

The Secret Life of Automation

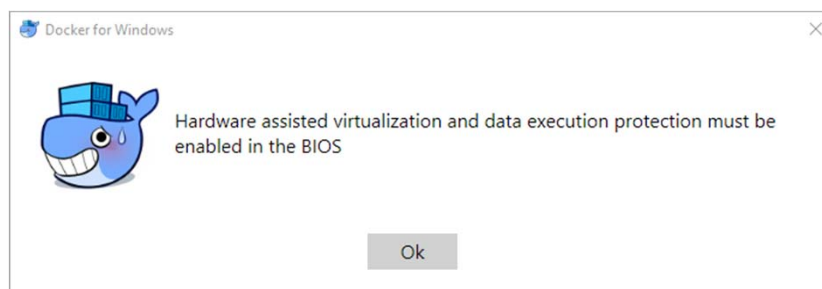
Michael Bolton

michael@developsense.com
<http://www.developsense.com>

@michaelbolton

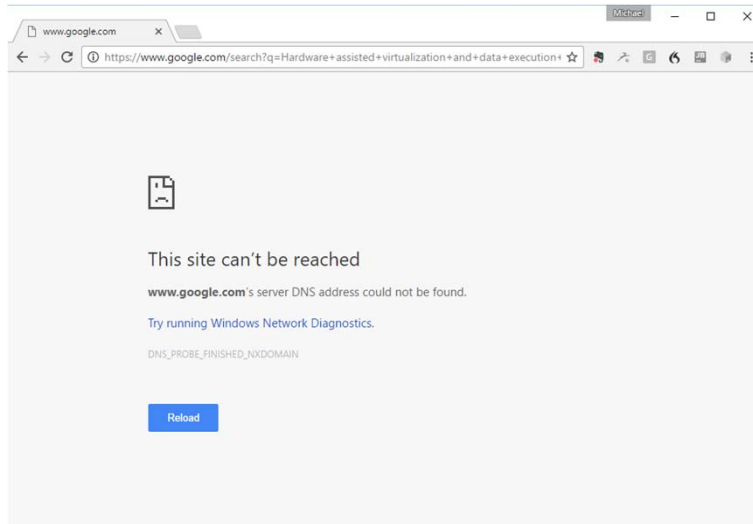
The Secret Life of Automation - 1

Here's what happened when I tried to install Docker.



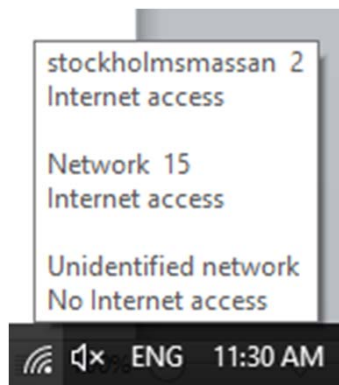
The Secret Life of Automation - 2

I guess I'll look that up.



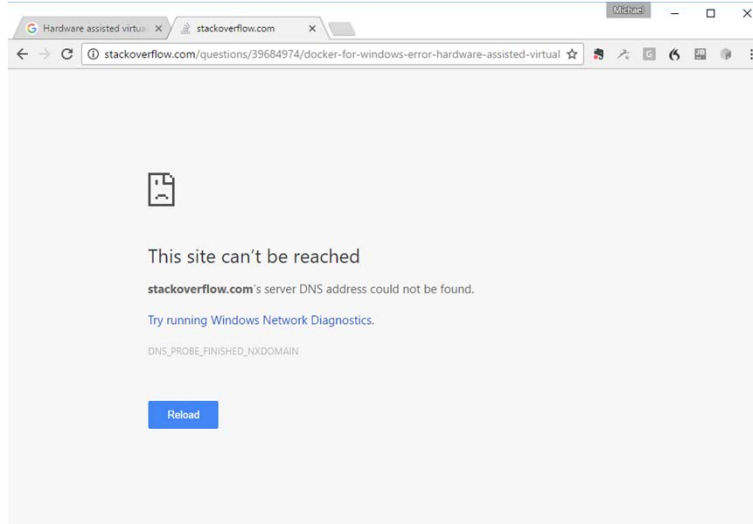
The Secret Life of Automation - 3

Wait... what?



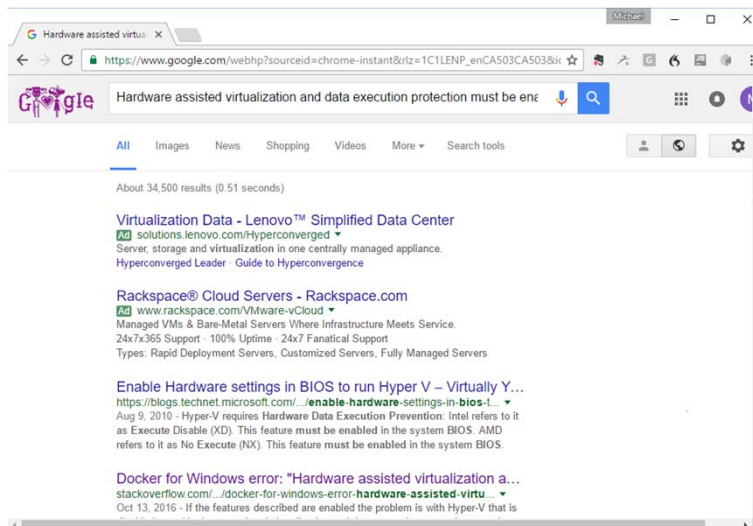
The Secret Life of Automation - 4

No Google? What about Stack Overflow?



