



It's  in a fancy suit



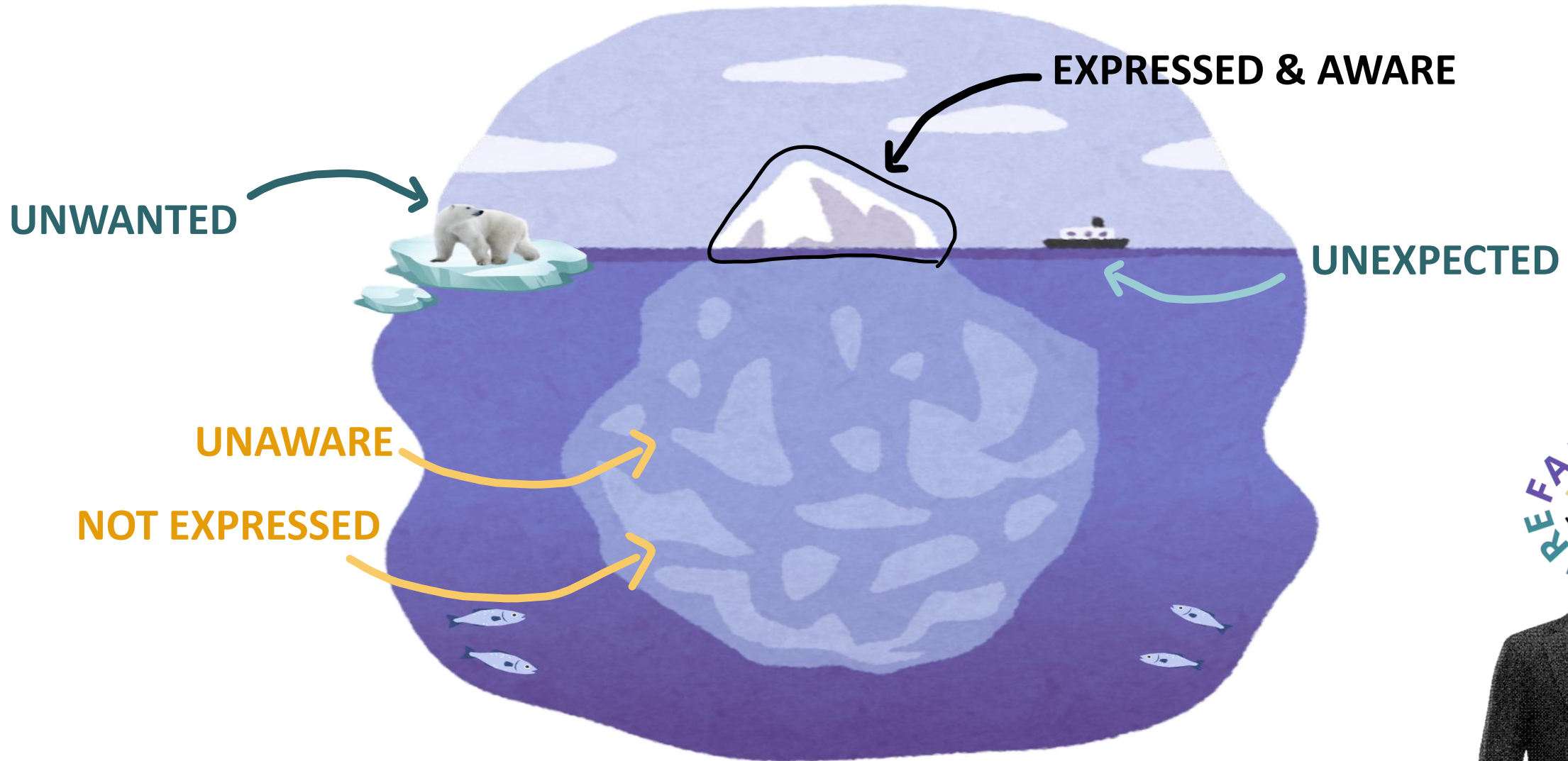
Lena Nyström

ASSUMPTIONS

- We want it to stay the same
- We know what it does, and what it does is correct
- We'll save time and money using it as blueprint



