Failure: Why It Happens & How to Benefit from It

LISA16
@vmbrasseur

@vmbrasseur, CC-BY-NC



Introduction

@vmbrasseur, CC-BY-NC

About me

VM (Vicky) Brasseur

Twitter: @vmbrasseur
Freenode IRC: vmbrasseur
Email: lisa2016@vmbrasseur.com

~20 years in tech, most of which involved with open source in some way

Currently a senior software engineering manager at HPE, soon to be SUSE thanks to a recently announced acquisition

Years ago I started studying failure as a bit of a hobby. What follows stems from that research.

[citation needed]

https://www.zotero.org/groups/failure/items

I'm going to say "research shows..." a lot. Here is the research in question.

I'll show this URL again at the end

134 items in that bibliography so far

Thousands of years & we don't seem to have learned much

Thousands of pages of research. I'll try to synthesize it here and share several common themes found in the research:

Common themes

» Factor: Complexity

» Factor: Assumptions

» Factor: Organisation (aka Culture)

»

» Fix: Experiments

» Fix: Introspection

These are the themes which are most common across all of the research, so these are what I'm going to share with you today.

Like I said: thousands of pages. I can't cover it all.

Hopefully there'll be time for questions, so if I don't cover something just ask later.

Speaking of "not covering..."

Won't cover

- Won't give you a silver bullet
- In depth

I will NOT give you the secret serum which will prevent all failure for you from here on out

I WILL give you a list of questions you should consider if you'd like to start making the best of failure, as you should.

I also will only present an overview. Because, um...3000 pages.

Along with these things, I'd like to mention another limitation of this talk...

Western bias to the research

» Western audience/culture

» Western psychology

All of the research I've studied was done under European psychology and primarily on Western subjects

Unrelated recent studies have found (Surprise!) that many Western psychological theories don't apply to Eastern cultures

Would love to see studies (written in English) which study these things in non-Western cultures

Anyway, please keep in mind that there's a Western bias to all this.

TWO SLIDES LEFT IN INTRODUCTION

Slides are already available!

http://archive.org/details/lisa16-failure

I'll show this URL again at the end



@vmbrasseur, CC-BY-NC

Please save 'em

And now, let's get into those common themes I mentioned earlier



Complexity

@vmbrasseur, CC-BY-NC

The world is complex. People are complex. Projects are complex. Heck, the complexity hiding in something as seemingly straightforward as a chocolate chip cookie is astonishing.

We as humans, however, don't seem to like complexity. That leads to some unfortunate tendencies.

Willful ignorance

For starters, we as humans have a tendency to ignore complexity

Everything around us is constantly changing & evolving, yet research shows we prefer to operate as though we're in a static environment.

Start a project -> take a snapshot of the world & operate within that

This, as you can imagine, leads to problems as the rest of the world continues to move forward without us

Dislike variation

Research also shows that we dislike variation. We like things to be Just So and to stay that way.

Therefore we prefer to operate in serial rather than parallel because we like to focus efforts on the "best" option

We prefer to focus on the static now rather than the ever-changing future.

Your task is uncertain because you can't know everything.

That's because we're more comfortable with what we think we "know"

Unfortunately, we don't actually know as much as we think we do

Latent Errors

Because we think we know more than we do, we have a tendency not to look around for those snakes in the grass which might bite us

These are the "Unknown unknowns." It's not just a funny thing a politician said once: It's a legit psychological concept

Overlooked near misses

Look a lot like successes, just because they haven't failed... yet

Enabling conditions

And that's because those latent errors have yet to meet their enabling condition

The trigger which turns a latent error to the dark side

Deepwater Horizon example: Incorrectly followed cementing procedure -> (ergo) Escaping gas.

* Welding nearby, Windless day

* Killed 11, injured 16 more, led to one of largest environmental disasters our country’s ever seen

Post-mortems?

Post-mortems are a great opportunity to look for latent errors

What was successful...pretty much by chance?

But here's the problem...

We kinda suck at post mortems

We focus on symptoms rather than causes

Never just one cause

Selection bias: we look at those things we prefer to rather than the hard stuff

Therefore we have a tendency to postmortem non-representative data sets

We kinda suck at post mortems (continued)

Outcome bias: usually means a positive outcome, but in this case...