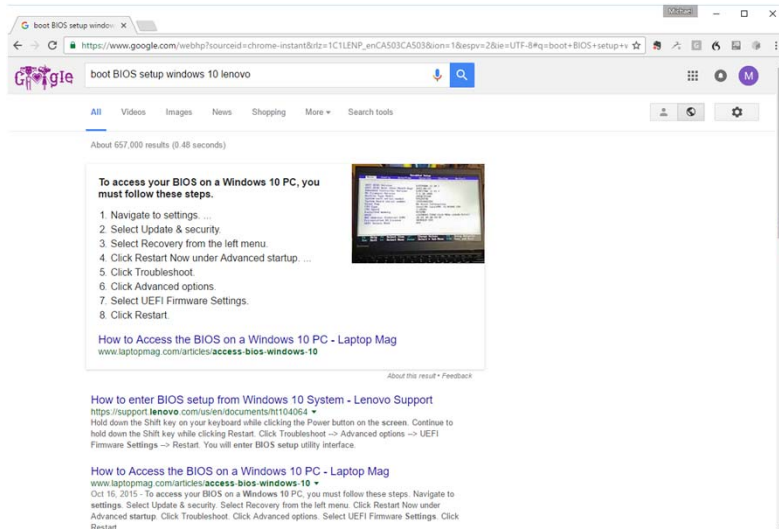
The Secret Life of Automation - 5

OK, Google's back.



The Secret Life of Automation - 6

OK, making progress.



The Secret Life of Automation - 7

Applicable Brands

- Lenovo, Idea laptops and desktops

Operating System

- Windows 10

Solution

↓ [To enter BIOS via function key](#)

Immediately pressing (Fn+) F2 or (Fn+) F1 key 5-10 times after power on button is pressed.

↓ [To enter BIOS from Windows 10](#)

The Secret Life of Automation - 8

- Windows 10

Solution

↓ [To enter BIOS via function key](#)

Immediately pressing (Fn+) F2 or (Fn+) F1 key 5-10 times after power on button is pressed.

↓ [To enter BIOS from Windows 10](#)

However, computer boots fast, maybe you often miss the trick to press F2. You can enter UEFI firmware from windows 10. Step



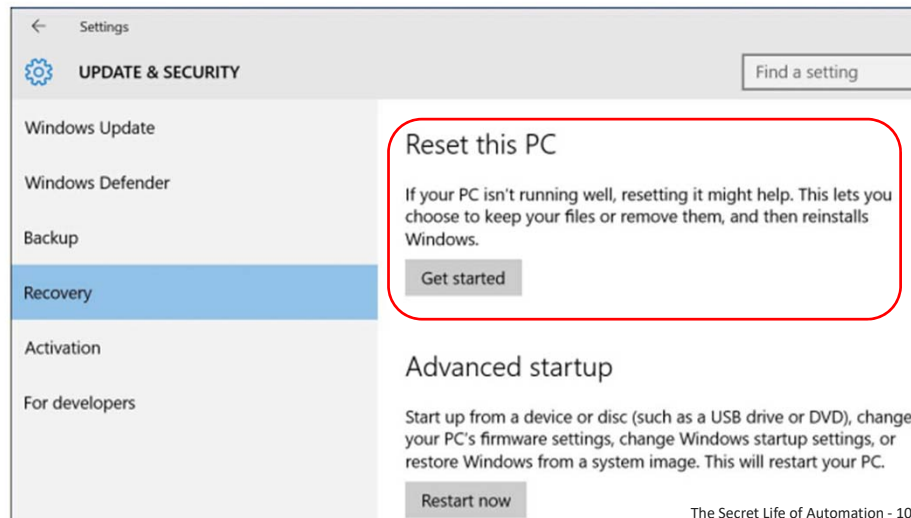
1. Click  --> Settings or click New notifications  at the lower-right corner --> All settings to open Settings (the screen, and then tap Settings.)

2. Click Update & security



The Secret Life of Automation - 9

3. Click Recovery, then Restart now.

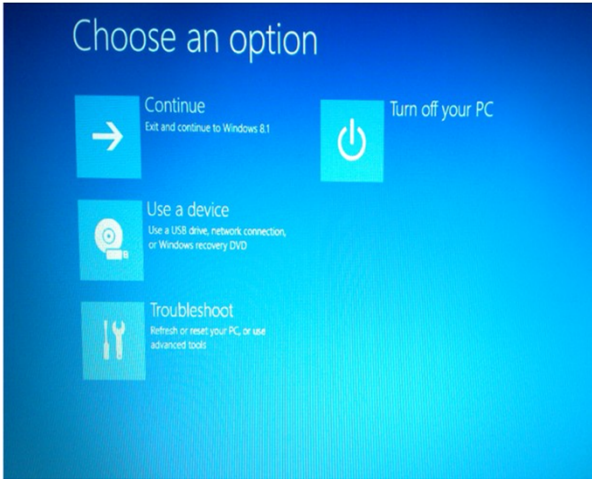


The Secret Life of Automation - 10

SHOP SUPPORT COMMUNITY

Product Home Drivers & Software How-Tos & Solutions Documentation Diagnostics Warranty & Repair Parts & Accessories

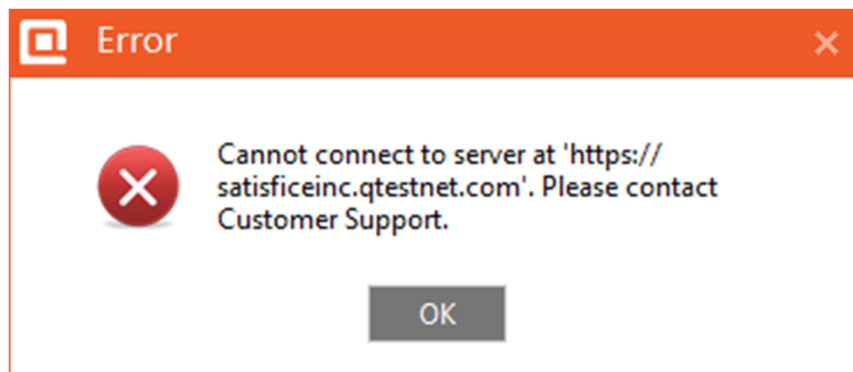
4. Options menu will be seen after executing the above procedures. Click Troubleshoot.



