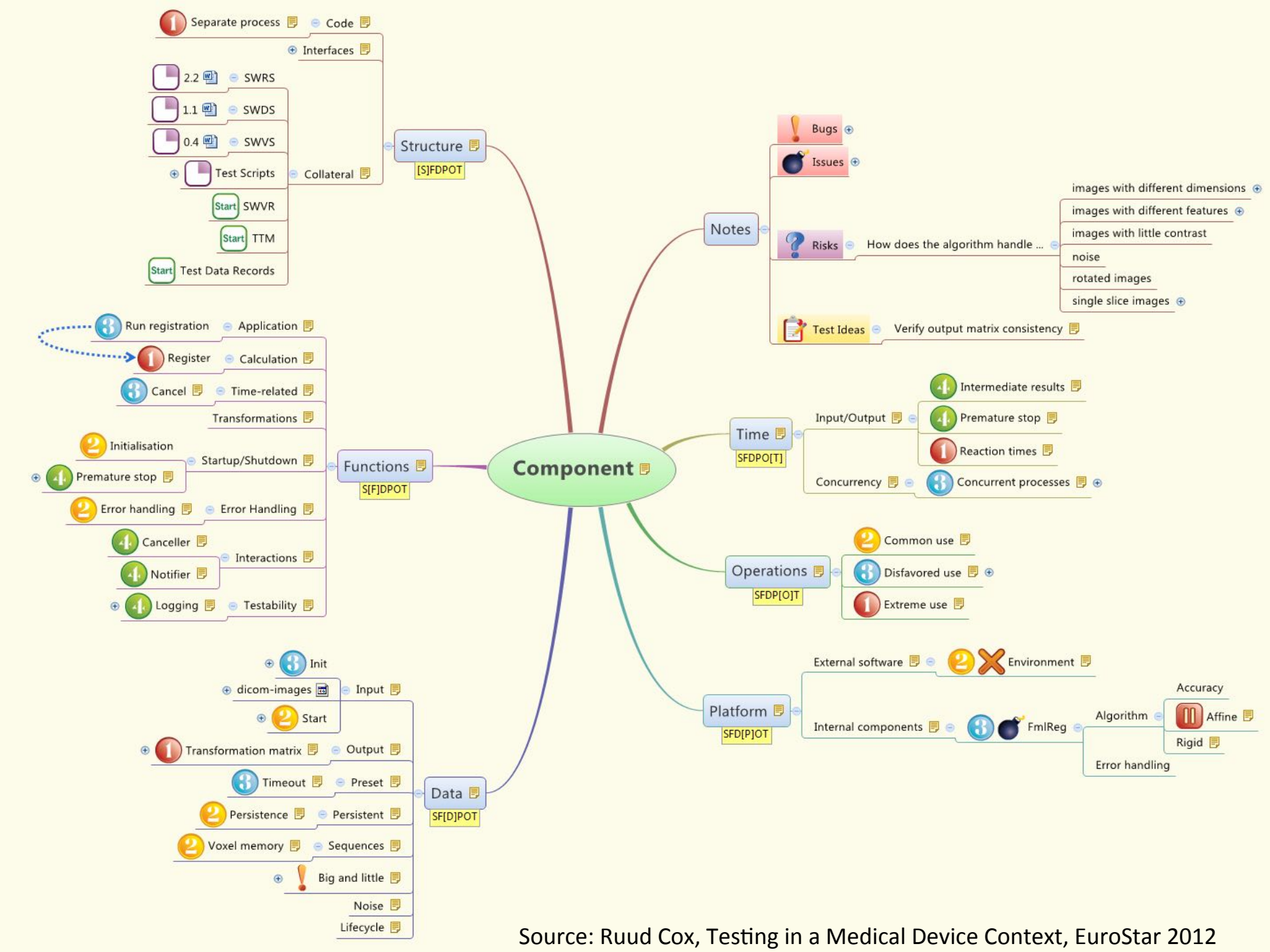
- Hipmunk is a remarkable new travel search site that aims to take the agony out of travel planning. The goal is to help you book travel faster and more efficiently.
- Hipmunk shows all relevant flight or hotel results on a single page, in a visual "timeline" that makes it easy to understand the tradeoffs 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.

