

TestNet – Exploratory Testing

Boost your testing power with exploration

Pascal Dufour & Huib Schoots, Oktober 2013

Acknowledgements

- Some of these slides are taken from the Rapid Software Testing course created by James Bach and Michael Bolton.
www.satisfice.com
www.developsense.com
- The test machine exercise is made by James Lyndsay.
www.workroom-productions.com
- Many of the ideas in this presentation were also inspired by or augmented by other colleagues including Jon Bach, James Lyndsay, Elisabeth Hendrickson, Markus Gärtner, Sigge Birgisson, Henke Anderson, Paul Holland, Rikard Edgren, Trish Khoo, James Bach and Michael Bolton.

Agenda TestNet – 31 oktober 2013

- Warming up oefening: Test Machine
- Korte presentatie Exploratory Testing & Session Based Test Management
- Oefening Sessions & Charters
- Test ronde 1 & debrief Test ronde 1
- Test ronde 2 & debrief Test ronde 2
- Test ronde 3 & debrief Test ronde 3
- Wrapup
- Q & A

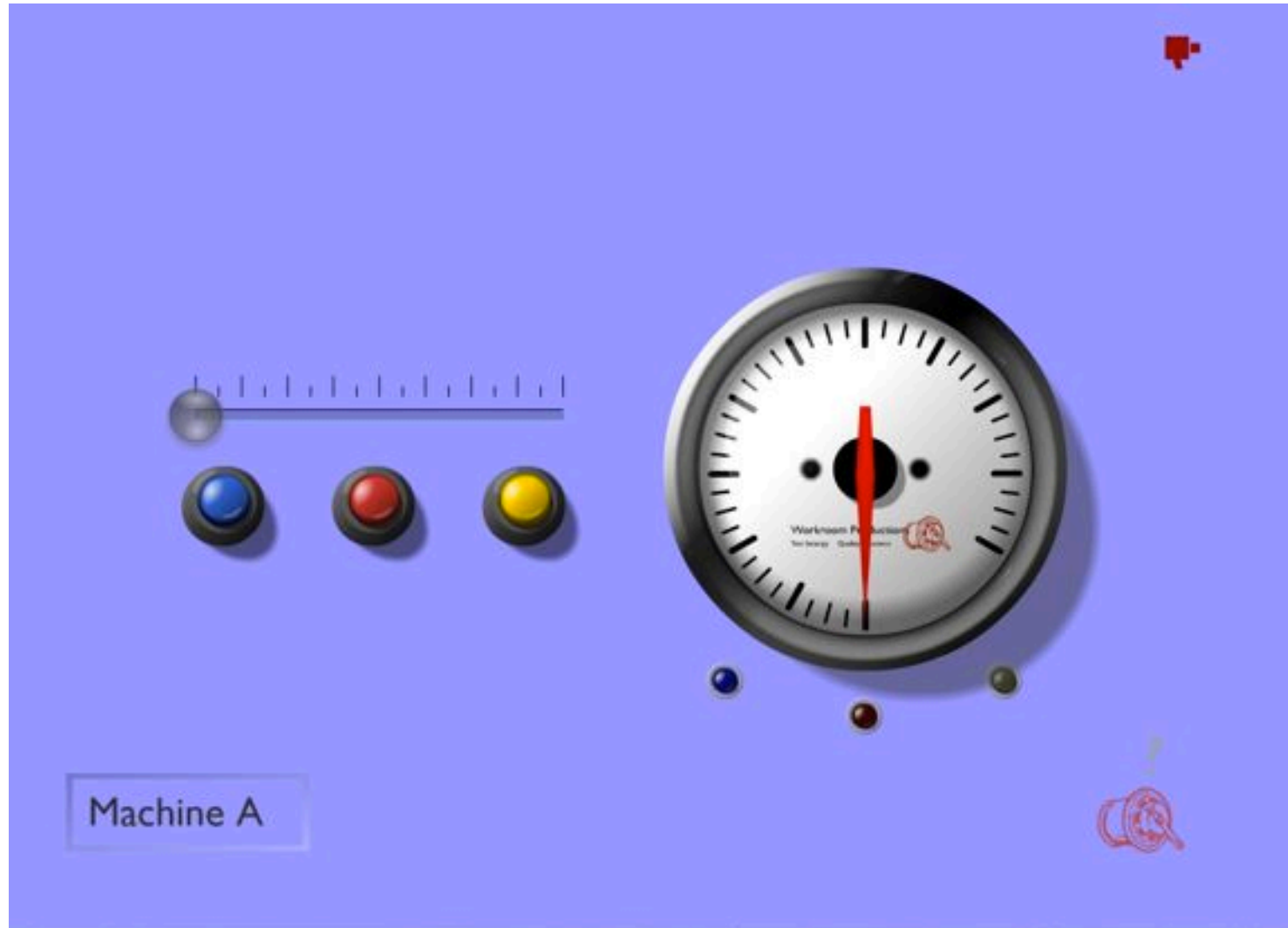
Uiteraard met zoveel mogelijk ruimte tussendoor voor vragen en discussie

Check

Is dit wat jullie willen leren?



Exercise Test Machine



Exploratory Testing & Session Based Test Management

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

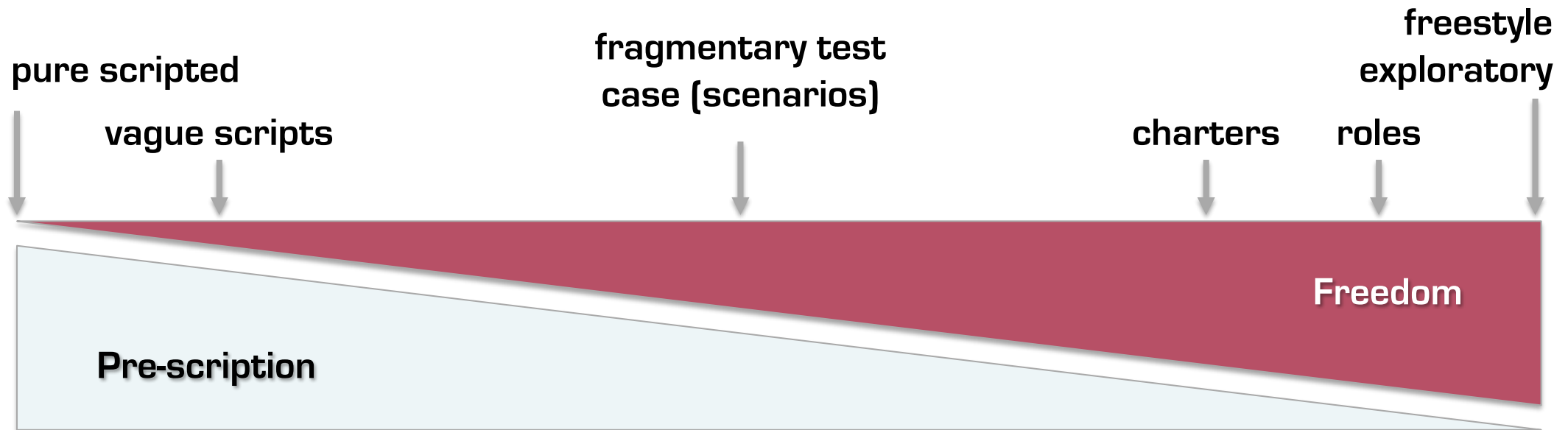
“An approach to software testing that emphasizes the personal freedom and responsibility of each tester to continually optimize the value of his work by treating learning, test design and test execution as mutually supportive activities that run in parallel throughout the project.”