5. Focus on Advanced options.

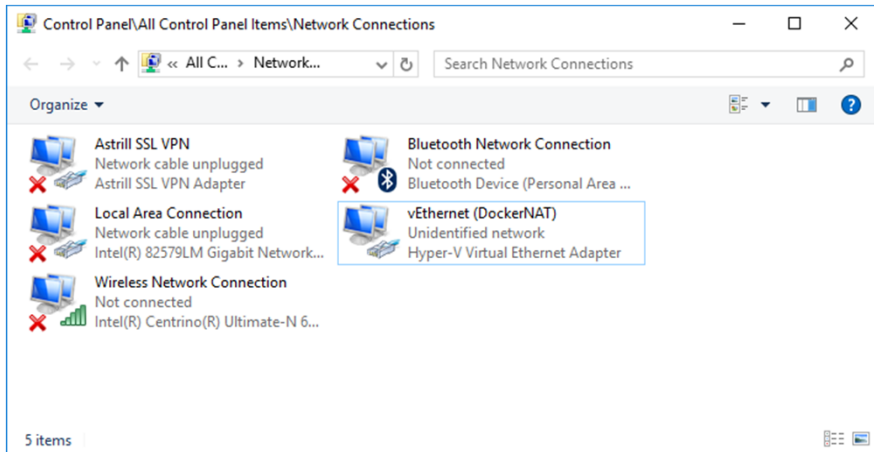
The Secret Life of Automation - 11

Geez, I should record this. But...



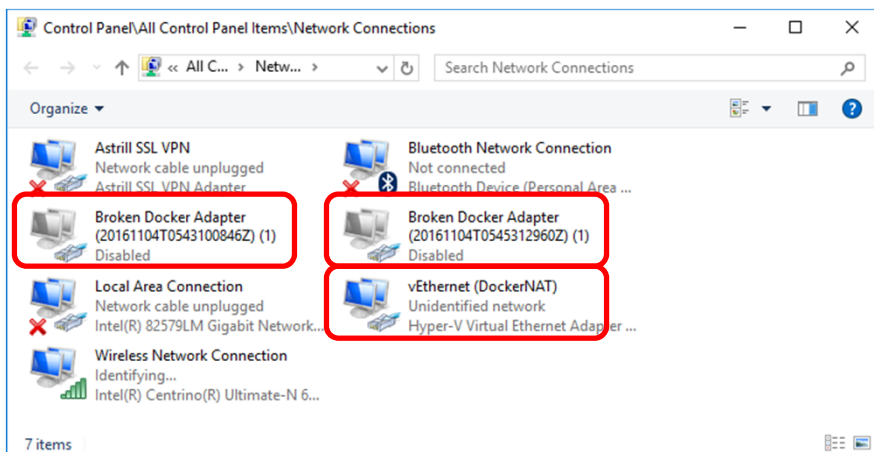
The Secret Life of Automation - 12

A few days later... Huh?



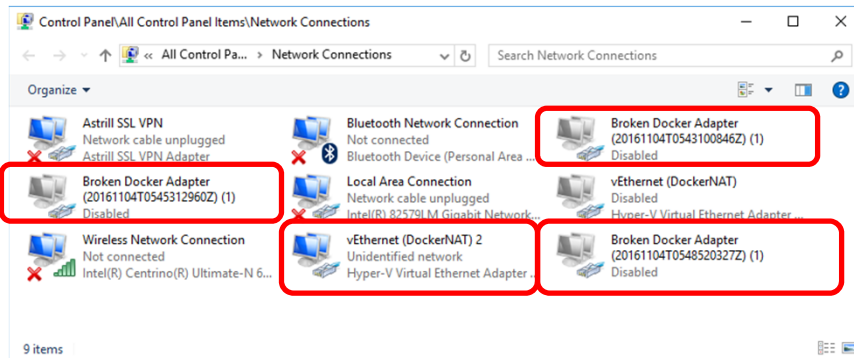
The Secret Life of Automation - 13

After a reboot...



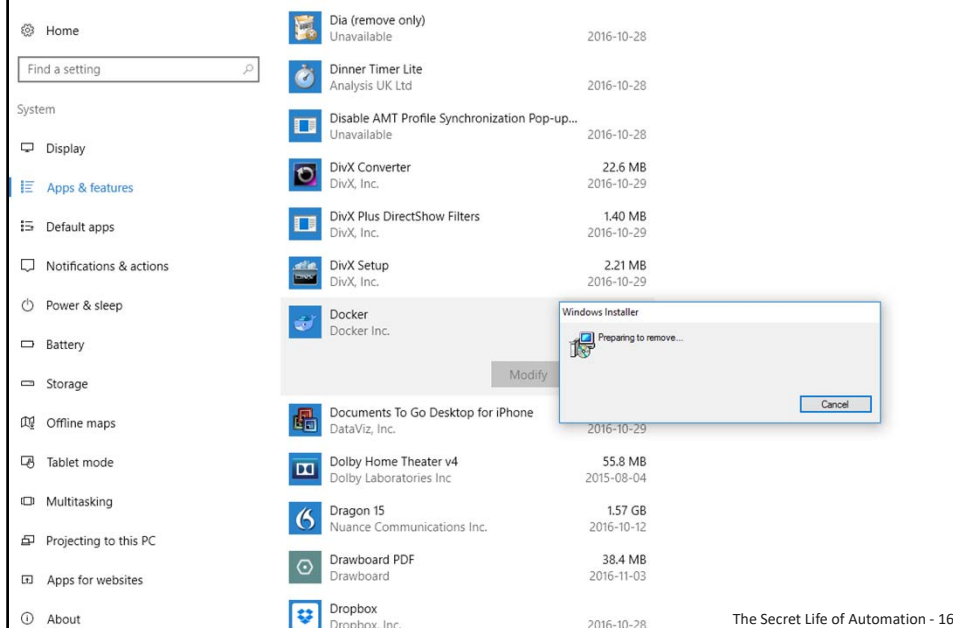
The Secret Life of Automation - 14

They're BREEDING now.