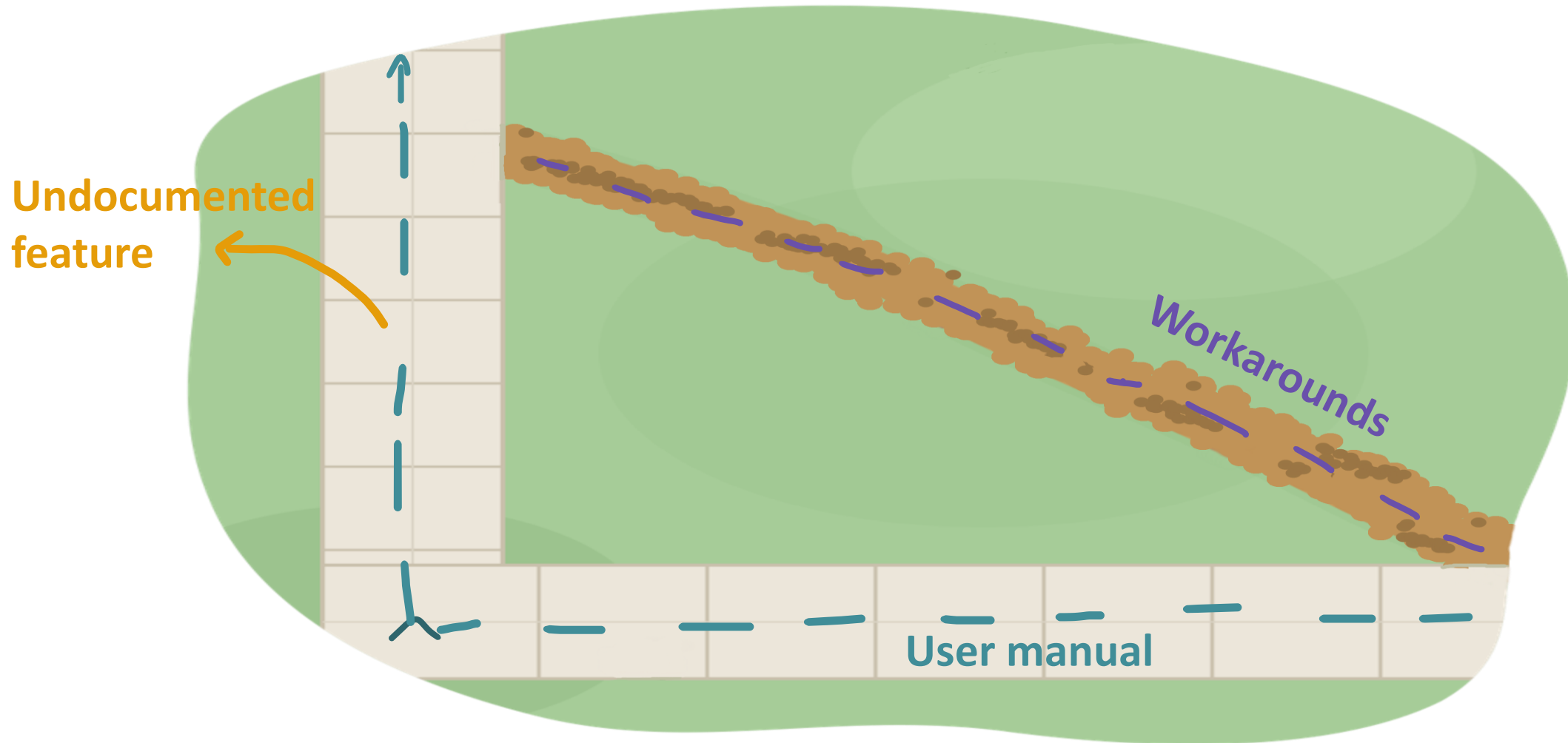
We want it to stay the same



FAILURE
FAILURE
FAILURE
FAILURE
FAILURE
FAILURE
FAILURE
FAILURE



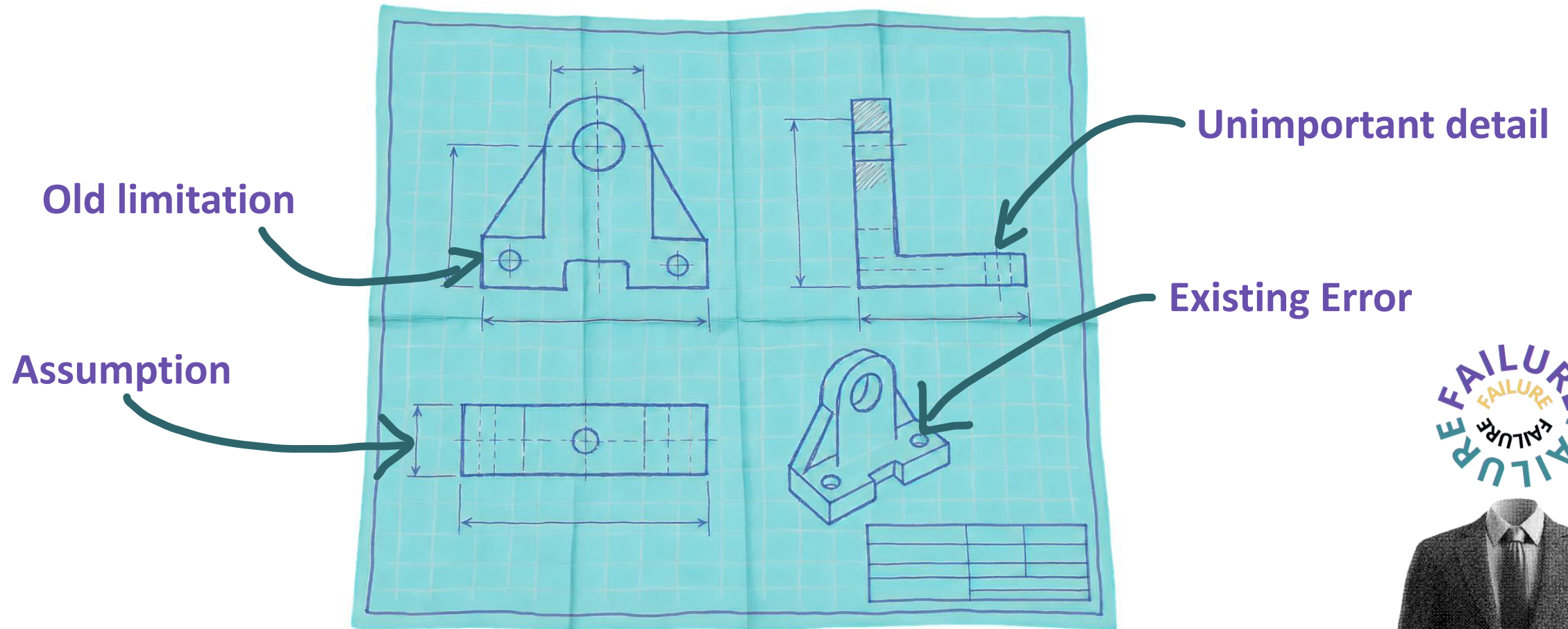
We know what it does, and what it does is correct



FAILURE
FAILURE
FAILURE
FAILURE
FAILURE



We'll save time and money using it as blueprint



FAILURE
FAILURE
FAILURE
FAILURE
FAILURE



Project 1:
the code is the oracle

PROJECT 1: THE CODE IS THE ORACLE

- **We want it to stay the same**
- We know what it does
- We'll save time using it as blueprint



PROJECT 1: THE CODE IS THE ORACLE

Questions I should have asked

- What are your main work processes?
- What's not working well today?
- Which are the most important parts?