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

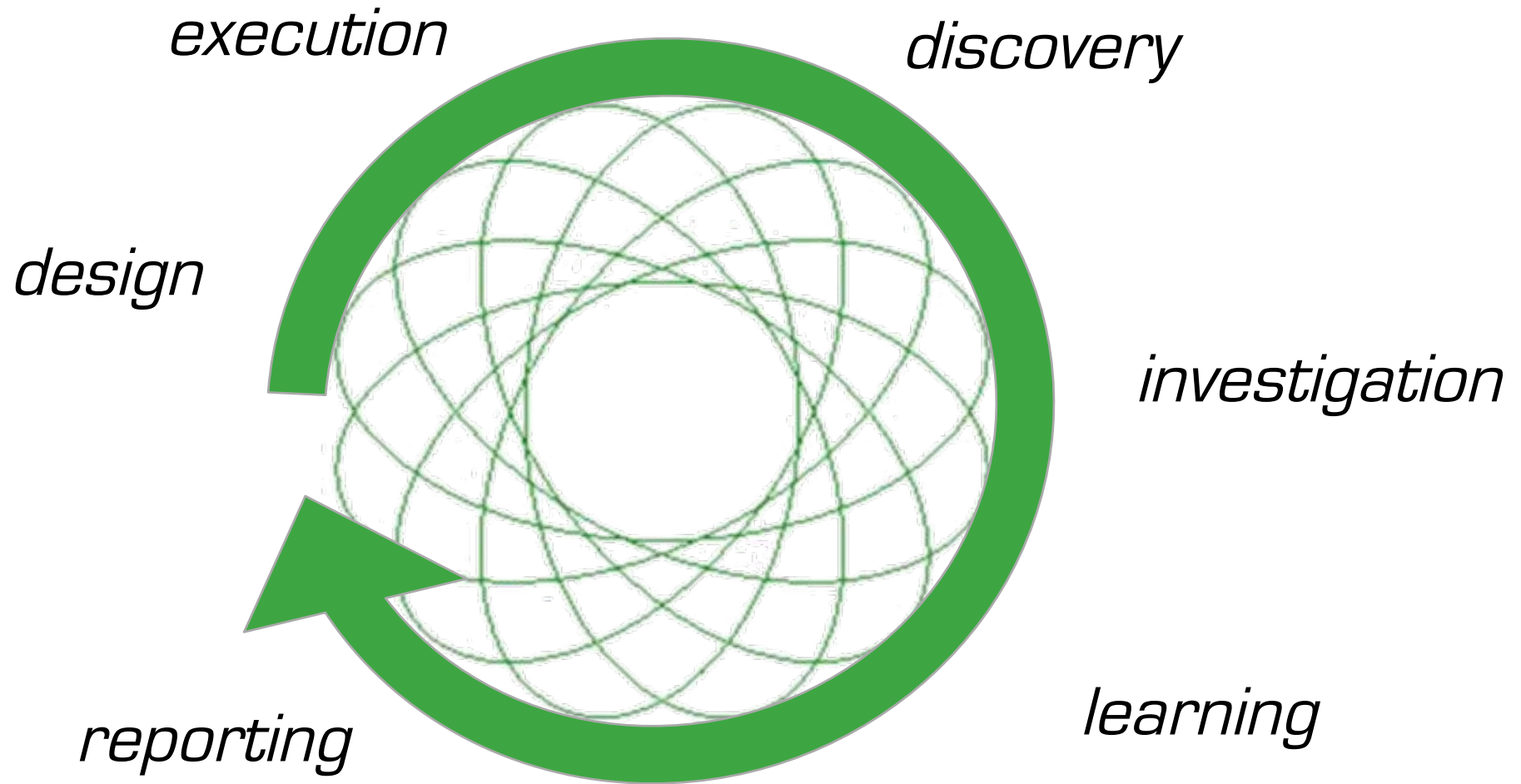


Exploratory Testing



Source: Telling Your Exploratory Story by Jon Bach

Exploration: searching for value and risk



Source original picture: Two futures of testing by Michael Bolton

Questions About Exploration...

arrows and cycles

Where does
exploration
come from?

What happens when
the unexpected
happens during
exploration?

What do we do
with what we
learn?

Will everyone
explore the same way?

Source: Rapid Software Testing by James Bach & Michael Bolton

This slide is used with kind permission

Questions About Scripts...

arrows and cycles

What happens when the unexpected happens during a script?



Source: Rapid Software Testing by James Bach & Michael Bolton

This slide is used with kind permission

Answers 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?

Source: Rapid Software Testing by James Bach & Michael Bolton

This slide is used with kind permission

You can put them together!

arrows and cycles



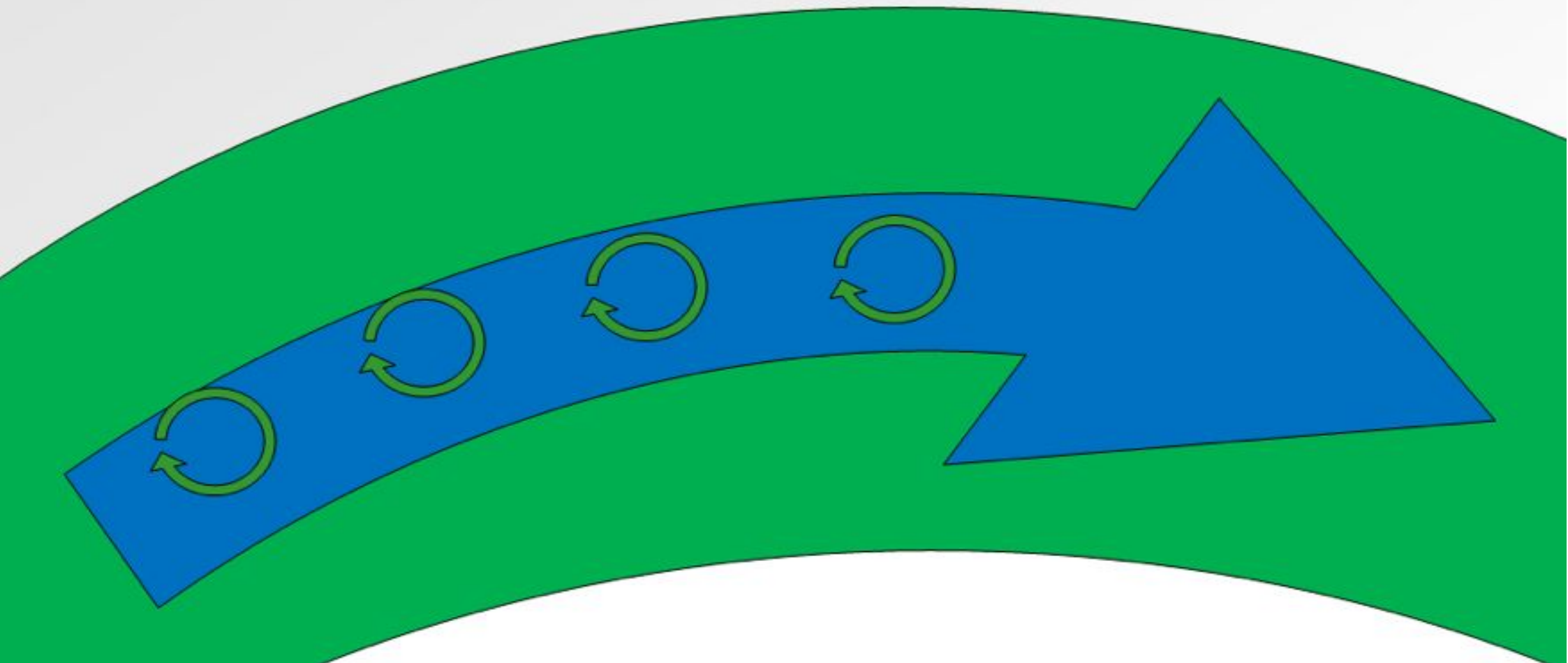
SCRIPTED
(task performing)