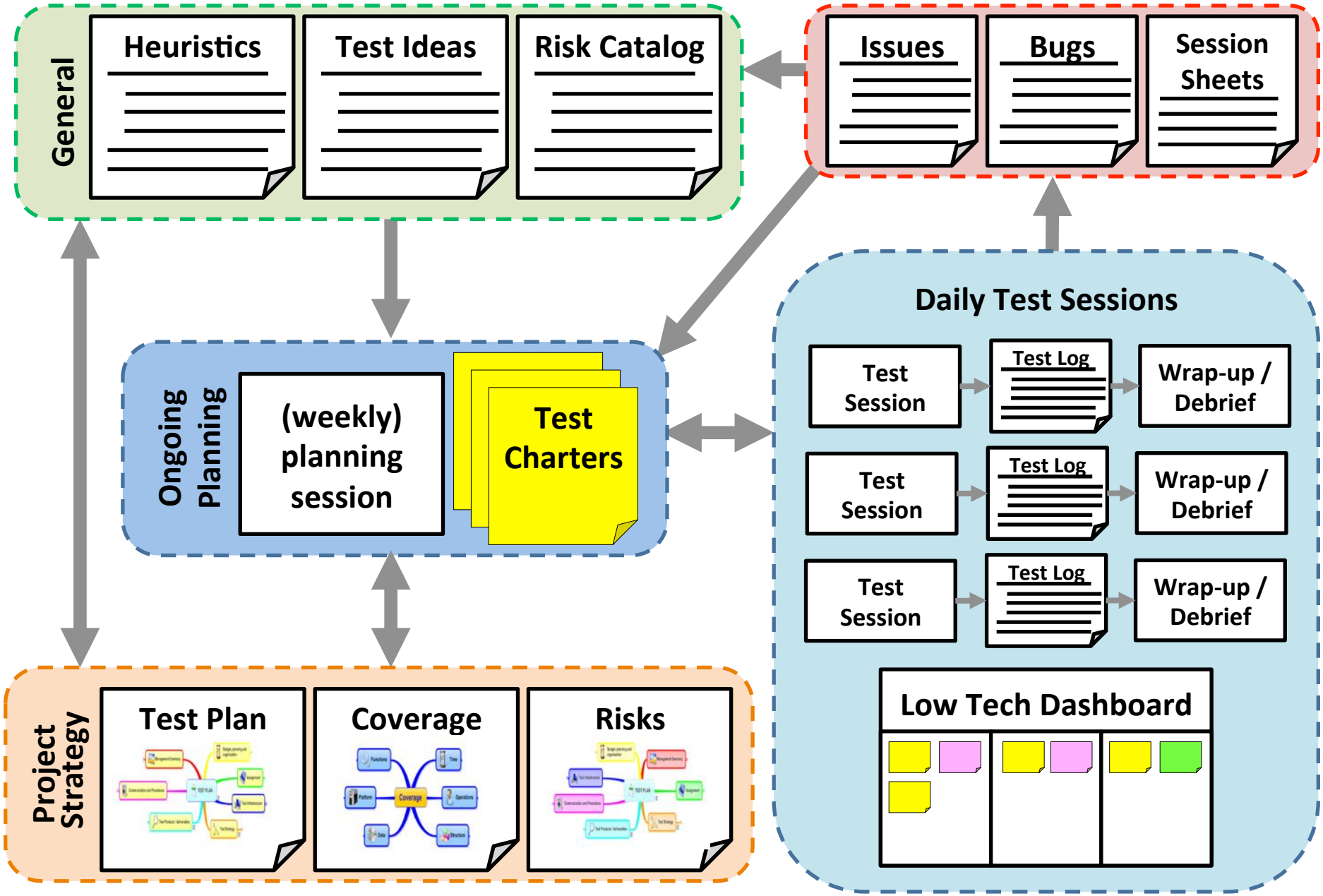


Managing Exploratory Testing


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



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)

