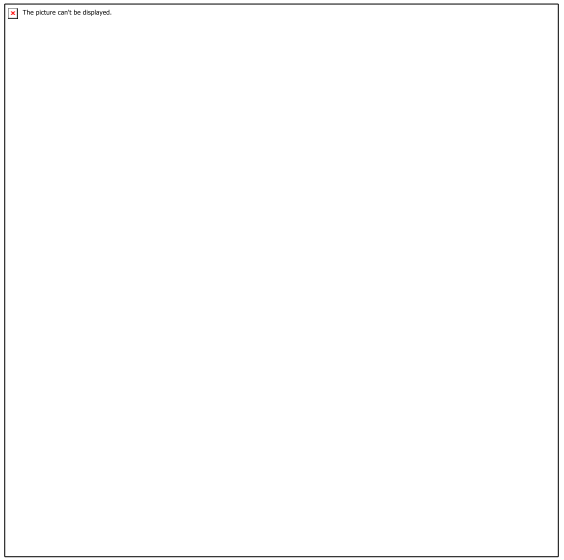
How AI is influencing Test Automation

www.testim.io

Daniel Gold



- Head of QA & Automation at Testim.io
- Blogger
- Lecturer & Speaker

daniel@testim.io

www.danielgold.net

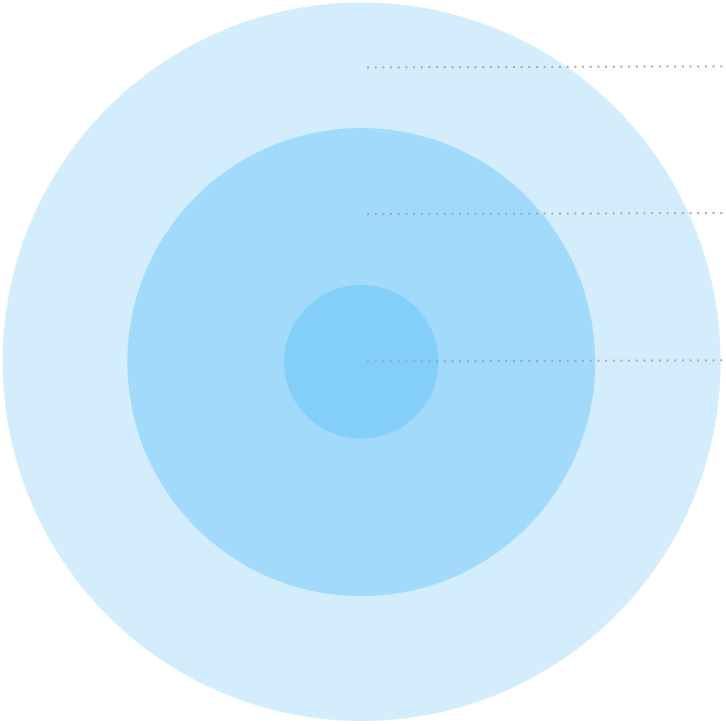
AI
The Source of all evil

AGENDA

- THE BASICS OF AI
- CURRENT TEST AUTOMATION CHALLENGES
- HOW AI HELPED US SOLVE THESE CHALLENGES
- THE FUTURE OF AI IN TESTING