Source: Rapid Software Testing by James Bach & Michael Bolton

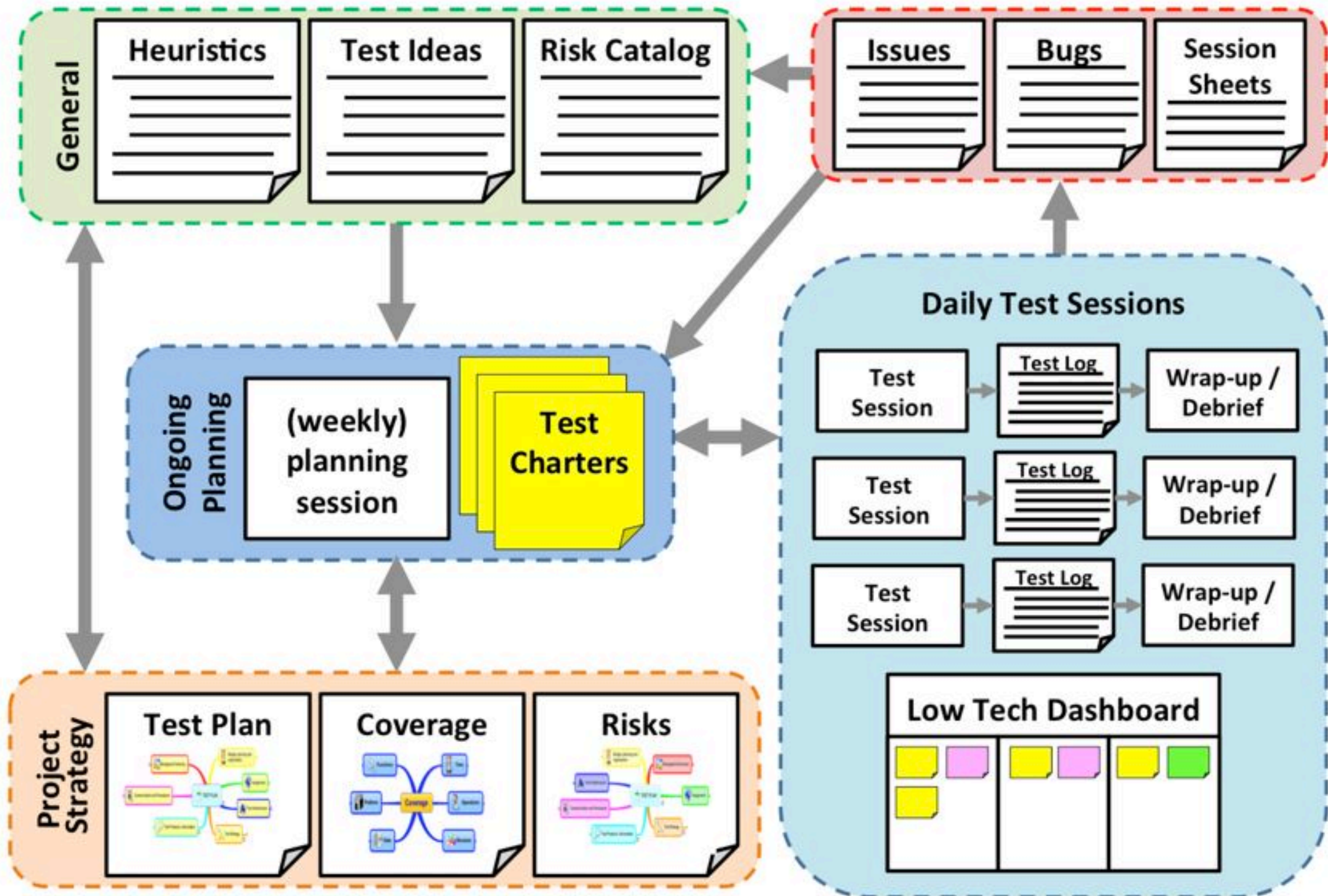
This slide is used with kind permission

You can put them together!

arrows and cycles



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



NS Reisplanner

How does the product interact with time? Any relationship between the product and time.

Time

- overgang naar zomer/winter tijd
- datumovergang
- verstrijken van tijd, wat doet het met reisadvies?

What does the product do? Everything that the product does.

Functions

- Test all stations
- Test several addresses
- Test several postcodes
- Andere locaties
- Via
- Datum selecteren
- Prijsberekening
 - Retour Zonder korting
 - Enkele reis Met korting
 - Retour Met korting
 - enkele reis Zonder korting
 - 2e klas
 - 1e klas
- Reisschema
 - Sporen details
 - Check: vertrek/aankomst/aankomst/reistijd Correct?
 - Alle opties opgenomen? Volledig?
- Koop kaartje

How the product will be used / How do people actually use the program?

Operations

- Millions of users
- 24/7 available

Everything on which the product depends (and that is outside your project).

Platform

- Geen specifieke tests

What are the pieces and how do they fit together? Everything that comprises the physical product

Structure

- Dropbox
 - Selecteren met muis
 - TAB en pijltjes toetsen
- Tekstbox
 - Oneindig veel karakters
 - Diakrieten
 - Zoeken op halve invoer UTRE --> alle stations die beginnen met met UTRE...
- Radiobutton
 - Selecteren met muis
 - TAB en pijltjes toetsen
- Zoekfunctie (TOM TOM)
- Datum zoeker
 - Door de weekse dag
 - Weekenddag
 - Dag in vakantierooster
 - 29/2 Schrikkel jaar
 - 25/12 1e kerstdag
 - Blegal
 - 30/2
 - 29/2 buiten schrikkeljaar
 - Normale tijd
 - Ochtend
 - Middag
 - Avond
 - Speciale tijden
 - Midden in de nacht
 - 12:61
 - Ongeldig
 - 25:01
 - 100:100
- Datum veld = Dates
- Tijd veld = Tijd
- Knop
- Detail info

What does the product do things to? Everything that the product processes.