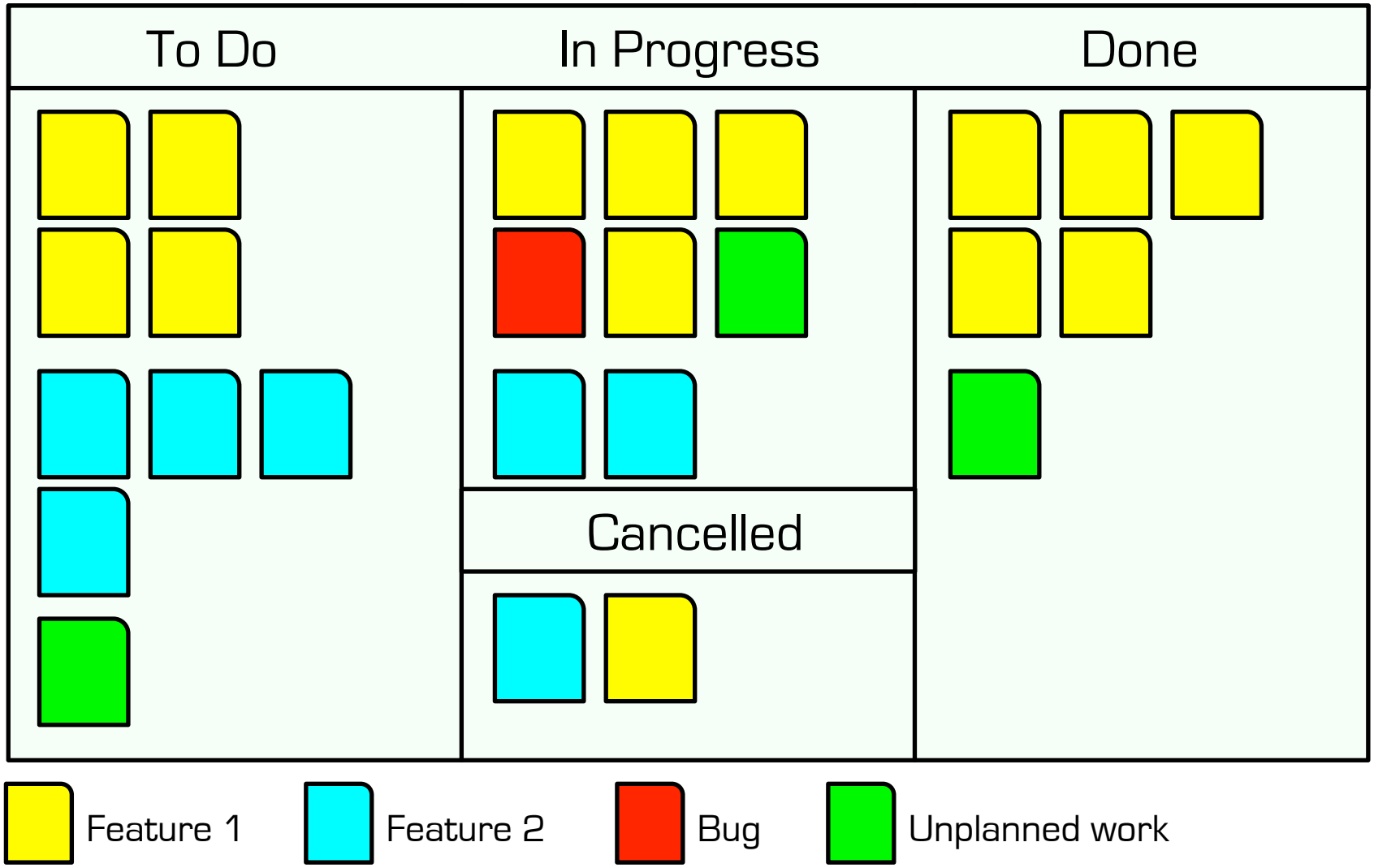
Explore Application X import menu. Identify important features with the goal of developing a coverage outline and a risk list.

