

# Exploratory on Purpose

**Jon Bach**

Manager for Corporate Intellect

[jonb@quardev.com](mailto:jonb@quardev.com)

**TISQA 2007**



# Exploration is discovery...

?

?

?

?

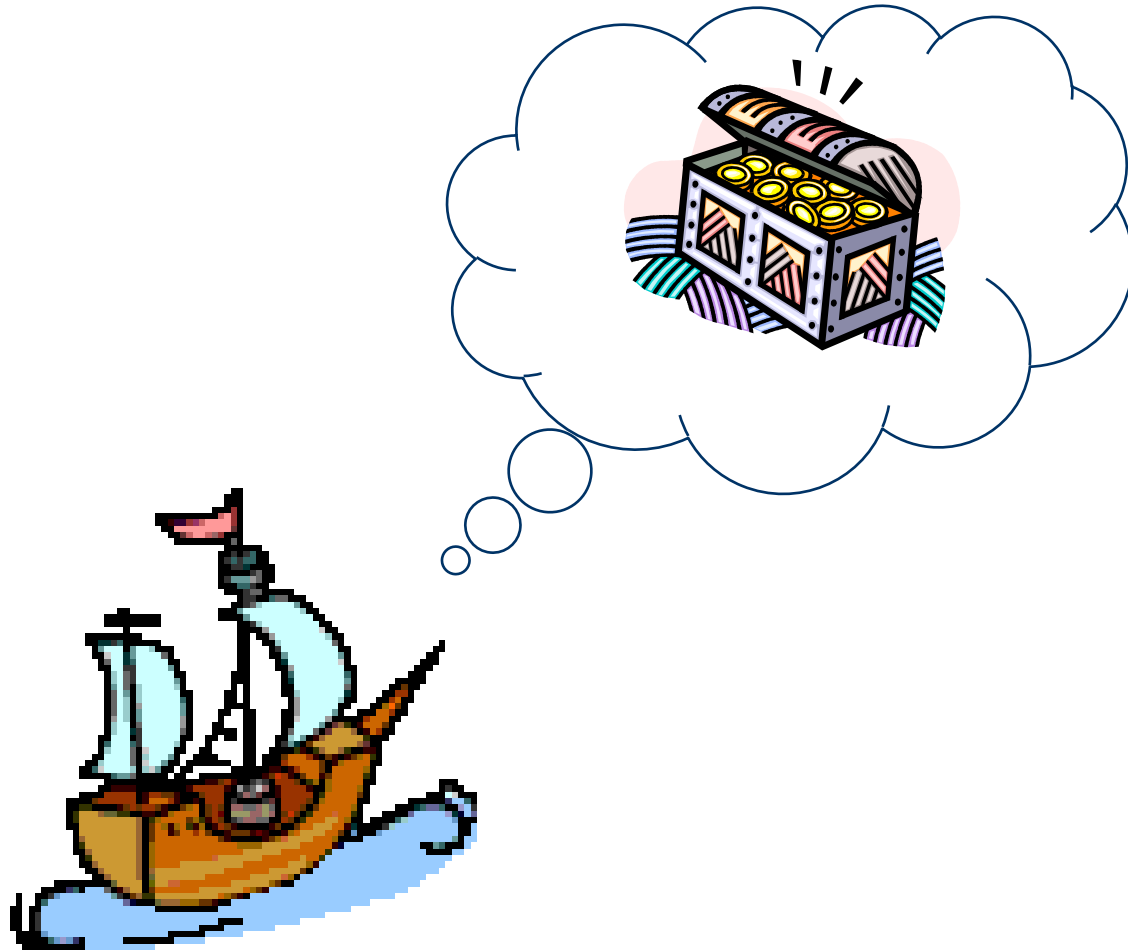
Parts Unknown

?

?