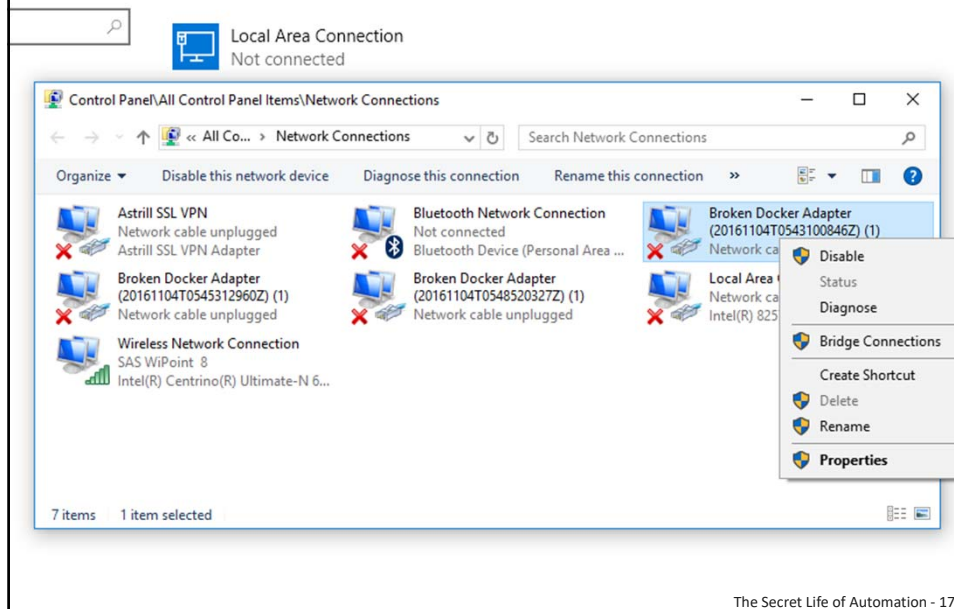
The Secret Life of Automation - 15

OK, I'm out. Uninstalling.



The Secret Life of Automation - 16

Ethernet **But the zombies live!**



Secret:

This sort of thing is going on all the time. Sometimes testers will talk about it with each other... but often not with management.

The Secret Life of Automation - 18

Tip (for managers):
Problems with any part of the development process (including deploying tools) are *to some degree* normal—and they're important sources of information. Make sure testers report tool successes and failures realistically.

The Secret Life of Automation - 19

Tip (for testers):
Problems with setting up and using tools are *to some degree* normal. Anything worth doing takes *some* effort. If it's too hard OR too easy, it might not be worthwhile. Report what happens truthfully.

The Secret Life of Automation - 20

There's lots we don't talk about!

success = commercial advantage
failure = embarrassment
current state = unknown

The Secret Life of Automation - 21

Find the roaches:

Boss's fantasy (searching for problems is easy!)



The Secret Life of Automation - 22

Find the roaches:

Tester's actual job (good searching for problems is hard)



Why test?

Program managers want to know ONE thing above all:

**Are there problems
that threaten
the on-time
successful
completion of the project*?**

* At any level of granularity


The Secret Life of Automation - 24

What is not being talked about?


what people do in “test automation” work
what people do in *all* testing work
what people do in development work
what machinery really does
how tools could *extend* skills
concepts and forms

The Secret Life of Automation - 25

OK, so what are we talking about?

 automation n. “A high degree of mechanization in **manufacture**, the **handling of material** between processes being automatic and **the whole system** being automatically controlled.”








—*Chambers Dictionary (iOS)*

 automation n. “the **use** or introduction of automatic equipment in a **manufacturing** or other process or facility.”

—*Concise Oxford Dictionary*

The Secret Life of Automation - 26








(How about “tool”?)

-  a working instrument, esp. one used by hand
-  the cutting part of a machine tool
-  someone who is used as the mere instrument of another
-  anything necessary to the pursuit of a particular activity
-  a fool (slang)
-  a despicable person (slang)
-  a utility, feature or function available as part of *e.g.* a word processing package or database (computing)

—*Chambers Dictionary (iOS)*

The Secret Life of Automation - 27

In Rapid Software Testing, we offer...

-  tool (n.)
 -  any contrivance used to fulfill a human purpose
-  “test tool” (idiom)
 -  *any* tool used in the service of testing
-  automation (n.)
 -  1. any process *entirely* performed by a tool
 -  2. a tool capable of such performance

The Secret Life of Automation - 28

OK, so what are we talking about?

testing

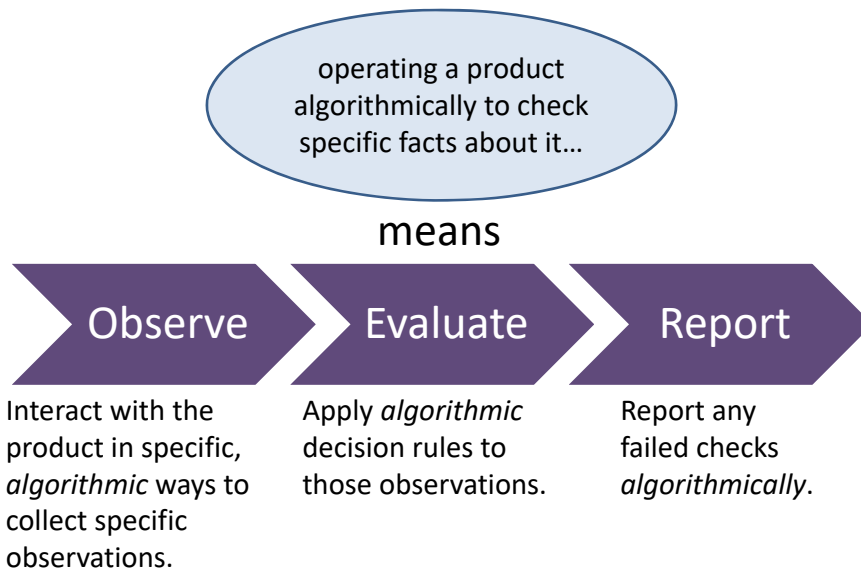
evaluating a product by **learning** about it through **exploration** and **experimentation**, which includes to some degree: **questioning**, **study**, **modeling**, **observation** and **inference**, including...

checking

the process of making evaluations by applying **algorithmic** decision rules to specific observations of a product

The Secret Life of Automation - 29

Call this “Checking” not Testing



The Secret Life of Automation - 30

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)

The Secret Life of Automation - 31

Testing Is *More Than Checking*

- *Checking* is okay, but it is mostly focused on confirming what we know or hope to be true.
- To escape problems with verification, we must do more than checking; we must *test*.