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*

Dashboards


To Do
In Progress
Done
Cancelled
Feature 1
Feature 2
Bug
Unplanned work

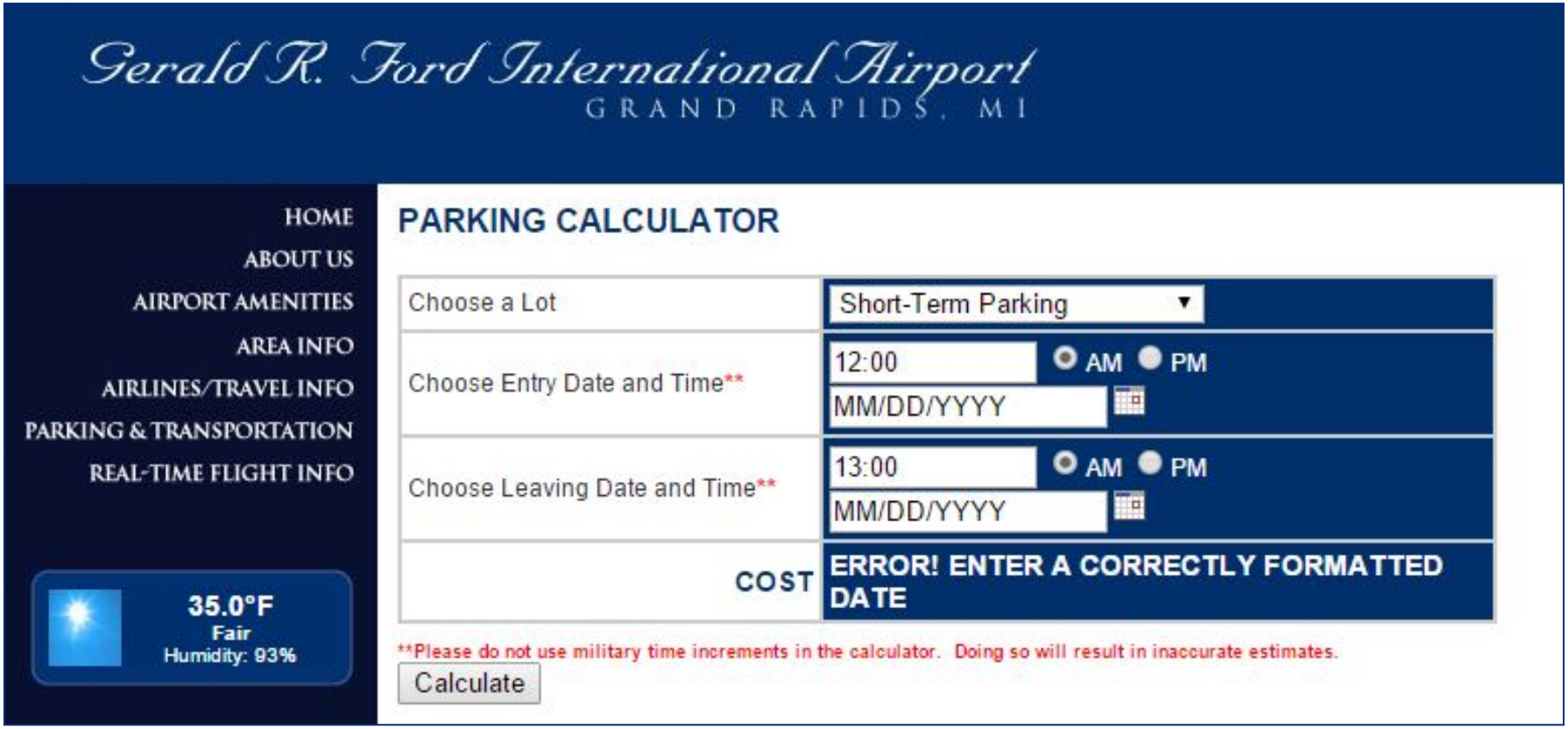
Charter Patterns: Evolving test strategy

- Intake Sessions (Goal: negotiate mission)
  "Interview the project manager about testing Xmind."
- Survey Sessions (Goal: learn product)
  "Familiarize yourself with Xmind."
- Setup Sessions (Goal: create testing infrastructure)
  "Develop a library of mindmaps for testing Xmind."
- Analysis Sessions (Goal: get ideas for deep coverage)
  "Identify the primary functions of Xmind."
  "Construct a product coverage outline."
  "Brainstorm test ideas."
  "Prepare a state model for state-based testing."
  "Perform a component risk-analysis to guide further testing."
  "Discover all the error messages in Xmind."

Charter Patterns: Evolving test strategy

- Deep Coverage Sessions (Goal: find the right bugs)
  - "Perform scenario testing based on the scenario playbook."
  - "Perform a tour that achieves double-transition state coverage."
  - "Perform steeplechase boundary testing on the major data items."
  - "Test each error message in Xmind."
  - "Perform a function tour using the 2300 node mindmap."
- Closure Sessions (Goal: get ready to release)
  - "Verify the latest fixes."
  - "Re-test tutorial with the latest build."
  - "Review help files and readme."
  - "Go over deferred bugs with Customer Support people."
  - "Perform clean-machine install test."