Data

- Geen specifieke tests

Plan uw reis

Van

Naar

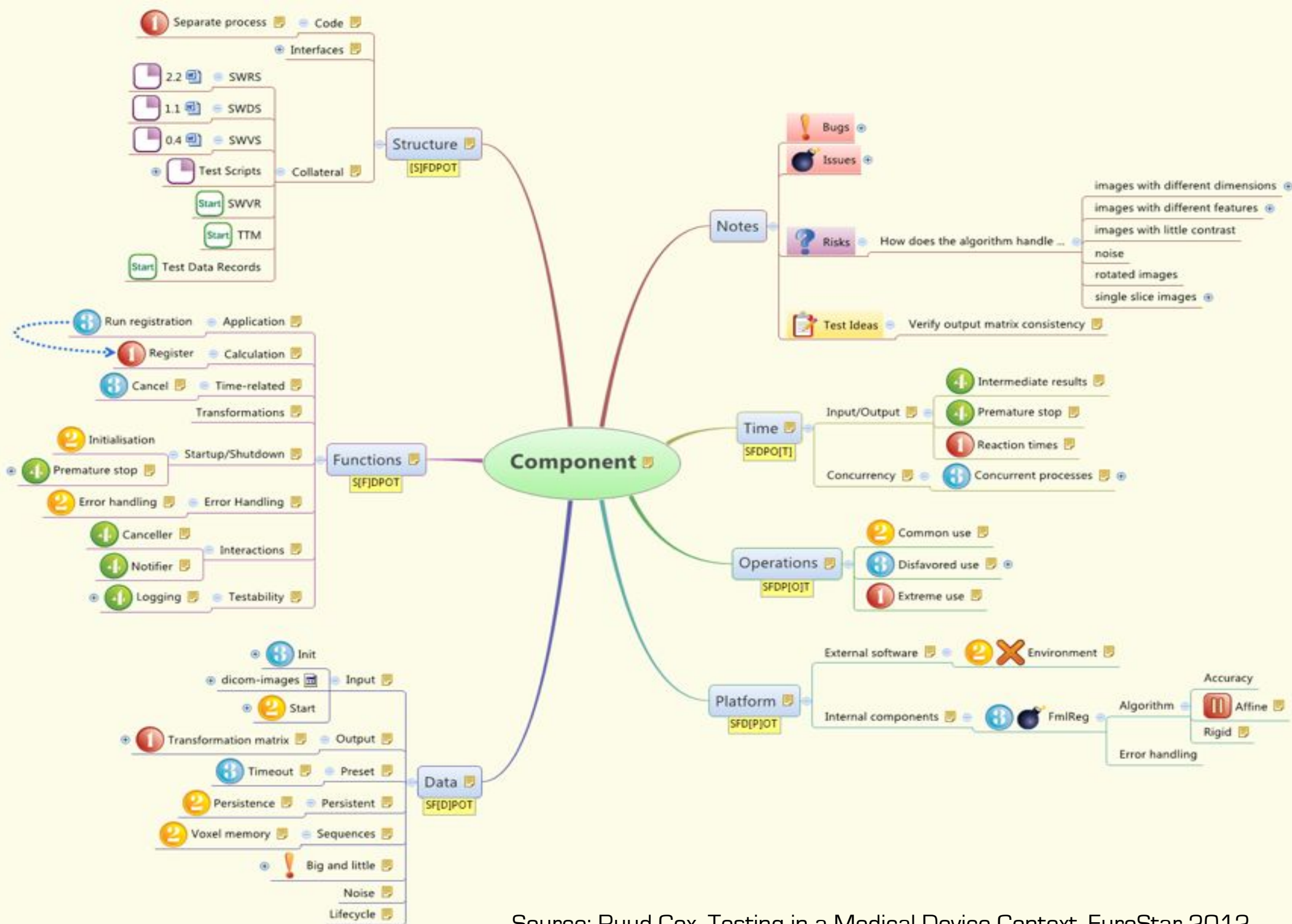
Via Station

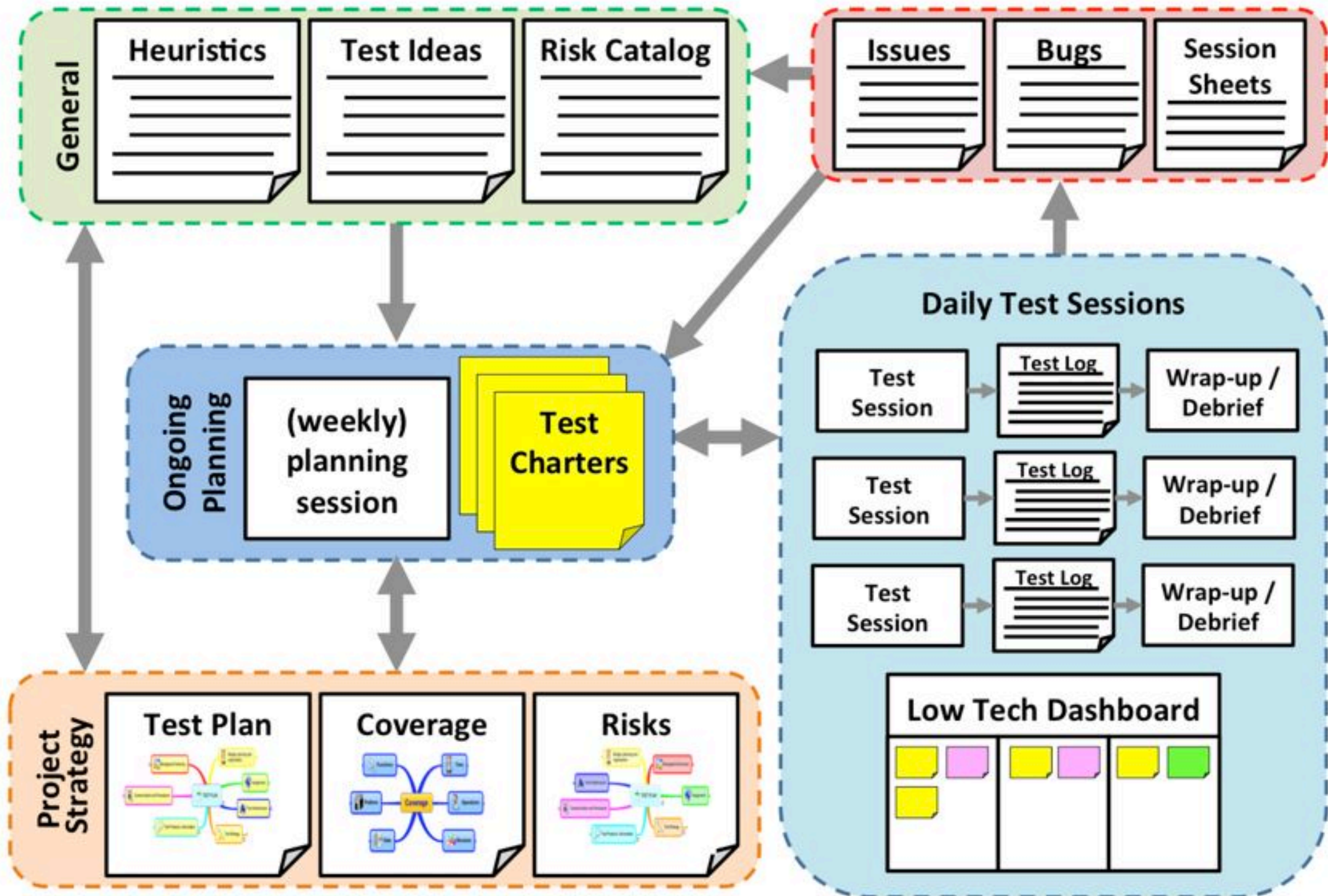
Datum dd-mm-jjjj

Tijd : Vertrek Aankomst

Geef reis en prijs →

Floating Topic



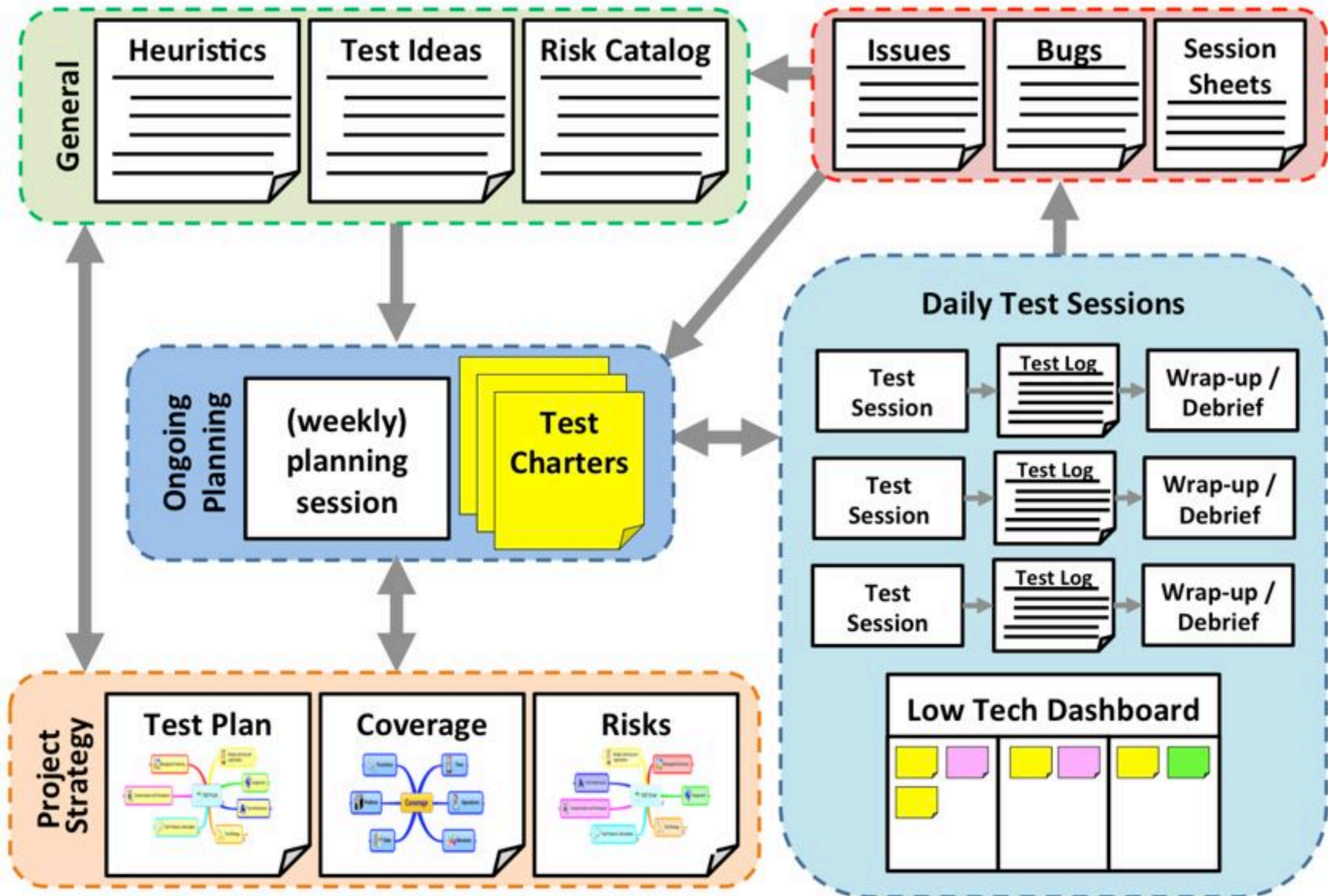


Test Charters

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

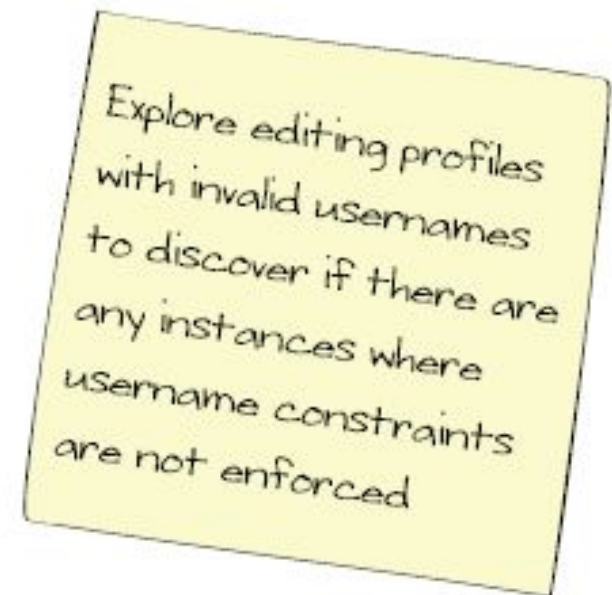
- Read Chapter 4 of the product specification. Prepare a mind map, and discuss it with Peter (programmer) and David (architect).
- Explore Application X import menu. Identify important features with the goal of developing a coverage outline and a risk list.
- My mission is to test <insert risk here> to <insert coverage here>
- Explore (target) With (resources) To discover (information)





Sessions

- The tests are run in sessions with a more or less fixed length
- Every session has a goal and one or several test strategies
- Uninterrupted testing time (no mail, calls, meetings, breaks)
- Duration is normally ≈ 90 min
- We do 2-3 sessions per day per tester