I'm very fast...
but I'm slow.

See <http://www.developsense.com/2009/08/testing-vs-checking.html>

The Secret Life of Automation - 32

Secret:
Confirmation is a problem.

The Secret Life of Automation - 33

On Confirmation

Most of the technology of “confirmatory” non-qualitative research in both the social and natural sciences is aimed at **preventing discovery**. When confirmatory research goes smoothly, **everything comes out precisely as expected**. Received theory is supported by one more example of its usefulness, and requires no change. As in everyday social life, **confirmation is exactly the absence of insight**. In science, as in life, dramatic new discoveries must almost by definition be accidental (“serendipitous”). Indeed, they occur only in consequence of some mistake.

Kirk, Jerome, and Miller, Marc L., Reliability and Validity in Qualitative Research (Qualitative Research Methods). Sage Publications, Inc, Thousand Oaks, CA, 1985.

The Secret Life of Automation - 34

In other words...

- Focusing on repetition and confirming the happy path inadvertently focuses on *avoiding* finding problems...
- ...but if you want to find the banana peels that the customers will trip over, you'd better
 - map the product (and iterate)
 - *vary* your paths
 - look for *problems*, in both products and tools
 - learn from *every* problem



The Regression Testing Fantasy

"I rerun my old tests to ensure that nothing has broken."



The Secret Life of Automation - 36

The Regression Testing Fantasy

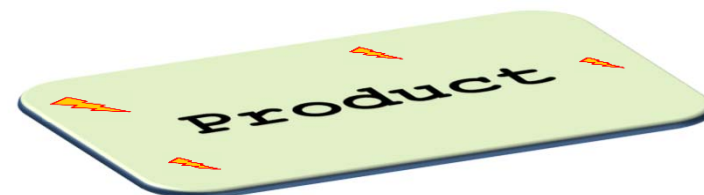
"I rerun my old tests to ensure that nothing has broken."

This can only be true if your old tests cover **everything** completely with **perfect** oracles so that **all** conceivable bugs are detected...

The Secret Life of Automation - 37

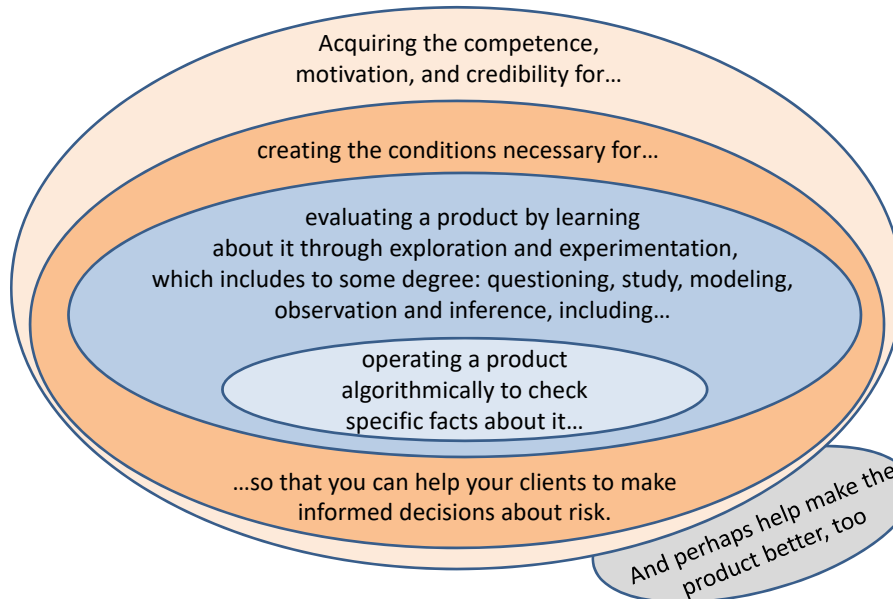
The Regression Testing Reality

"We run a smattering of old checks to ensure that they still find no bugs...
And we assume that any bug not found is also not important."