Only post-mortem the things which went "wrong"

Actively ignore latent errors (near misses)

Actively ignore successful outcomes

Ignore complex process which led to a successful outcome

The decision process is not the outcome

How can we minimize the impact of these tendencies?

» Pre-mortems

» Plan, but plan to change the plan

Pre-mortem: Look for the latent errors

Get a skeptic in the room & on your team

Get an "outsider" in the room & on your team

Check in from time to time to see whether your plan still holds true (remember: the environment continues to evolve & doesn't wait for you)



OK, on to our next theme: Assumptions

Root of all evil?

Nearly every piece of research points to assumptions as a frequent contributor to failure

This is one of those lessons we never seem to learn for some reason

Research really doen't have very nice things to say about assumptions

Often based on incorrect or invalid information

For starters, assumptions are usually based on incorrect or invalid information (see the aforementioned snapshots of the world)

We then use these assumptions for a basis for a set of heuristics

For example: used to create estimates, which are themselves very wrong

Other popular assumptions: user needs, market questions, "what business are we really in?"

Often stem from poor problem definition

You can't solve a problem you can't define

Therefore assumptions about the problem and its solution are often way off base

Impossible to fail fast if you don't define success and therefore also failure

As you consider the problem, treat each "requirement" as an assumption -> needs to be questioned and confirmed

Unquestioned assumptions become fact in the minds of the hearers

"We assume this will take 2 weeks but we need to confirm" -> No one hears anything after the "2 weeks"

Common in cultures of accepting assumptions rather than questioning them

People claim they don't have the time to verify assumptions, but they do seem to have the time to work on the wrong thing for the wrong people at the wrong time.

How can we minimize the impact of these tendencies?

» Take the time to define the problem

» Perform pre-mortems to list and verify all assumptions

» Document everything

» Revisit assumptions throughout the process

Gather risks & what-ifs

Create contingency plans

Document!

Environment changed, assumption still valid?

Still need to assume? Or assumption converted to knowledge?

Beware confirmation bias: don't look only at the information which will verify the



Organisation (aka Culture)

Businesses. Clubs. Projects. Any organised group comprised of humans.

Because they're comprised of humans, organisations are a major contributor to failure.

This is the big one, so I'm going to spend a fair chunk of time about it

Illusion of control

Ellen Langer: Psychologist at Harvard

We're kinda arrogant, as species go.

We have a tendency to over-estimate the impact our actions can have. We overestimate the things we can control or even just influence.

Therefore, we also have an inflated sense of what leadership and our leaders can do.

Latent errors

Reminder: these are the "near miss" situations. They look like a success because they didn't explicitly fail.

Enabling conditions: Deepwater Horizon. Incorrectly followed cementing procedure -> (ergo) Escaping gas.

* Welding nearby, Windless day
* Killed 11, injured 16 more, led to one of largest environmental disasters our country’s ever seen

If we look for & find them, we have a chance to control latent errors.

We have almost no chance to control enabling conditions. They're out of our hands entirely.

We therefore cannot truly control the outcome of most

How is your company/project/ group organised?

The organisation of your, uh, organisation can have a dramatic effect on its success rate and reaction to failures

Silos?

Define in an organisational sense

Dilute responsibility

Hamper communication

Obscure information

Organised around processes?

Orgs which mirror processes can make it harder to change processes (require a reorg)

Multi-disciplinary teams can help here (also with the silos)

How does your organisation react to failures?

Punishment for failures just compounds the cost of failures

Failures stop being reported

Can't learn from mistakes which aren't reported, and that's really how you reduce future failures: By learning from those which have already occurred.