**...that starts with an idea...**



*(C) Jon Bach / Quardev Laboratories*

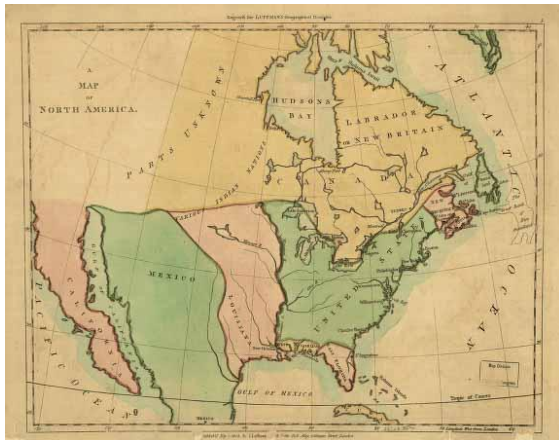
**and ends with a perception...**



Earth stops here...

# ...depending on the mission

**Before  
exploring**



**After  
exploring**



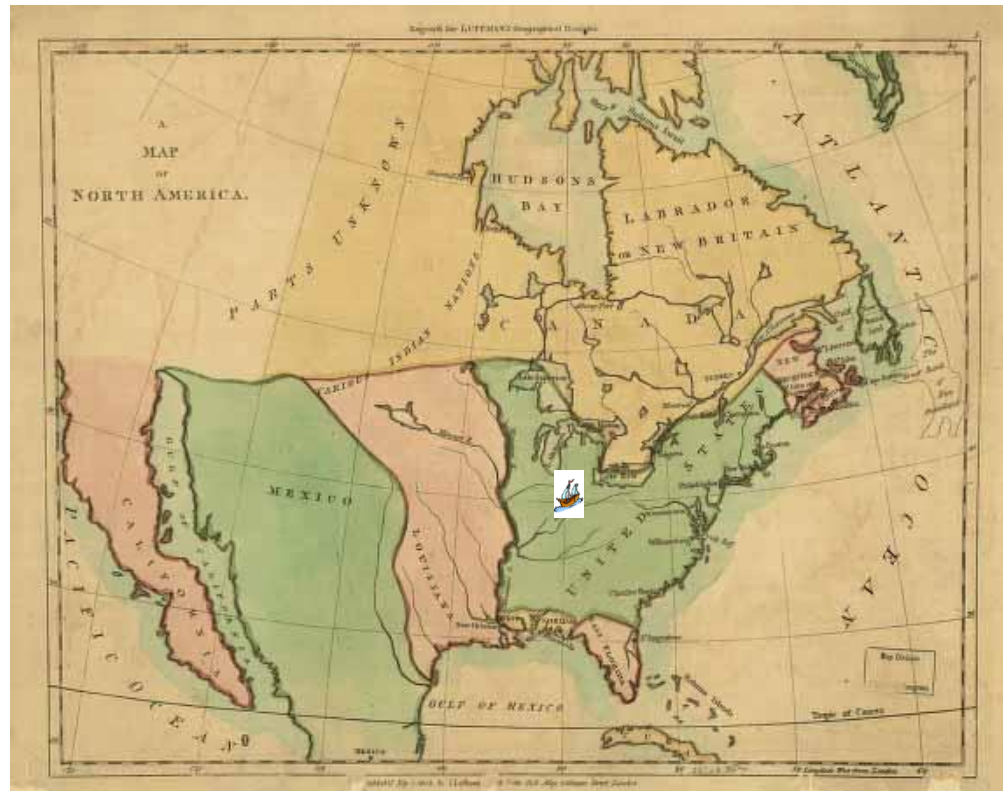
# **...mission, mission, mission**

---

If you don't know your mission,  
you're not testing.

# Lewis & Clark, 1802

Mission: Find a water passage across North America...