The Secret Life of Automation - 38

Testing Is *Learning About a Product*

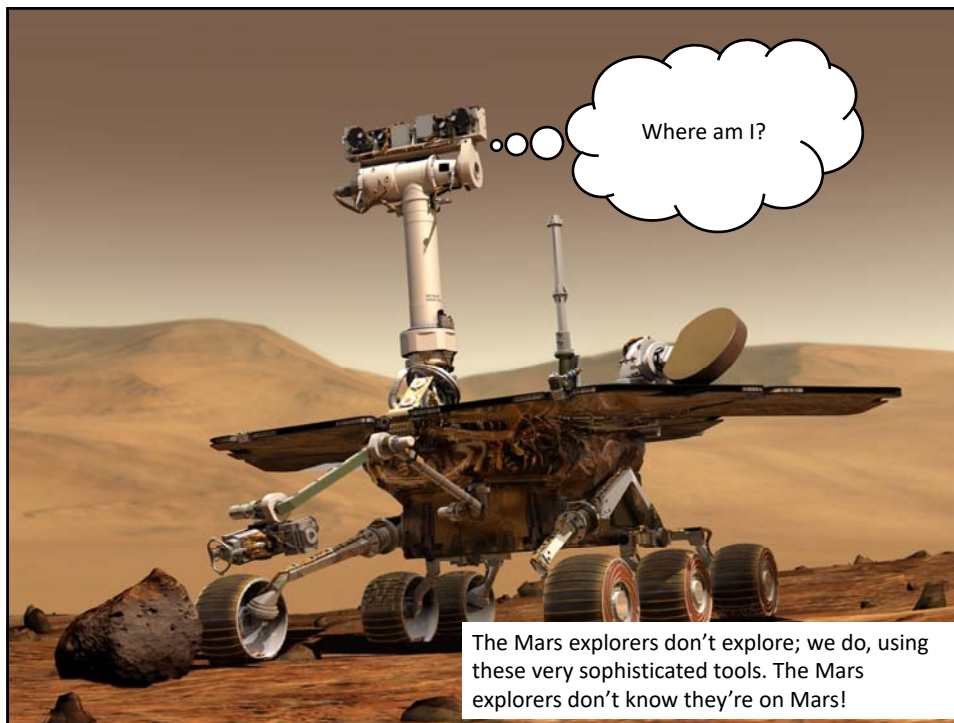


The Secret Life of Automation - 39

Secret:

Testing cannot be automated because exploration and experimentation cannot be automated.

The Secret Life of Automation - 40



Secret:
Automated *testing* does not exist.

The Secret Life of Automation - 42

**Secret:
Automation
ain't
testing.**

The Secret Life of Automation - 43

**Secret:
Automated *checking* **does exist**, and
it can be powerful. But it cannot
replace testing expertise, and it must
not attempt to displace it.**

Excellent automated checking REQUIRES

- expertise in testing
- expertise in applying tools (and recognizing when they can fool us)
- excellent tools

The Secret Life of Automation - 44

Claim:
**Tools can help us to do
more testing
faster
than we've ever done it before.**

The Secret Life of Automation - 45

Secret:
**Depending on the tools and
depending on how we use them,
tools might be helping us to do
more **lousy, shallow** testing
faster **and worse**
than we've ever done it before.**

The Secret Life of Automation - 46

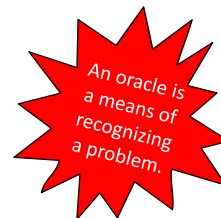
What IS BEING automated?

- TWO things performable by algorithms...
 - pressing of buttons in the GUI
 - checking output against specified results

The Secret Life of Automation - 47

What COULD be automated?

- MANY things performable by algorithms, including
 - pressing of buttons in the GUI
 - checking output against specified results
 - setup and reconfiguration of test environments
 - provision of input
 - aggregation of input
 - searching, sorting, filtering of data
 - conversion of data from one form to another
 - altering sensory modes (visualization, sound)
 - comparable or parallel product oracles
 - probes to access to internal states of a program
 - randomization
 - mapping and perturbation of state machines
 - generation of alerts



The Secret Life of Automation - 48

When Does Tool Use Go Wrong?

- When the product's design makes life harder for the tool.
- When tools are used for only for confirmation, rather than exploration, experimentation, discovery, investigation, and learning.
- When tools are inappropriately aimed at problems that require human recognition.
- When people fail to apply skepticism to tools or to other aspects of testing.
- When people believe that tools can test.
- When tool use isn't focused on this question...

The Secret Life of Automation - 49

**Are there problems
that threaten
the on-time,
successful
completion of the project?**

The Secret Life of Automation - 50

Secret:

You can use tools to help map, probe, visualize, disrupt, amplify, report, randomize... and **EXPLORE.**

The Secret Life of Automation - 51

Analysis to the rescue!

By exploring and analyzing the product (with powerful tools!) we may discover

- ways to produce useful maps of the product
- obstacles that we could surmount with help
- “broken leg” problems to influence strategy
- specific risks to focus on
- radical shortcuts for preventing problems
- radical shortcuts for exploring specific risks
 - in other words, opportunities for tool *support and efficient, worthwhile checking*

...without writing any check code!

The Secret Life of Automation - 52

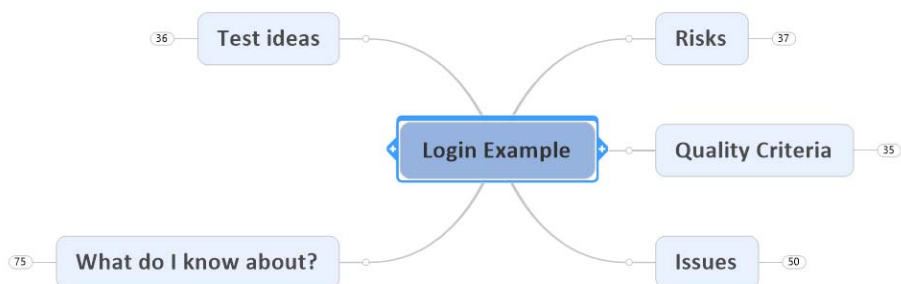
What I Have Learned Online

- The most thoroughly tested part of any application is

You are now logged in.

The Secret Life of Automation - 53

Login Stuff We MIGHT Want To Test



The Secret Life of Automation - 54

What I **Haven't** Learned About Online

- risk (referring to the product)
- problem (referring to the product)
- bug, error, defect, etc. (referring to the product)
- quality
- value
- coverage
- oracles
- investigation
- discovery
- learning

The Secret Life of Automation - 55

Secret:

People tend to focus on the How and the What of automated button-pushing (it's dazzling!). But not on the Why.

The Secret Life of Automation - 56

Let's remember this question:

**Are there problems
that threaten
the on-time,
successful
completion of the project?**

The Secret Life of Automation - 57

Now, for the new word...

GEMPOB

(I made this word up.)

The Secret Life of Automation - 58

Here it is in a bunch of different fonts...

GEMPOB

GEMPOB

GEMPOB

GEMPOB

GEMPOB

GEMPOB

The Secret Life of Automation - 59

What does this ugly, stupid word mean?

- **GETting**
- **M**achines to
- **P**ress their
- **O**wn
- **B**uttons



**gempob (n.) getting machines to
press their own buttons**

The Secret Life of Automation - 60

Why the fixation on GEMPOB?