Parking Calculator



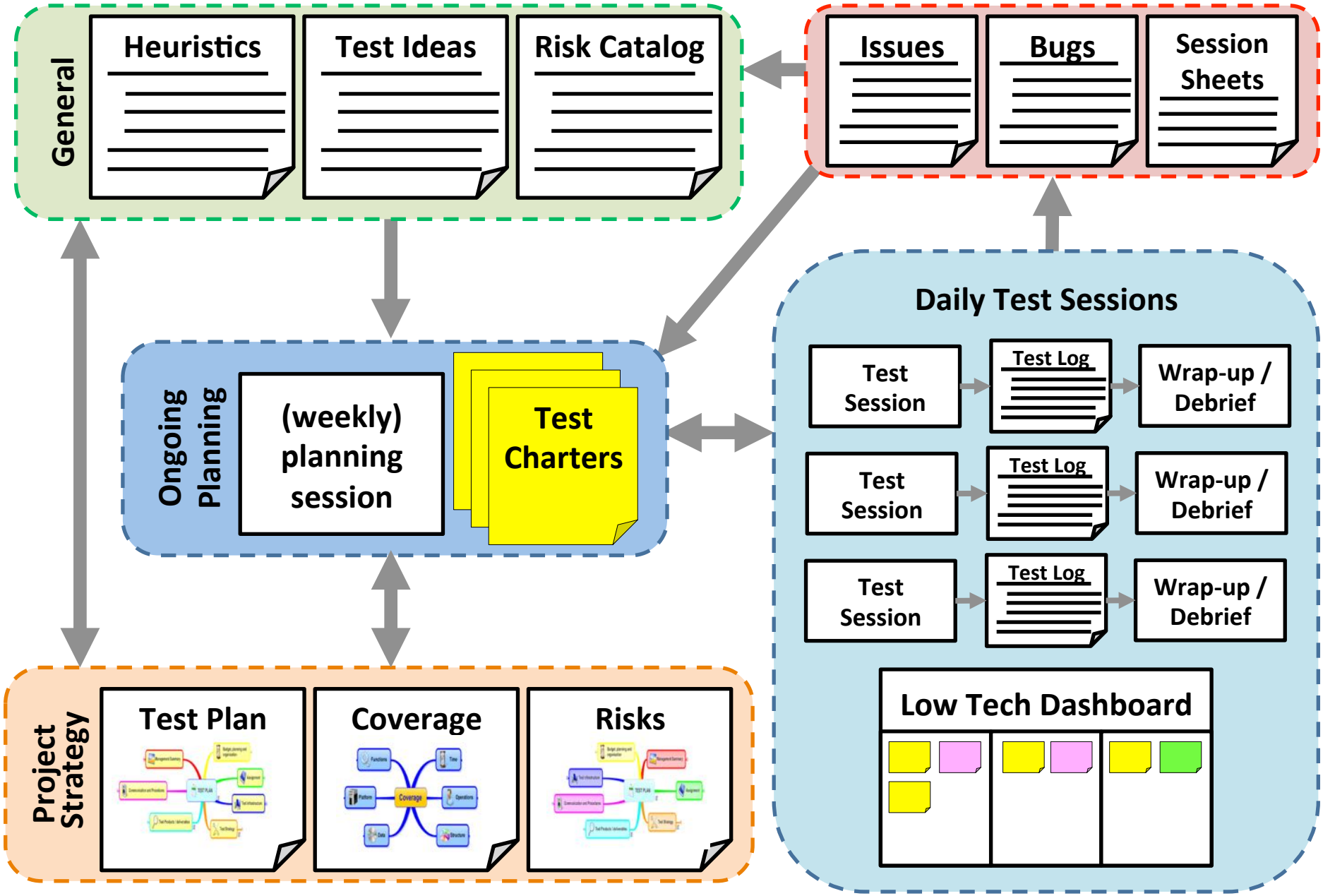
Parking Calculator

- This parking calculator is provided for the convenience of travelers to estimate parking costs. Actual parking costs may vary depending upon exact times of arrival and departure. An additional Lost Ticket fee will be assessed if the original parking ticket cannot be produced when exiting the parking facilities.