(C) Jon Bach / Quardev Laboratories

# The charter from Jefferson

---

*“The object of your mission is to explore the Missouri river, & such principal stream of it, as, by its course & communication with the water of the Pacific ocean may offer the most direct & practicable water communication across this continent, for the purposes of commerce.”*

<http://www.monticello.org/jefferson/lewisandclark/instructions.html>



# Sponsors and stakeholders

---

- Test Manager
- Product Manager
- CEO
- Customer
- Developer
- Marketing
- Tech Writer
- Customer Support
- Other testers

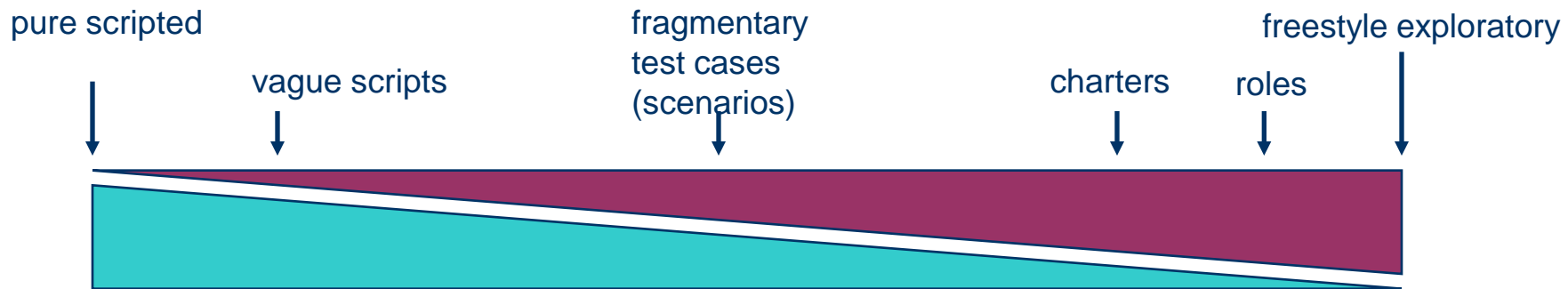


# Chartering

Making your own decisions about what you will work on and how you will work. Understanding your client's needs, the problems you must solve, and assuring that your work is on target.

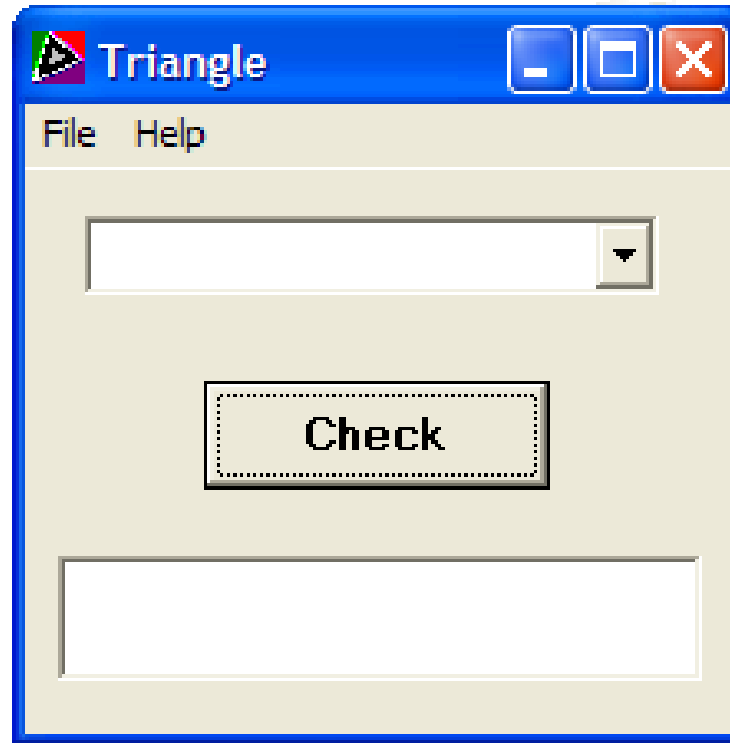


# The “tester freedom” scale



To know where a test falls on this scale, the tester must ask themselves: *“to what extent am I in control of the test, and from where did the idea originate?”*

# Mission: Find a severe bug



# How did you *\*find\** that?

## Some Exploration Skills and Tactics

***“MR.Q COMP GRABC R&R?”***

Modeling

Resourcing

Questioning

Chartering

Observing

Manipulating

Pairing

Generating/Elaborating

Refocusing

Alternating

Branching/Backtracking

Conjecturing

Recording

Reporting

***Exploratory testing is a mindset using this skillset.***

# Charter-creation method #1

---

## Session-Based Test Management

Time-boxed missions for the tester, after which they deliver a test report with Notes, Bugs, and Issues.