PROJECT 1: THE CODE IS THE ORACLE

The Oracle Problem

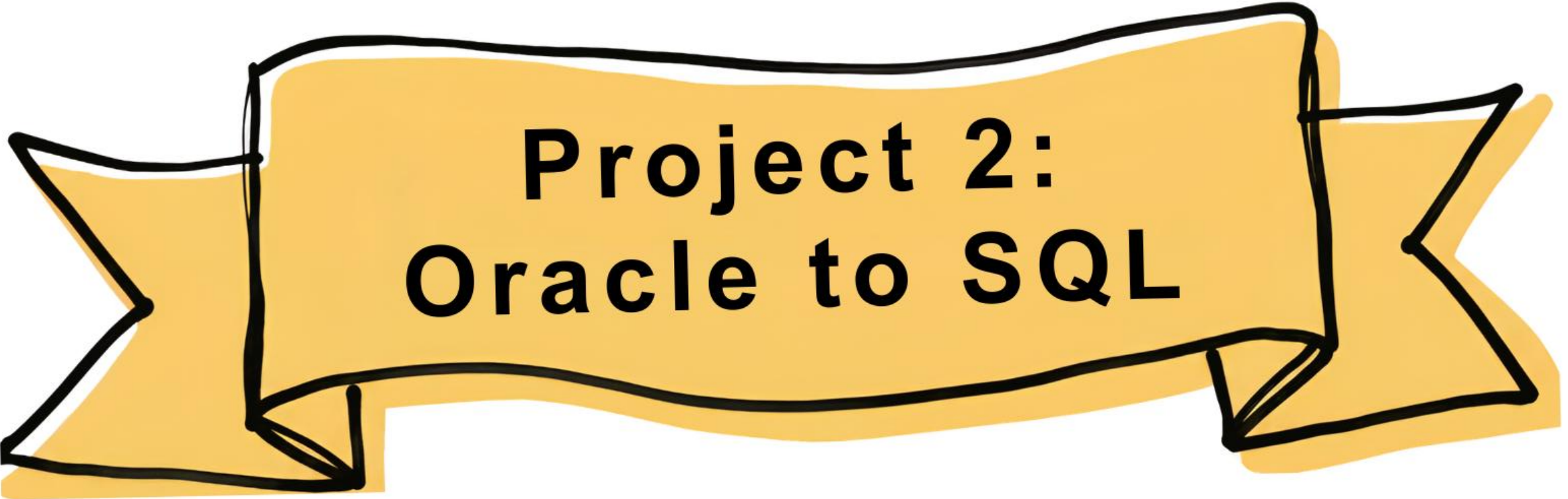
“What exactly am I supposed to verify?”

“What is important?”

“What else am I missing?”

If we don't specify what we expect and take the time to do that early – we don't save time. We just push the cost downstream.



A yellow banner with a black outline, featuring a wavy, ribbon-like shape with pointed ends. The banner is centered horizontally and contains the text "Project 2: Oracle to SQL" in a bold, black, sans-serif font.

Project 2: Oracle to SQL

PROJECT 2: ORACLE TO SQL



- We want it to stay the same
- **We know what it does**
- We'll save time using it as blueprint