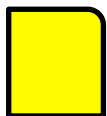
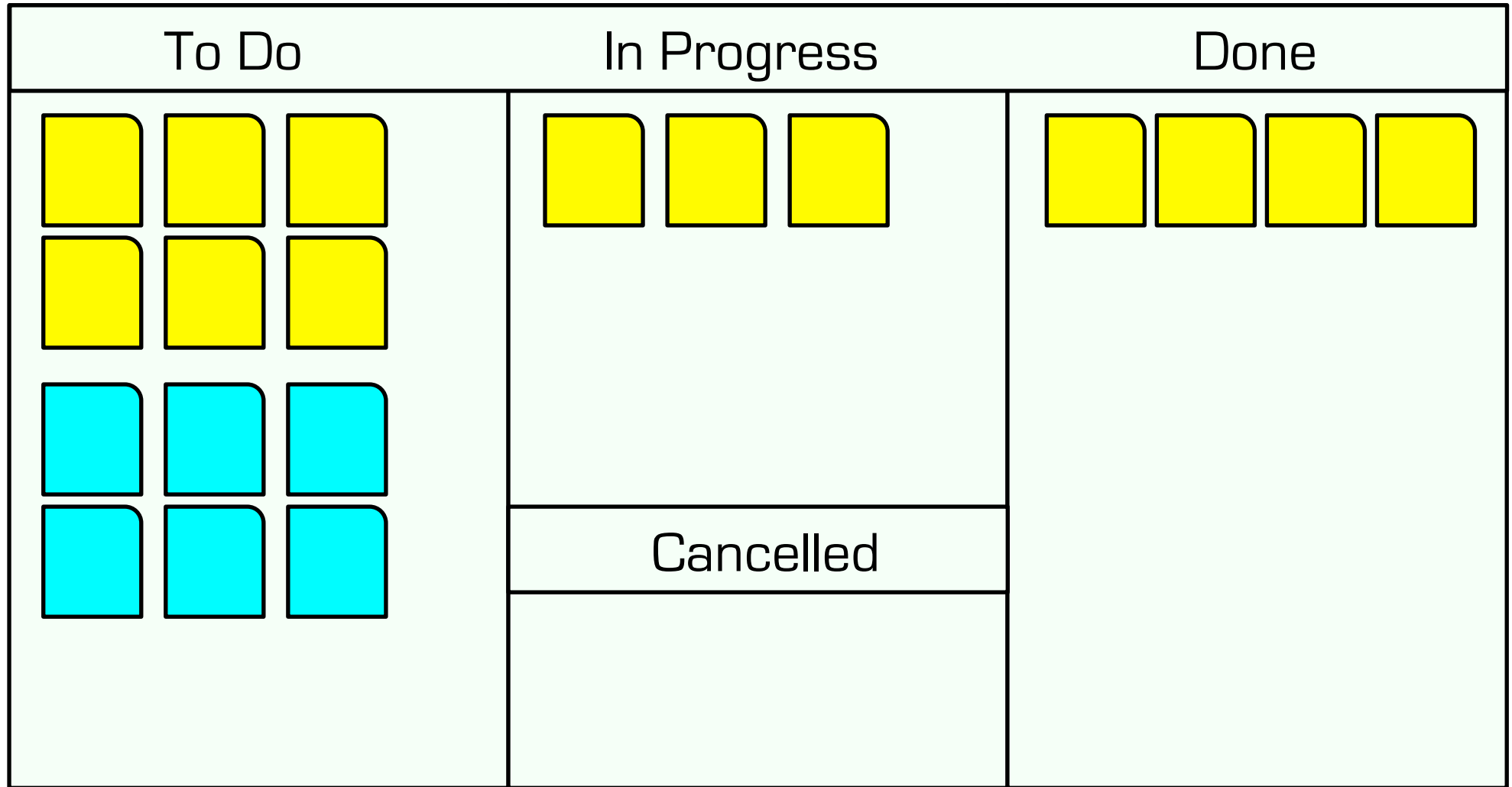
Explore editing profiles
with invalid usernames
to discover if there are
any instances where
username constraints
are not enforced

Types of sessions

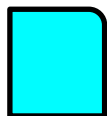
- Survey / discovery
- Deep coverage / targeted
- Broad coverage
- Touring
- ...



Charters: plan sessions on a dashboard

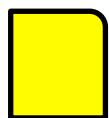
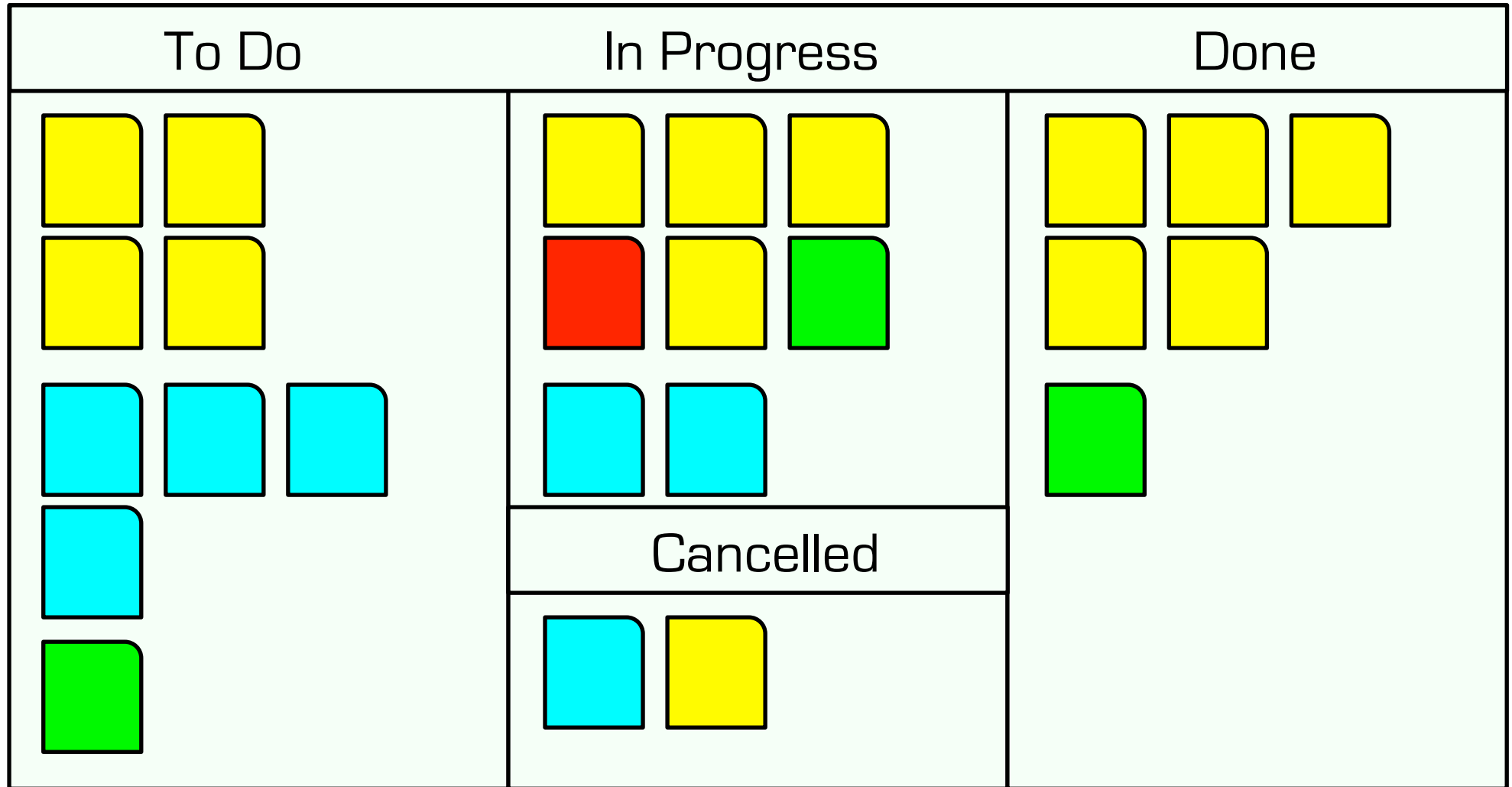