# Starting points

---

Test Plan Evaluation Model

Test Planning Checklist

Heuristic Test Strategy Model

# Some sample session charters

- **Installation:** When installed, does Triangle! put any files in the wrong places? Does it leave any files for the uninstall? Check the registry keys, use InCtrl to see what changes are made. Installation is new, so we want to be sure it's clean.
- **Boundary testing:** We got word from customer support that there are run-time errors when using integers over 32000 but no one can repro it. Best recon is on Win XP Pro with Office 2003 running in the background. Sam K. in CSS says you can use his machine, and he also has customer specs.
- **Ship drill:** Start Triangle! right out of the box. For example, is the readme ready to go? We're waiting from word on Legal as to the License Agreement, but that shouldn't hold you up. Also make sure you hit Vista and see what issues arise there.
- **Claims testing:** Triangle is meant for first graders, but we plan to ship a version to General Dynamics in a few months. Try some usability profiles or personas to see what functions become more or less risky. Also, discover the algorithm by which Triangle! reports its results. Is it way off from what a user would expect. Does it cause the user to lower their confidence?



# What to document

	Historical Explorer	Tester
<b>Observations</b> (To the degree you think they are relevant to stakeholders)	<ul style="list-style-type: none"><li>• drawings of flora / fauna</li><li>• descriptions of indigenous people</li><li>• landmarks</li></ul>	<ul style="list-style-type: none"><li>• feature model</li><li>• text from log files</li><li>• text from dialogs</li></ul>
<b>Conjectures</b> (Inferences based on experiences. After I test, I think I know something)	<ul style="list-style-type: none"><li>• what is this thing?</li><li>• where should we go today?</li><li>• how do we get there?</li><li>• new orders from HQ?</li><li>• are those people hostile?</li></ul>	<ul style="list-style-type: none"><li>• test ideas</li><li>• questions</li><li>• product and project issues</li><li>• concerns</li><li>• risks</li></ul>
<b>Project information</b> (Independent of observer)	<ul style="list-style-type: none"><li>• mission</li><li>• supplies and staff</li><li>• latitude / longitude</li><li>• death and disease</li><li>• supply status</li></ul>	<ul style="list-style-type: none"><li>• charter</li><li>• test actions</li><li>• config info</li><li>• build details</li><li>• tools used</li></ul>

# Charter-creation method #2

---

## Open-Book Testing

The act of creating open-ended questions such that...

# ...testers...

...are immersed in the product right away, building a model or mind map.

...learn how they are provoked into critical thinking by being exposed to many types of questions (test ideas).

... quickly find bugs and raise issues in answering the questions they are given

# The Backstory

Excerpts from Flight Sim Ground School “open book” exam:

- Which airplanes are equipped with slotted flaps?
- Describe the steps needed to start the engine of the DC3.
- If the vacuum system failed in the 172, what instruments would no longer function?
- Joe, an aviation history fanatic, buys Combat FS and complains to tech support that the P-51 flight model is “all wrong” because in a dogfight, “the airplane stalls at 200 MPH.” Joe flies with the Realism Settings set to Hard and he knows that the P-51 should stall at about 100 MPH. Explain to Joe how it is possible for an airplane to stall at twice its published stalling speed.

# In other words ...

---

**The answer may matter less than how it was derived.**

For example,

Which airplanes are equipped with slotted flaps?