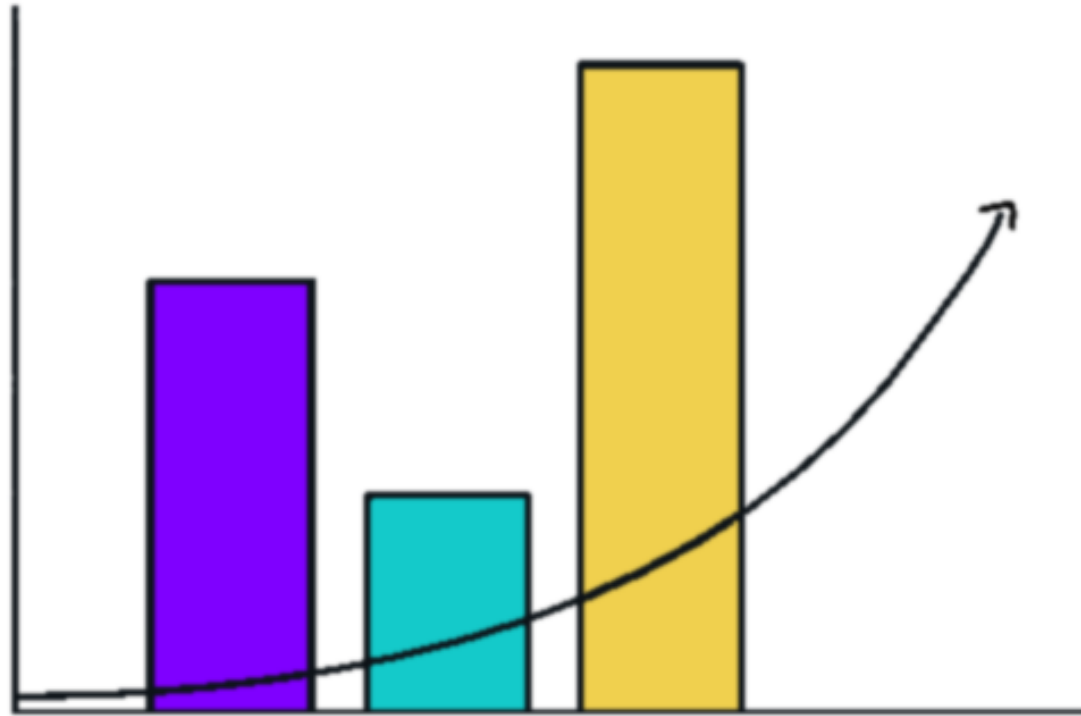
PROJECT 2: ORACLE TO SQL

Important questions I did ask

- Why does the documentation and the code differ?
- What is not the same between Oracle and SQL Server?
- It will take months more to rebuild this bug, do we really want that?



Cost of an early mistake doesn't go away
Coding cost has flattened, **discovery** cost has not



Project 3: System Switch



**TEST
SCOUTS**

PROJECT 3: SYSTEM SWITCH

- We want it to stay the same
- We know what it does
- **We'll save time using it as blueprint**



PROJECT 3: SYSTEM SWITCH

Important questions I did ask, but too late

- In what ways does the new system differ from the old?
- Do we understand what we are supposed to test, and how?
- Who is responsible for what?
And how do we work together?



Not a new story



MY MAIN LEARNINGS

- Don't assume the smart questions have been asked
- You don't save time by skipping steps
- A blueprint is not the same as the truth



HOW TO BREAK THROUGH

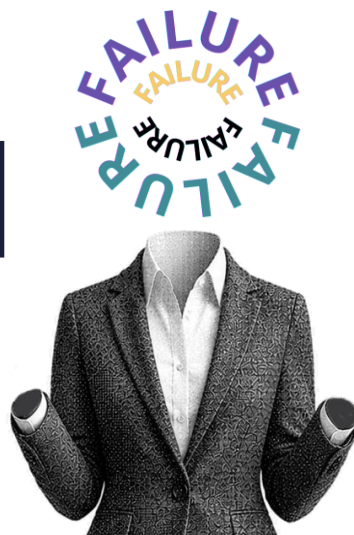
- Ask “how should this work” not “what does it do now”
- Find who owns the workaround(s)
- Build your tests around intended behavior

FAILURE
FAILURE
FAILURE
FAILURE
FAILURE





It's a signal to stop and think
To find what questions are unasked





“Everything should work as before”
is simply not true. If we wanted that,
we could simply do nothing, for free.



Thank You!

lena.nystrom@testscouts.se
<http://testing.pejgan.se/>