Popular test framing pattern #1

1. Testing is misconceived as being about *confirming that the product works*; checking.
2. There is a large test space to cover.
3. This space is currently being covered by (shallow) “manual test cases” (i.e., human checking).
4. These checks took a long time to develop and write up (hey, the binders alone were expensive).
5. THEREFORE we don't want to throw out checks.
6. Since they're already checks, we can reuse them (even though they're crappy).

The Secret Life of Automation - 61

Why the fixation on GEMPOB?

Popular test framing pattern #2

1. Testing is misconceived as being about *confirming that the product works*; checking.
2. There is a large test space to cover.
3. People want *something* done to prevent total embarrassment (e.g. a missing screen)
4. Therefore, they'll often settle for showing that something exists and that it *can* work.
5. Getting a tool to *visit* every screen *sounds* cheap.
6. As long as we can do that, deeper bugs are simply bad luck; too hard to find.

The Secret Life of Automation - 62

A Representative Case

- Client had 1100 automated checks, developed over several years
- These took approximately 24 hours(!) to run
- Of these 1100, 100 were regularly running red
- Of those 100, 25 were known environmental problems (therefore considered non-bugs)
- Of the remaining 75, about 10% (~7 total) were regularly false-positive reports (that is, non-bugs)

What questions would you ask?

The Secret Life of Automation - 63

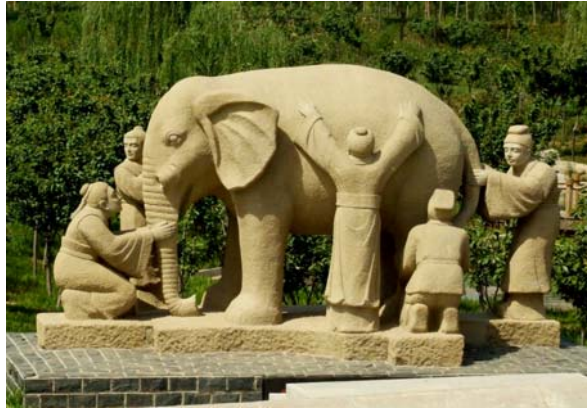
Analysis!

- ? What are these checks actually checking?
- ? Is a check a single condition, or multiple conditions?
- ? Do the checks include variation to expose bugs?
- ? How often are the checks reviewed and analyzed for relevance and accuracy?
- ? Are the checks duplicating conditions already checked by programmers?
- ? Are the programmers doing *any* of their own checking?
- ? Why the heck are the checks taking 24 hours?!
- ? What role does setup play in the timing?

The Secret Life of Automation - 64

Plus the invisible elephant...

- If 7% of the “failing” checks were unreliable, why presume that the “passing” checks were 100% reliable?



The Secret Life of Automation - 65

Secret:
There are no flaky checks.
But there *are* flaky interpretations
of results from checking.

The Secret Life of Automation - 66

Secret:
When there are “flaky” checks,
***something* has become**
desensitized.

The Secret Life of Automation - 67

Secret:
When programmers, testers,
“automators”, and toolsmiths are
separated, secrets will multiply.

The Secret Life of Automation - 68

Secret:

People who have both the tester and builder mindsets are rare, and even they find switching mindsets to be hard.

The Secret Life of Automation - 69

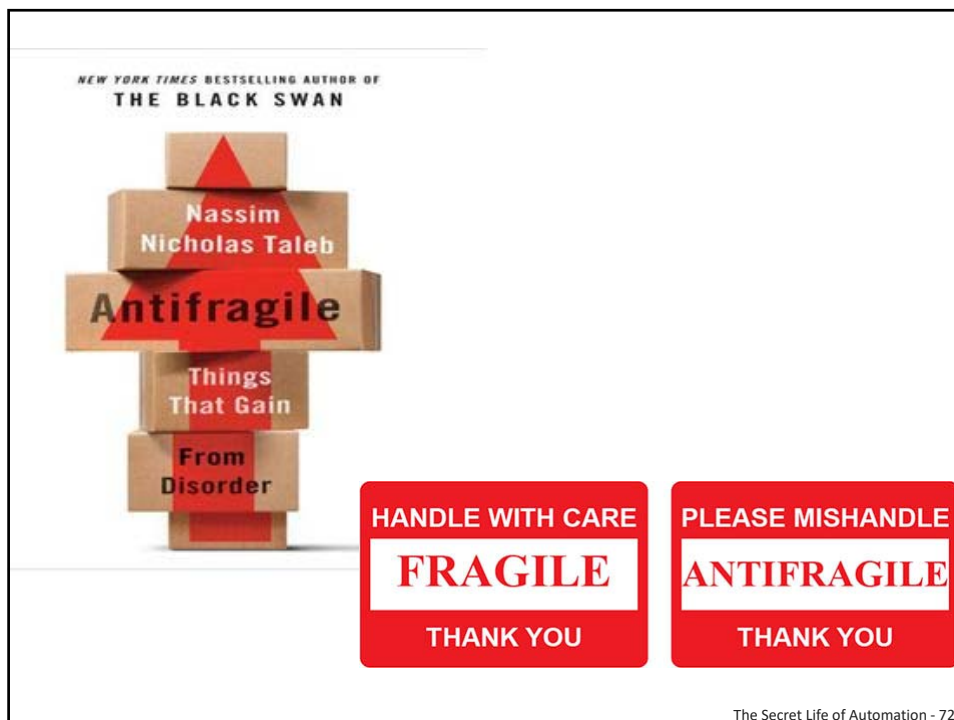
Secret:

Some people try to make testing repetitious and over-focused, and therefore fragile.

The Secret Life of Automation - 70

Secret:
Successful testing should be an
antifragile* and *antifragilizing
activity.

The Secret Life of Automation - 71



The Secret Life of Automation - 72

Secret:
Don't use tools simply to demonstrate consistency (although unit checks might be helpful for developers).