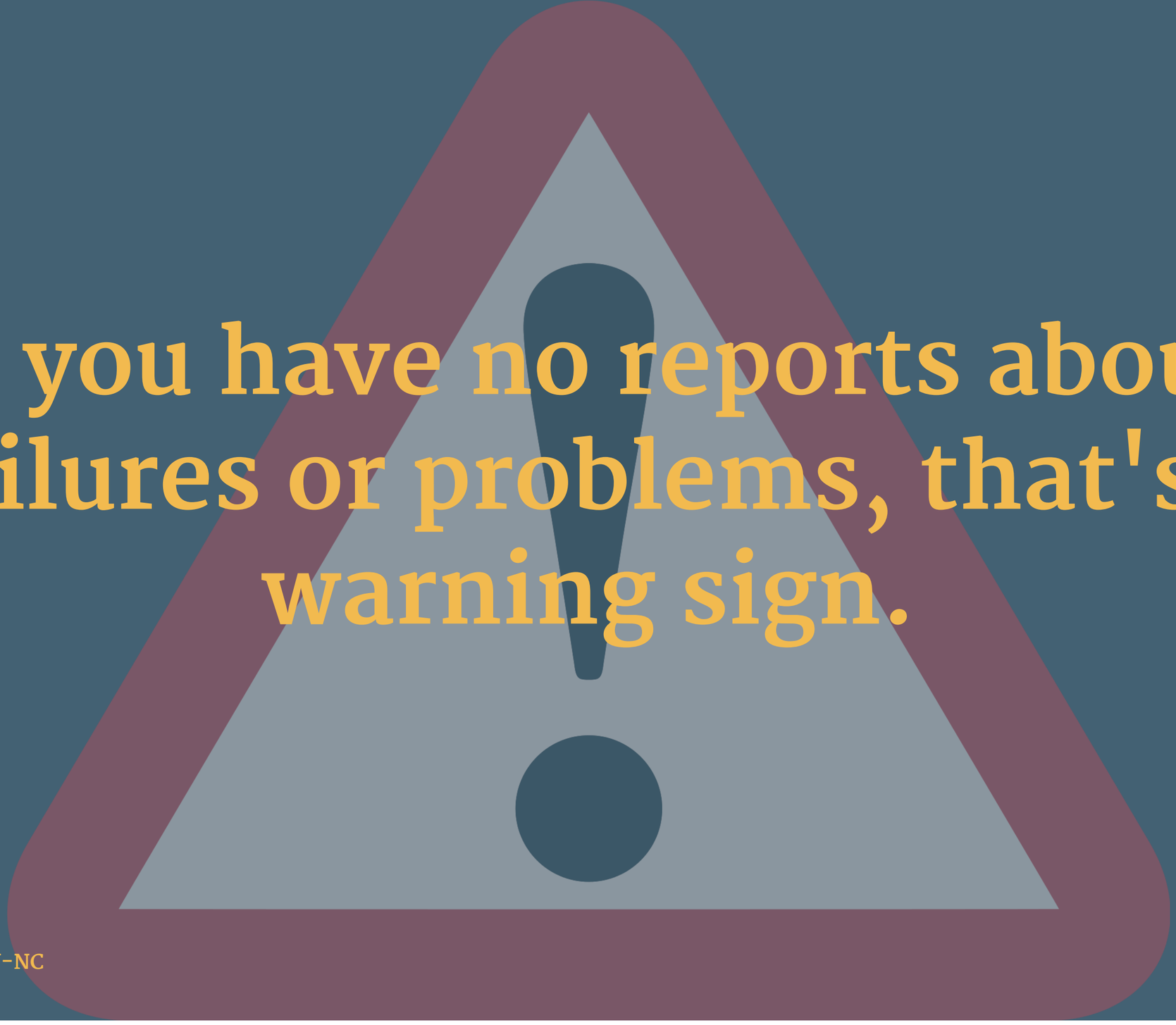
Unshared failures are just experience, not shared learning, and therefore are of low value. That's because experience comes for free. It just happens to us. Learning, however, does not.

People become afraid to try new things: innovation stifled

If people fear being punished for failure, they're not going to risk trying new things.

This can lead to a dramatic stifling of innovation in processes, ideas, technologies.

Your organisation could stagnate and potentially even die.



@vmbrasseur, CC-BY-NC

It doesn't mean they're not happening. It just means they're being hidden, which is much worse.

Normalisation of deviance

Diane Vaughn: American sociologist; studied the Space Shuttle Challenger explosion

An example of latent errors, hiding right out in the open.

Actively ignore latent errors.

Example: log files

Just one race condition away from catastrophe.