Feature 1

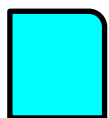


Feature 2

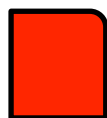
Dashboards



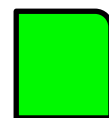
Feature 1



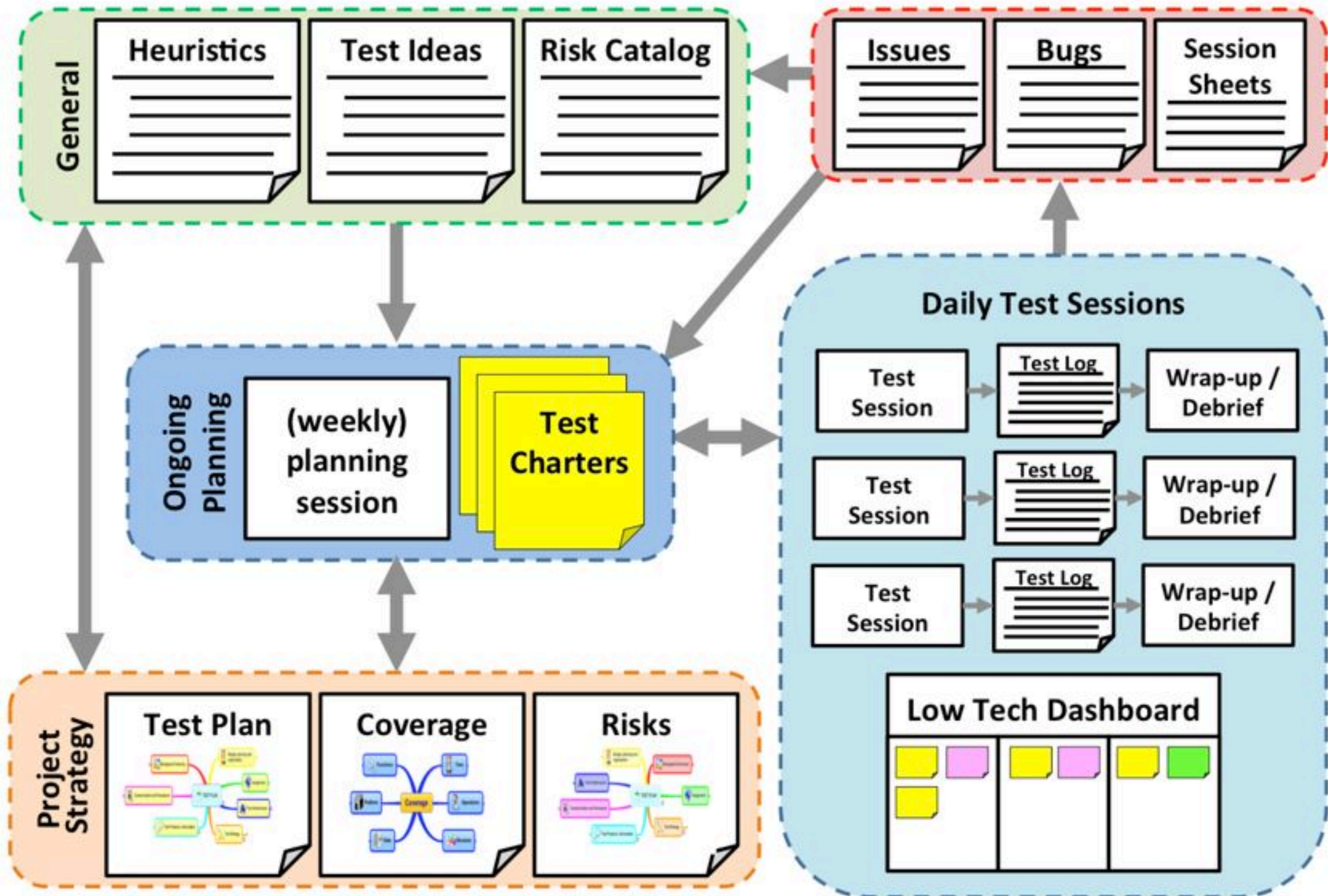
Feature 2



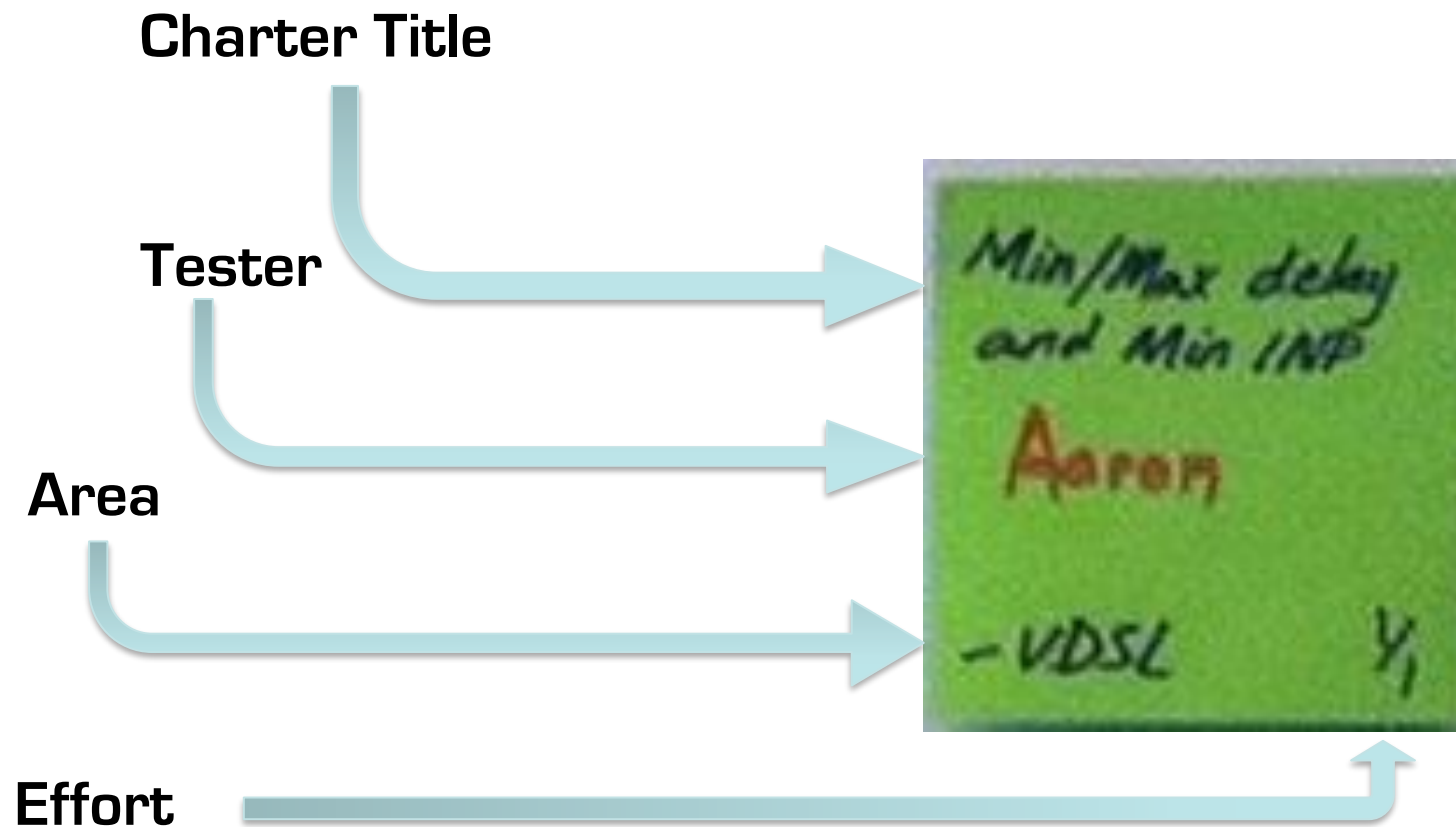
Bug



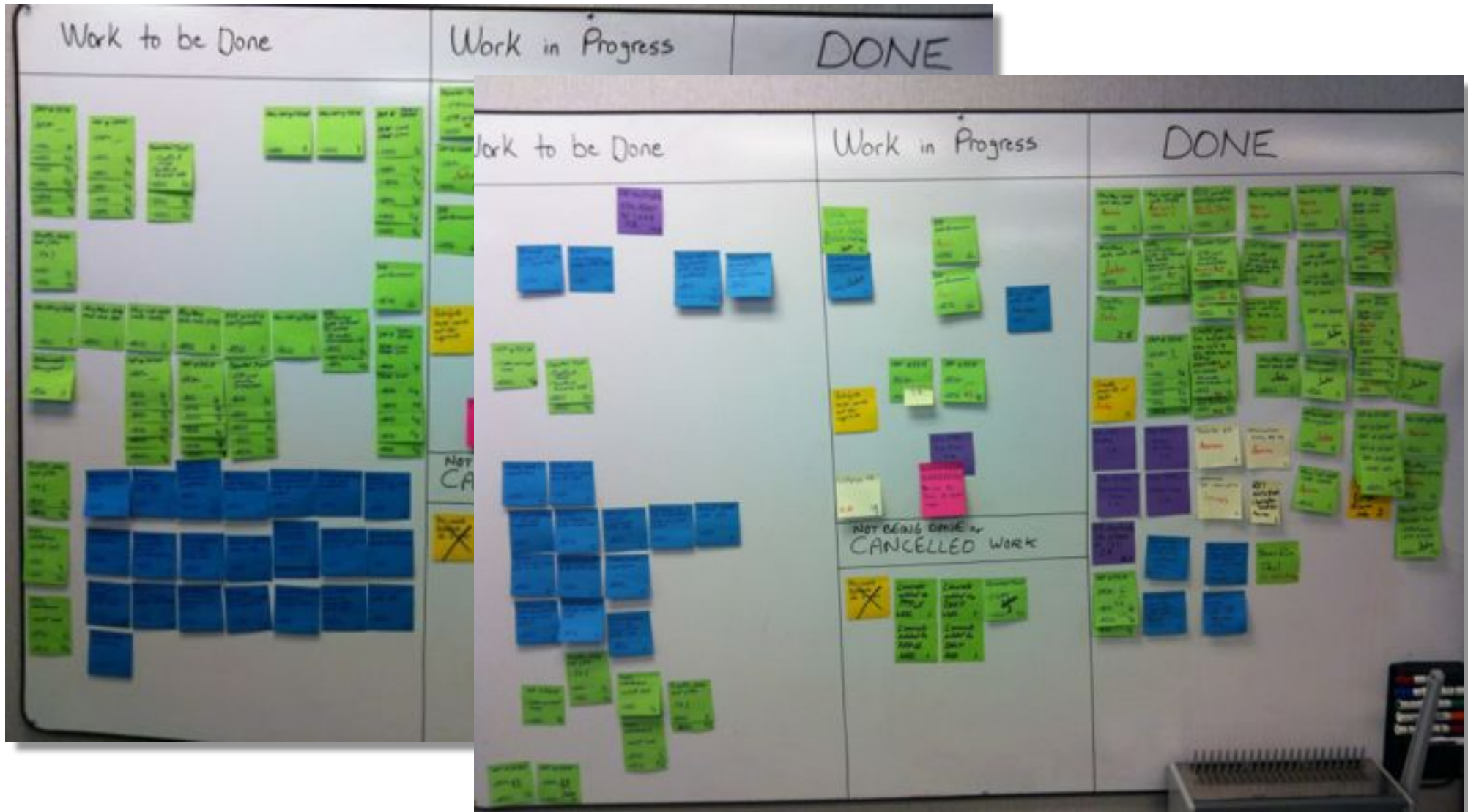
Unplanned work



Dashboards - Stickies



Dashboards – The whiteboard