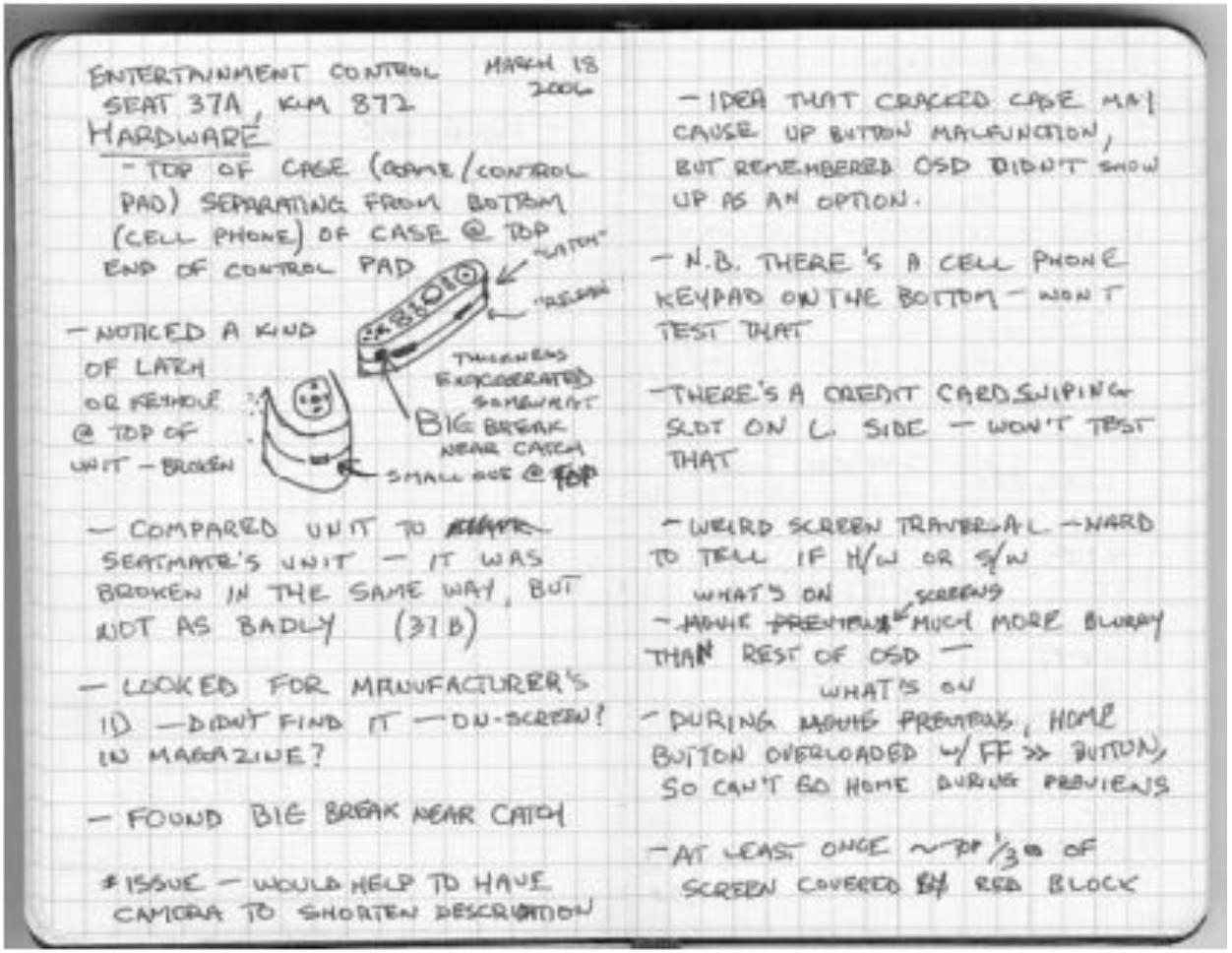
- Original website: http://www.grr.org/ParkCalc.php
- Older more buggy version: http://adam.goucher.ca/parkcalc/

Analysis & deep coverage session

- Step 1: do a quick analysis session of "ParkCalc" in exactly five minutes to learn about the app and brainstorm on what could be tested in the product. Prepare for a deep coverage session.
- Step 2: test a part of application for 15 min. Keep track of your ideas and actions as you go.


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

Notes



ENTERTAINMENT CONTROL SEAT 37A, KM 872 — MARCH 18 2006

HARDWARE

- TOP OF CASE (GAME/CONTROL PAD) SEPARATING FROM BOTTOM (CELL PHONE) OF CASE @ TOP END OF CONTROL PAD

- NOTICED A KIND OF LATCH OR REMOVE @ TOP OF UNIT – BROKEN

- COMPARED UNIT TO SEATMATE'S UNIT – IT WAS BROKEN IN THE SAME WAY, BUT NOT AS BADLY (37B)

- LOOKED FOR MANUFACTURER'S ID – DIDN'T FIND IT – ON-SCREEN? IN MAGAZINE?

- FOUND BIG BREAK NEAR CATCH

* ISSUE – WOULD HELP TO HAVE CAMERA TO SHORTEN DESCRIPTION

- IDEA THAT CRACKED CASE MAY CAUSE UP BUTTON MALFUNCTION, BUT REMEMBERED OSD DIDN'T SHOW UP AS AN OPTION.

- N.B. THERE'S A CELL PHONE KEYPAD ON THE BOTTOM – WON'T TEST THAT

- THERE'S A CREDIT CARD SWIPING SLOT ON L. SIDE – WON'T TEST THAT

- WEIRD SCREEN TRANSITIONS – HARD TO TELL IF H/W OR S/W

WHAT'S ON SCREENS

- MOVIE PREVIEWS MUCH MORE BLURRY THAN REST OF OSD –

WHAT'S ON

- DURING MOVIE PREVIEWS, HOME BUTTON OVERLOADED W/ FF >> BUTTON, SO CAN'T GO HOME DURING PREVIEWS

- AT LEAST ONCE ~TOP 1/3 OF SCREEN COVERED BY RED BLOCK

Can also be a tool like Rapid Reporter, notepad or a Mind Map

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)

Debrief - PROOF

Past: What happened during the session?

Results: What was achieved during the session?

Outlook: What still needs to be done?