BASICS OF AI

DE-MYSTIFYING AI



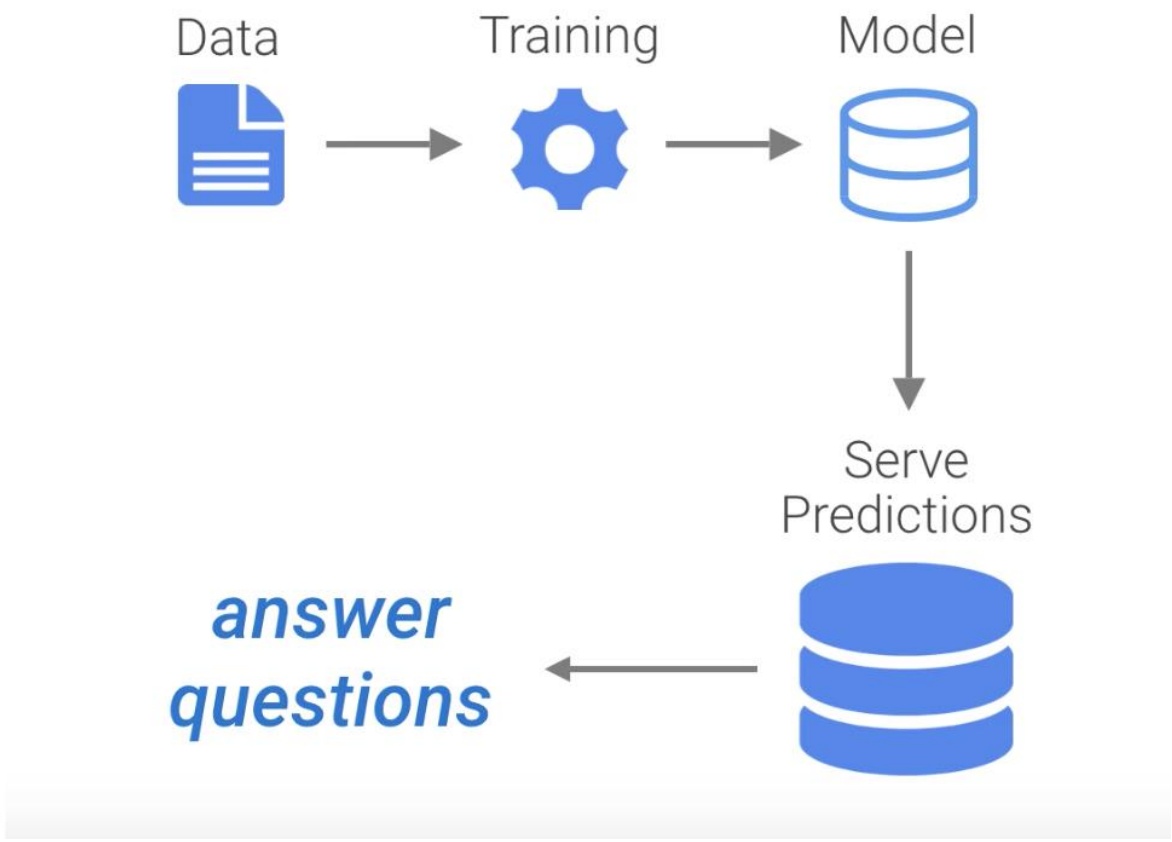
Artificial Intelligence (AI)

Machine Learning (ML)

Deep Learning (DL)

A Few Key Elements

1. Training Data
2. Data Collecting Pattern Recognition
3. Tagging
4. Input Output
5. Confidence Level


Data
Training
Model
Serve
Predictions
answer
questions

Good Boy – Good Output

Supervised Learning

[Courtesy of Dean Pomerleau]

DEMO: QUICK DRAW

CURRENT CHALLENGES

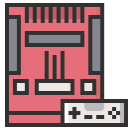
EVOLUTION OF TESTING

Manual Testing

1980 - 1990

Waterfall methodology

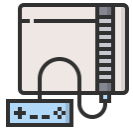
Bulky Automation Tools



1990 – 2003

Experimentation with Different Development Approaches

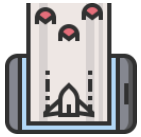
More Robust Automation tools + Open Source Frameworks