More pressure/less time -> more errors & overlooked latent errors

We have an unfortunate tendency, particularly in tech, to push ourselves too hard. To try doing too much in too little time. To put in long hours, often unnecessarily.

This leads to a lot of problems. The obvious ones are fatigue-driven errors, but there are more subtle and dangerous problems lurking there.

Rely more on rules of thumb & untested assumptions

Inability/reluctance to pull the plug

Remember back when I mentioned that we like to focus on just one thing? Work in serial rather than parallel?

Related to that, we have a hard time NOT seeing things through to the "end."

Sunk cost fallacy

Project champions: Reality distortion field

* Group think

* Deeply held belief that it will succeed: BLIND FAITH

* Lead us to continue to believe things even when facts show they're just not true

Just too nice: Don't want to hurt anyone's feelings.

Stems from us not learning proper communication skills

What to do about it

Organisational changes can be very difficult to start and harder to finish

All of these solutions are much easier said than done, but well worth the effort.

Suggestion: Leading by example

Any organisational change must be strongly supported by the leaders

Leaders must discuss own failures & what they learned

Suggestion: Develop a culture of psychological safety (1/2)

A mistake isn't a mistake until someone names it so. Before that it's experience.

"A mistake is in the eye of the beholder"

Experience is automatic. Learning is not.

* Develops intuition & skill

* Example: Very senior technical staff aren't very senior because of their age. It's because "they've seen some shit, man." They've experienced failure and have learned from it, developed intuition & skill.

* Teaches you what doesn't work so you can avoid future failure

Suggestion: Develop a culture of psychological safety (2/2)

due to an intolerant environment.

Studies are finally starting to come out showing that psychologically safe environments are the most productive, the most innovative, the most cost-effective.

Example: blameworthy stats

* Asked a lot of CEOs about blameworthiness.

* Actual blameworthy failures: 2-5%

* Percent treated as blameworthy: 90%

We need to root out that sort of attitude and replace it with one which recognizes and embraces the value

Suggestion: Include truth-seeking people

Someone who questions "we've always done it this way"

Designated "project skeptic"

"Outsiders" in post- and pre-mortems

Bring on truth-seeking people

Suggestion: Make projects more survivable

Don't start them at all

Fund/approve only to checkpoints

Spin them off



@vmbrasseur, CC-BY-NC

OK, let's talk about how to make failure work in your favor, starting with experiments.

Experiments are controlled failure, and experiments are great.

We literally would not be here without experiments.

Evolution requires failure

In life forms as in all things: No change is possible without some failures

Evolution is a constant series of "works for now" solutions

Experimentation should similarly be continuous if you'd like your business/project to evolve

Successful experimentation requires accepting failure

Redefine "success" to "learning opportunity"

Seek truth first and "success" second

Experiments can be risky, but that risk can be balanced

Portfolio of high risk & sure things

^ Invest for the long term, not the short term

Nature of successful experiments

- Small & survivable
- Expectation some attempts will fail
- Explicit success and failure criteria
- Have an exit strategy

don't really understand what that entails

* Read the next two bullet points

Keeping experiments small means actions are closely situated to outcomes -> easier to determine cause & effect

Experiment where you're already familiar w/some of the variables so you're not starting completely from zero w/learning

Know when you've failed

^ Test often

^ Be willing to pull the plug

Introspection

So! You want to get better and failure and learn from it.

The best way to do that is to start talking openly about it and inspecting your failure environment

What sort of questions should you be asking of your project/organisation?

Questions to ask

- What are we actually trying to accomplish?
- What is our end user actually trying to accomplish?
- What are our assumptions?
- Do our assumptions still hold true?
- What are the possible latent errors?

Questions to ask (continued)

» What does "failure" look like?

» What is our exit strategy?

» What is our postmortem process?

» How does our organisation view & treat failure?

» Do we have a culture which provides psychological safety?

The answers to these questions will be different for each project, each organisation

Therefore the solutions will also be different for each project, each organisation

There is no silver bullet

Wrap it up

Those links again

Slides: http://archive.org/details/lisa16-failure

Bibliography: https://www.zotero.org/groups/failure/items

Twitter: @vmbrasseur
Freenode IRC: vmbrasseur
Email: lisa2016@vmbrasseur.com

I'll leave this up while we do Q&A