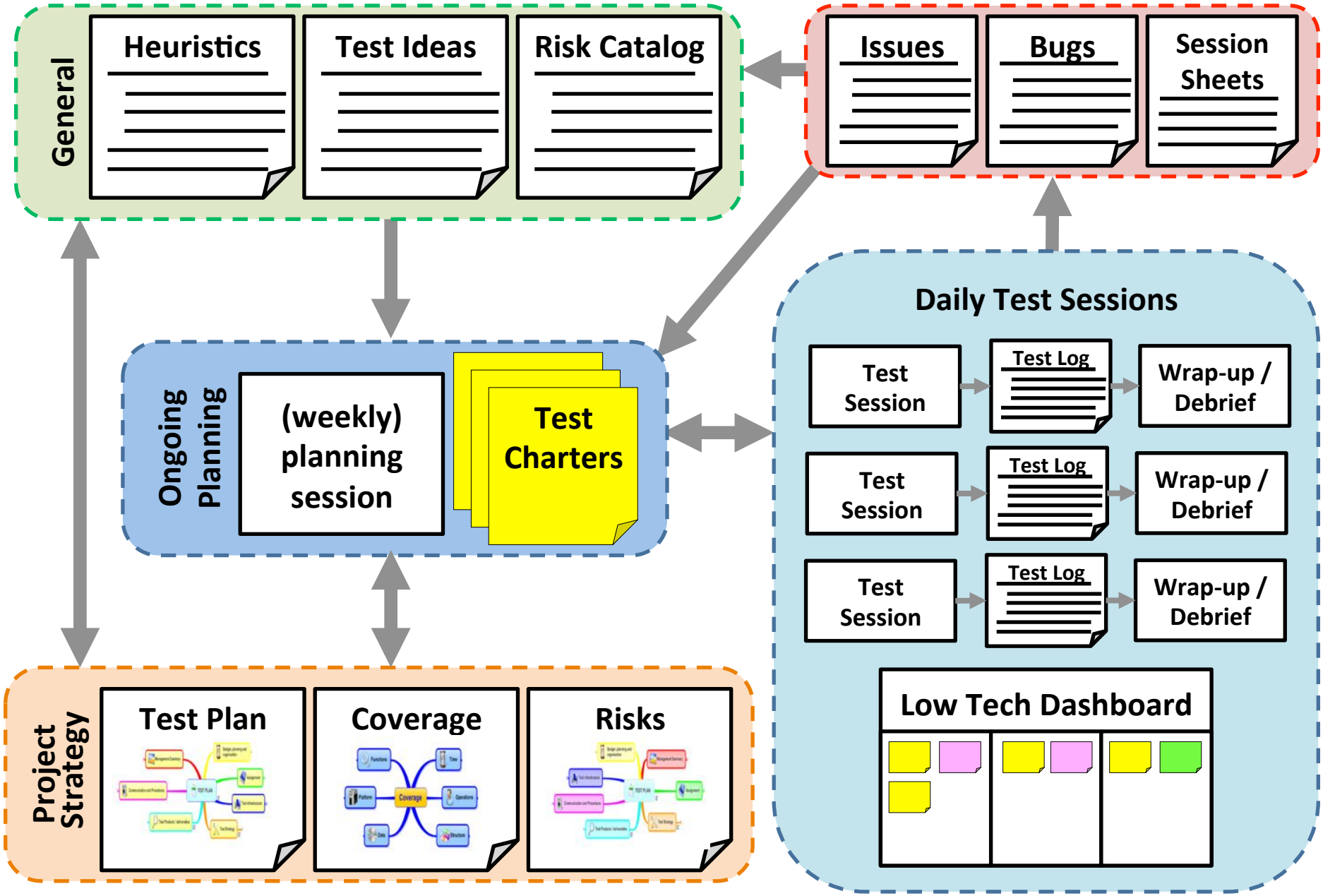
Obstacles: What got in the way of good testing?

Feelings: How does the tester feel about all this?

Discuss session sheet to assure that both understand it.

Discuss session and ask questions: new sessions may be chartered.

Learn!


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

Why ET is powerful

- Do what really needs to be done
- It facilitates experimentation, serendipity, discovery and learning
- It finds more bugs
- Create engagement: help people use their brains
- Take advantage of tacit knowledge and skill
- Use insights from experiments to inform the next
- Using the full creative power of exploration

Mastering Exploratory Testing

Challenges?

Exploratory Testing =

(like testing in general) is 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
- Debrief & retrospect
- Testing Dojo
- Train creating test ideas fast
- Learn more about observation, experiments, biases, social science, ...