...can lead to the following strategies to find the answer:

- Select aircraft / spot plane view should show me the type
- Aircraft details: should tell me the type of flaps
- Flap function (shortcut key, panel icon, keyboard) can be tested to verify the flap animation is consistent with model
- Flight model affect (whether the flaps *behave* as slotted) can be tested in flight

# Back at my desk, I reframe...

OBT question	The spirit of the test
Which airplanes are equipped with slotted flaps?	Confirm or refute a rumor that the texture maps for slotted flap animations weren't ready, but were dropped into the build anyway
Describe the steps needed to start the engine of the DC3.	Investigate a claim from the Developer, a DC3 pilot, that omissions in the "pilot checklist" feature are now fixed
If the vacuum system failed in the 172, what instruments would no longer function?	When enabled, does the "system failures" feature show that the altimeter and airspeed indicators are incorrect?
Can a 737 stall at a speed that's twice as fast as its published stalling speed? <i>(C) Jon Bach / Quardev Laboratories</i>	Any plane should stall at any speed if the attitude of the angle of attack is too steep.

# Questions



- test ideas
- test cases
- test scenarios
- test plans
- test scripts
- test designs
- test strategies
- test heuristics

# Questions



- test ideas
- test cases
- test scenarios
- test plans
- test scripts
- test designs
- test strategies
- test heuristics

These comprise the exam to which software will either pass or fail.



# Proposed process

---

- Interrogate: The test manager or tester develops a list of questions to answer.
- Manipulate: The testers execute actions to answer the question.
- Observe: Testers take notes on what they find.
- Plan: Testers determine any follow-up questions (tests) that occur to them, in preparation to debrief their results.
- Evaluate: Testers and test manager meet to compare answers (test results).
- Negotiate: After the debrief, testers and test managers talk about the appropriate next steps in mission or coverage

# Resources

---

Questions and answers can originate from the same sources:

- Documentation / Specifications
- Web forums
- Previous products
- Team members
- Competing products
- PSS data / KB articles
- Your expertise
- Heuristics
- Help files
- Manuals

# A few non-obvious (?) sources for charters

---

- Bug database
- Testers (paired testing)
- Programmers (different domain expertise)
- Similar (or competing) products
- Customer Support
- Claims made by marketing
- Emails / Meetings / RSS feeds

# Testing ourselves

---

**Chartering is an opportunity for testers and managers to cultivate and improve testing skill:**

How did you arrive at that answer?

What did you see along the way?

Was there anything confusing about the questions?

Any riffs off of questions?

What test ideas did others have with the same question?

# What managers might ask

How did you spend your time?

What did you find?

Did you need some help / tools?

Do you think there's more to do here?

Was this charter reasonable?

## Agenda: "PROOF"

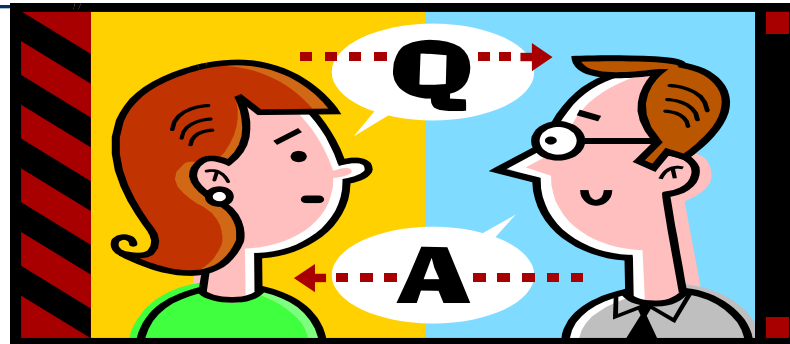
**P**ast

**R**esults

**O**bstacles

**O**utlook

**F**eelings



# The real message

<b>What's being asked</b>	<b>What they may be thinking</b>
<i>What was your mission?</i>	<i>Remind me what I told you to do...</i>
<i>How did it go?</i>	<i>What do I worry about next?</i>
<i>How far did you get?</i>	<i>Are we closer to shipping?</i>
<i>Need anything?</i>	<i>Can I speed this along?</i>
<i>When will you be done?</i>	<i>Will I get my bonus?</i>

# Conclusion

---

- *Brainstorming, scenario, user stories, dimensions*
- Chartering Method: Open-Book Testing
- Management Method: Session-Based tests
- Idea Method: Heuristic Test Strategy Model
- Technique inventory: stress, flow, risk, claims, etc...