The Secret Life of Automation - 73

Secret:
We don't know how to test until we've *tried* to test.
And we don't know how to apply tools until we've *tried* to apply tools.

The Secret Life of Automation - 74

Secret:

Don't worry about building tools knowing you'll throw them away. Take advantage of the fact that as you're *building* tools, you're testing.

The Secret Life of Automation - 75

Claim:

Automated checking frees up more time for exploratory testing.

The Secret Life of Automation - 76

Secret:
No, it doesn't, really.

The Secret Life of Automation - 77

Secret:
***Execution time* may be reduced, but there's also preparation, (maybe) programming, (maybe) debugging, troubleshooting, maintenance, analyzing failed checks, repair...**

The Secret Life of Automation - 78

Secret:

Preparation, programming, debugging, troubleshooting, maintenance, analyzing failed checks, repair, etc. may afford learning... but will it be about things customers care about?

The Secret Life of Automation - 79

Technology Rule 1:

Any novel, non-trivial task will take longer than you think it will.

The Secret Life of Automation - 80

Technology Rule 2:
**Rule 1 applies even after you've
taken Rule 1 into account.**

The Secret Life of Automation - 81

Technology Rule 0:
There will be bugs.

The Secret Life of Automation - 82

The Programmer's Credo

**“We do these things
not because they are easy,
but because we THOUGHT
they were going to be easy.”**

—Maciej Cegłowski (@pinboard)

The Secret Life of Automation - 83

Secret:

**Too often, tool work becomes
solution-problemming.**

The Secret Life of Automation - 84

**Secret:
We shape our tools; thereafter they
shape us.**

—Marshall McLuhan

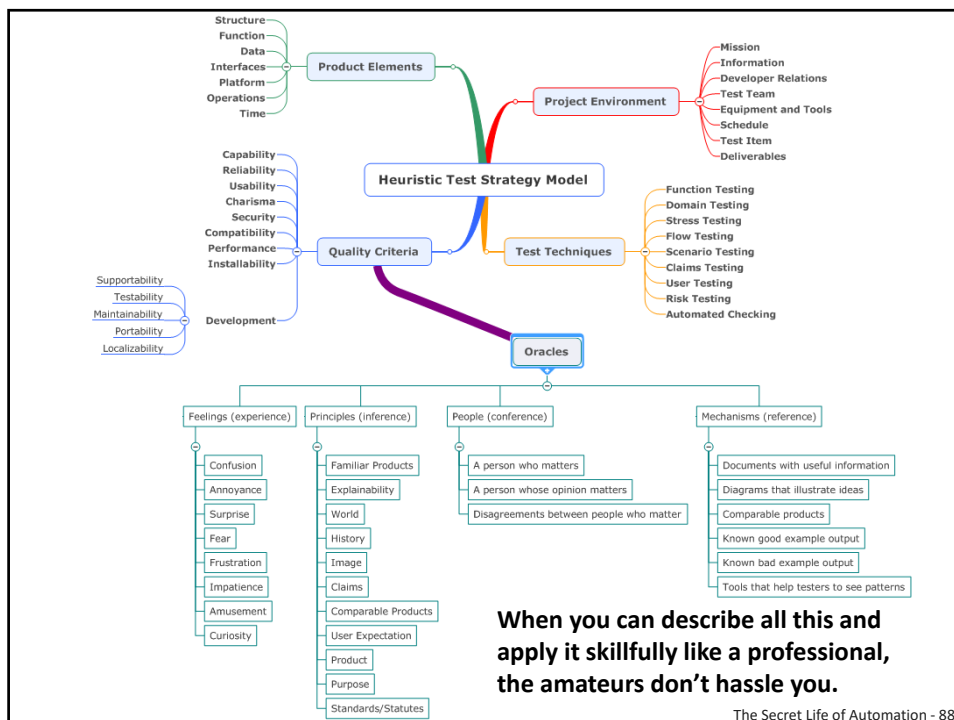
The Secret Life of Automation - 85

**Secret:
You can't schedule epiphanies.
But you can leave yourself exposed
to them, and learn from every one.
Learn from every bug.**

The Secret Life of Automation - 86

Secret:
When you diversify your coverage and risk models (and talk about them), you'll get less pressure to attempt and rely upon confirmatory checking.

The Secret Life of Automation - 87



The Secret Life of Automation - 88

Want to improve your testing in a hurry?

Replace...

- Verify that...
- Validate
- Confirm that...
- Pass vs. fail
- “Automated testing”

With...

- Challenge the belief that...
- Investigate
- Find problems with...
- “Is there a problem here?”
- “Programmed checking” and “tool-assisted testing”

The Secret Life of Automation - 89

More Stuff: Use The Google

- My blogs
 - “On Green”
 - “On Red”
- With James Bach
 - “A Context-Driven Approach to Automation in Testing”

The Secret Life of Automation - 90

Ideas for Teams

- Development tasks *include* testing tasks. It is not wise to isolate development and testing.
 - If you treat outsourcers as outsiders, you WILL fail.
- Testing benefits from diversity and requisite variety. Diversify your team and your tactics.
- Learn from every bug.
- Revisit your checks; analyze their relevance.
- The belief that you MUST automate user-level checks suggests problems in your development process. Fix THEM.

The Secret Life of Automation - 91

Antidotes (for the craft)

- The Secret Life of Automation is that testing, by and large, exists in a constant state of existential crisis. At least at places where it is deemed a 'failure' or a 'cost'. So we choose to automate—but what, precisely are we automating? Is it only GEMPOB?

The Secret Life of Automation - 92

**This talk is based on personal
experience and on interviews with
friendly and generous colleagues.**

Thank you to James Bach, Adam
Goucher, Pete Houghton, Ben Simo,
Andy Tinkham, and Anonymous

The Secret Life of Automation - 93

**This talk is based on personal
experience and on interviews with
friendly and generous colleagues.**

Selection Biases

People who don't need my help don't ask me for help.

People who *really* need help don't ask *anyone* for help.

The Secret Life of Automation - 94