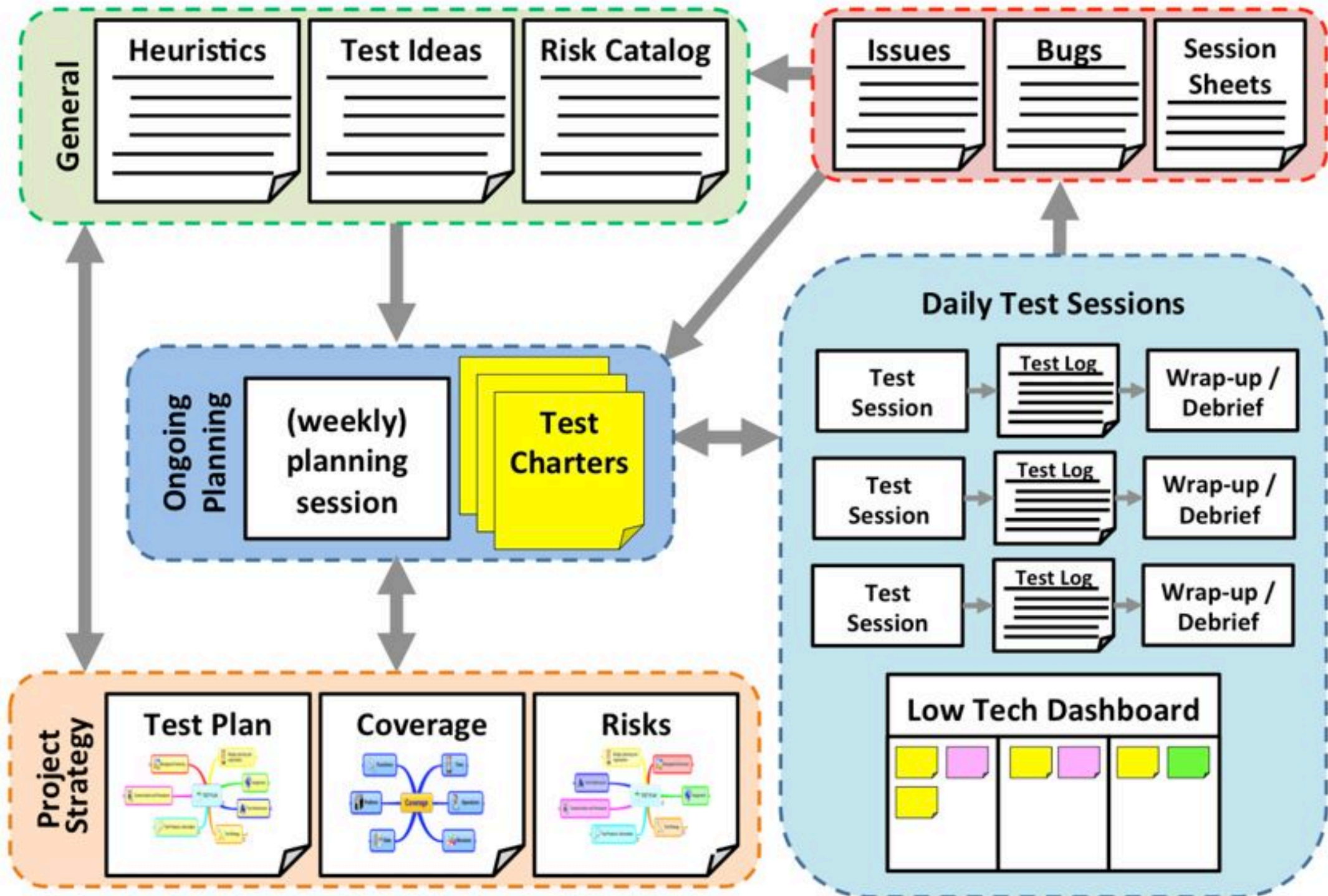
Note taking

- Session report <charter name>
- Name of tester
- Time & Date
- Environment
- Mission
- Notes
- Bugs
- Problems/issues
- Questions



Use mind map, pen & paper or tool (i.e. rapid reporter, iTester, session testers, etc)

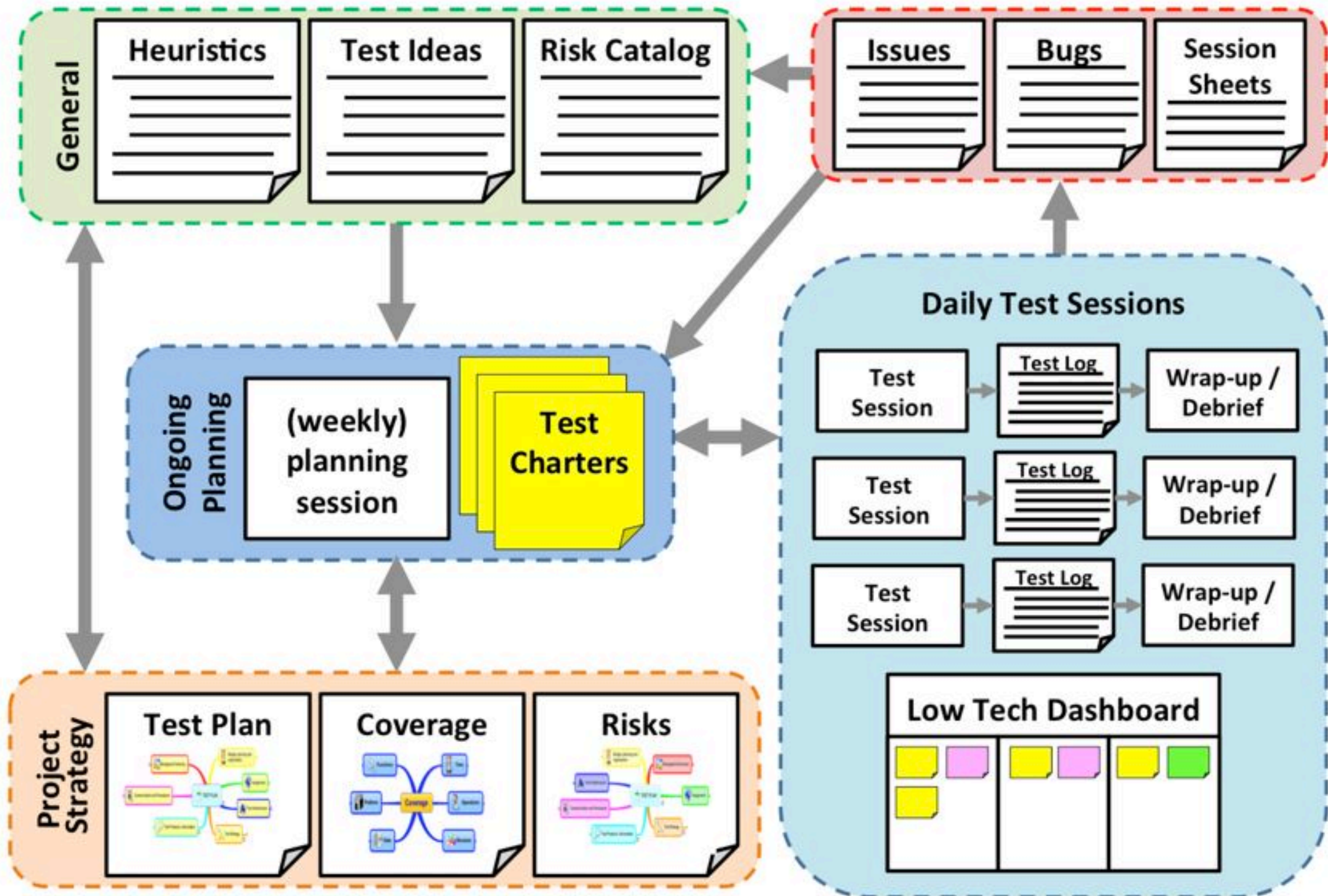
Tip for easy use: print list above or create simple txt-template