"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

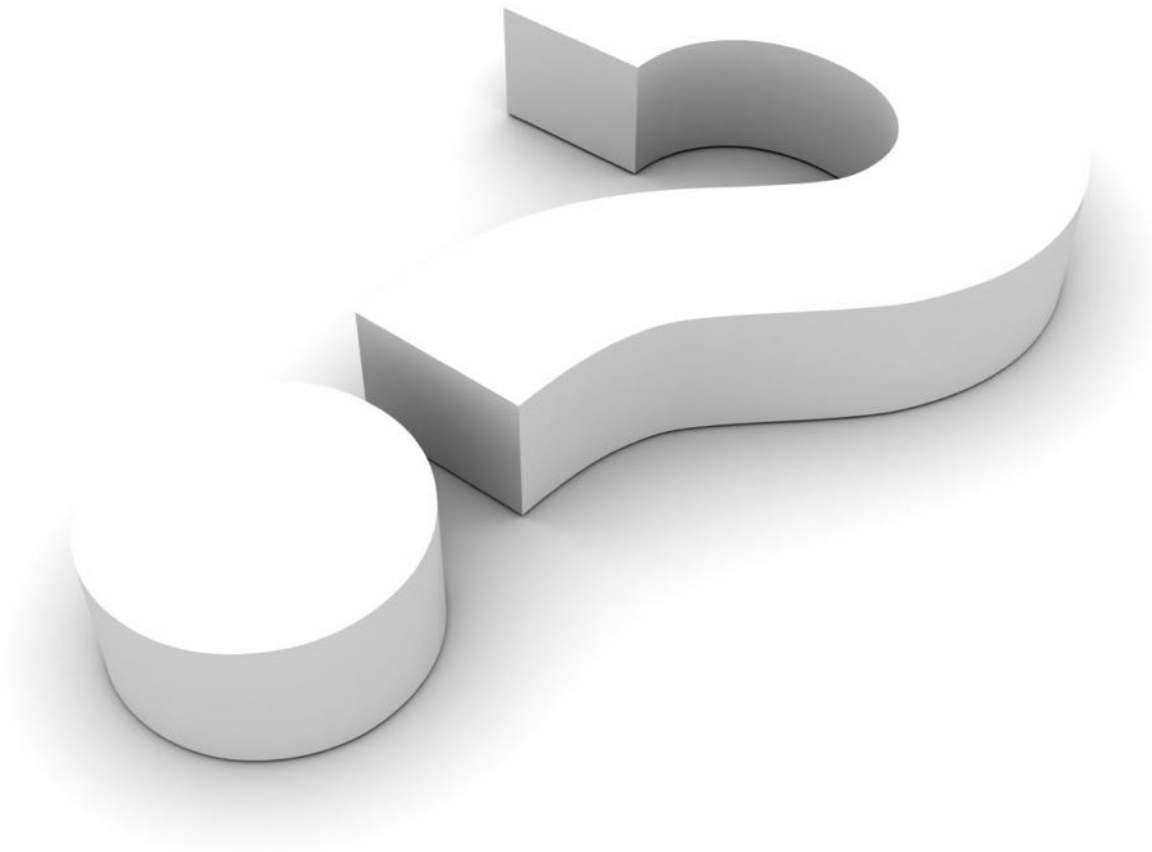
Image: http://corkboard.me


The Pragmatic Programmers
Explore It!
Reduce Risk and Increase Confidence with Exploratory Testing
Elisabeth Hendrickson
Edited by Jacquelyn Carter


EXPLORATORY
SOFTWARE
TESTING

Value

Vragen, opmerkingen, discussie, 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

- Some of these slides are taken from Rapid Software Testing by James Bach & Michael Bolton
  http://www.satisfice.com/info_rst.shtml
- RST Appendices
  http://www.satisfice.com/rst-appendices.pdf
- Keri Smith
  http://www.kerismith.com
  http://www.flickr.com/photos/kerismith/
  http://redefineschool.com/keri-smith-explore-art/
  http://www.kerismith.com/popular-posts/100-ideas/
  http://superforest.org/2010/03/the-rebels-manifesto-keri-smith/
  http://www.kerismith.com/blog/revolution/
- “Evolving Understanding of Exploratory Testing” en “Structures of Exploratory Testing”
  http://www.developsense.com/resources.html
- Test cases are not testing: towards a culture of test performance by James Bach & Aaron Hodder
  http://www.testingcircus.com/testing-trapeze-2014-february-edition/
- Verzamelde links op mijn website:
  http://www.huibschoots.nl/links (take a look at the social science section too)
- Artikel “Exploratief testen gedefinieerd, hardnekkige mythes ontkracht!”
  http://improveqs.nl/files/Exploratory_Testen_-_Huib_Schoots_-_Oktober_2013.pdf
- A lesson in exploratory testing
  http://trishkhoo.com/2012/10/a-lesson-in-exploratory-testing/