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?

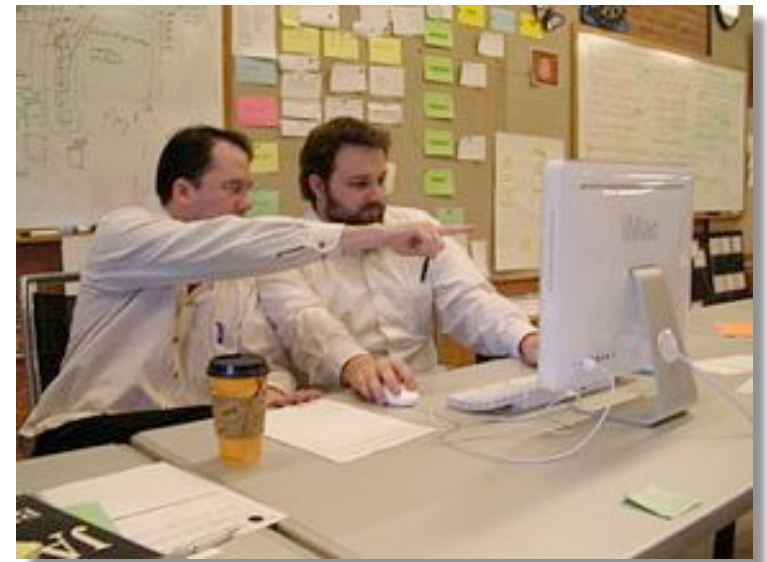


Discuss session sheet to assure that both understand it.

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

Pairing

- Focus & defocus
- Learning
- More creative and faster idea development
- Review on the go
- Sociable and more fun



Why ET is powerful ...

- 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



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



What can go wrong...

- Missing things
- Own interpretation
- Biased interpretation
- People think you are fooling around
- Misconceptions
- Undocumented ET
- Thoughtless ET
- Undirected ET
- Misdirected ET
- Bad testing story
- Insufficient information
- Poor self awareness/management
- Poor focussing



Testing is like playing music...



How to start with exploratory testing?

- Just do it!
- Remove expected results to stimulate investigation, thinking and creativity
- Remove detailed steps from test cases
- Only use logical test cases
- Do ET for small parts and extend it to other areas
- Use ET to learn about the system



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,



Some quotes

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

(Rikard Edgren – EuroStar 2012 & Gitte Ottosen – ATD 2012)

- “A tester knows that things can be different”

(Jerry Weinberg)

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

(RST)

Let's Test!!!



Exercise: Test Charters

Team up in groups of 3 or 4 and create test charters, prioritize them and estimate them for sessions of max. 20 minutes.

Fill the dashboard in 20 minutes.



Exercise: Survey Session



Work in pairs: do a survey session for 20 minutes.

Discuss roles first:

- **Who does what?**
- **Who takes notes?**
- **Change roles halfway**
- **Approach?**
- **Heuristics?**

Debrief after the session in 5 minutes.



Exercise: Deep Coverage Session #1



Work in pairs: do a deep coverage session for 20 minutes.

Discuss roles first:

- **Who does what?**
- **Who takes notes?**
- **Change roles halfway**
- **Approach?**
- **Heuristics?**

Debrief after the session in 5 minutes.





Exercise: Deep Coverage Session #2

Work in pairs: do a deep coverage session for 20 minutes.

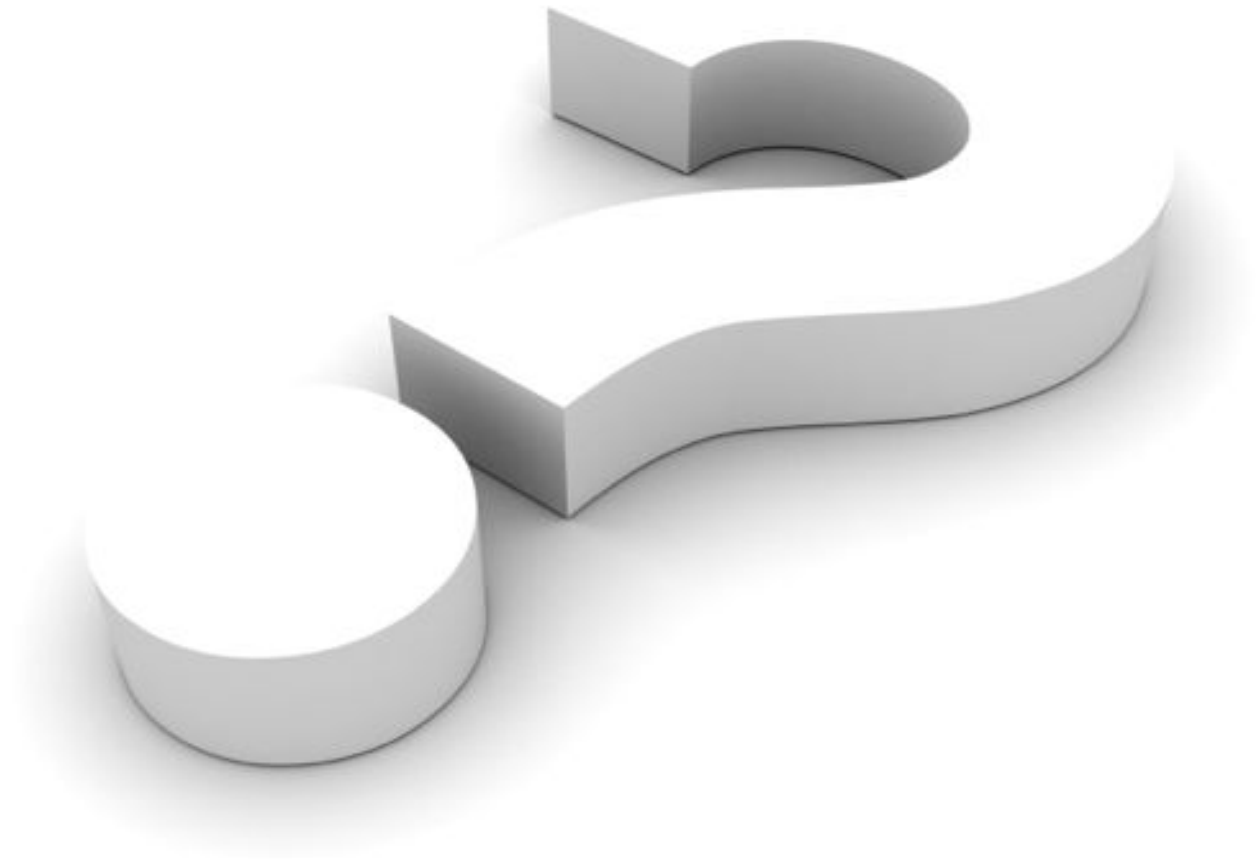
Discuss roles first:

- **Who does what?**
- **Who takes notes?**
- **Change roles halfway**
- **Approach?**
- **Heuristics?**

Debrief after the session in 5 minutes.



Questions?



Contacts

Huib Schoots



huib.schoots@codecentric.nl



@huibschoots



www.huibschoots.nl/blog
www.huibschoots.nl/links



+31 (0) 6 24 64 10 33

Pascal Dufour



pascal.dufour@codecentric.nl



@pascal_dufour



www.pascaldufour.nl



+31 (0) 6 81 92 88 98

codecentric Nederland BV
Laan der Verenigde Naties 60
3314 DA Dordrecht

www.codecentric.nl