2003 - 2010

Agile Approaches, faster release cycles

More about Scale



2010 - CURRENT

DevOps, Continuous Testing, CI/CD

Autonomous Testing, Machine Learning and AI

THE FUTURE

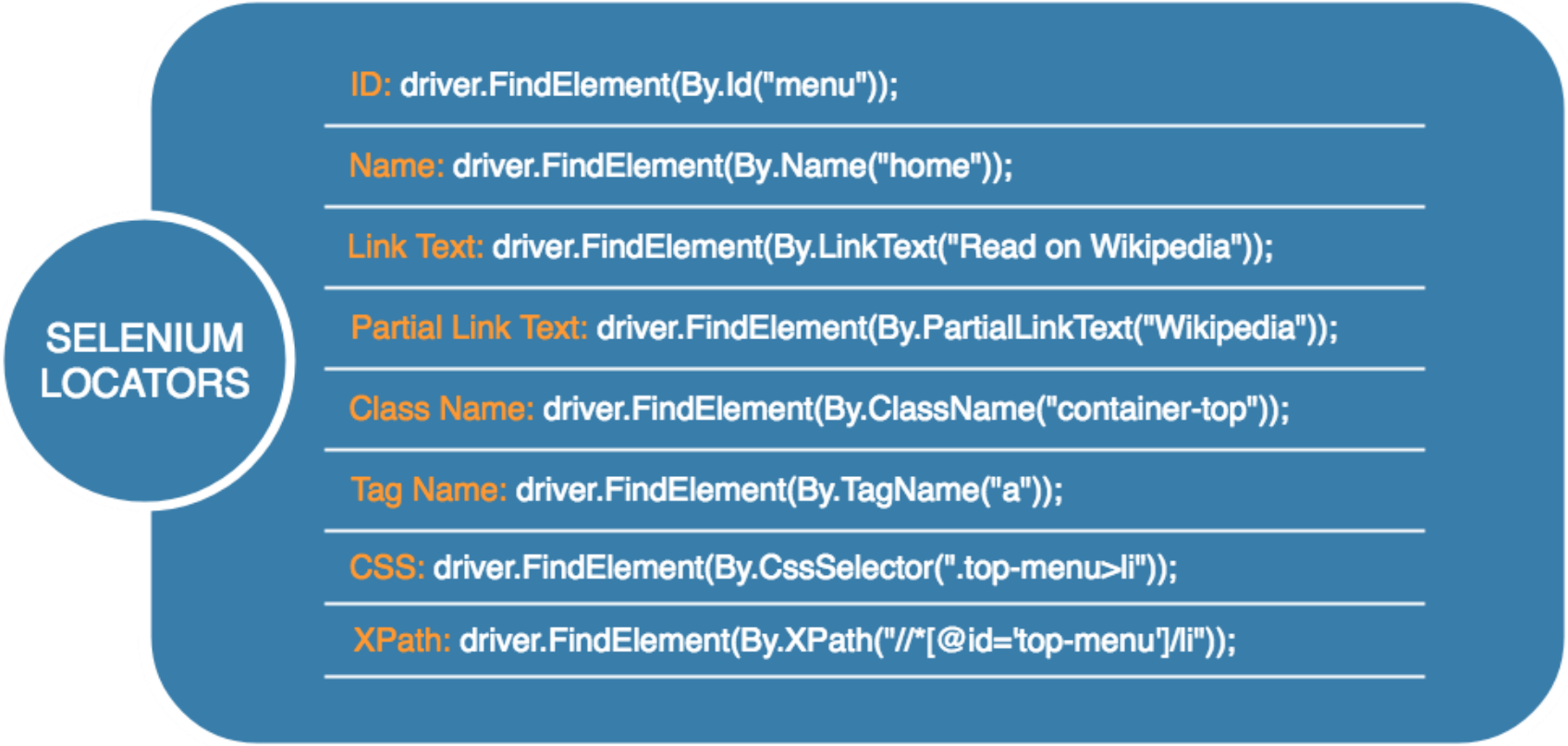
Collaborative, Smart testing

CHALLENGES

1. SKILL SET
2. AUTHORING Time
3. Correct Planning
4. MAINTENANCE
5. SCALE

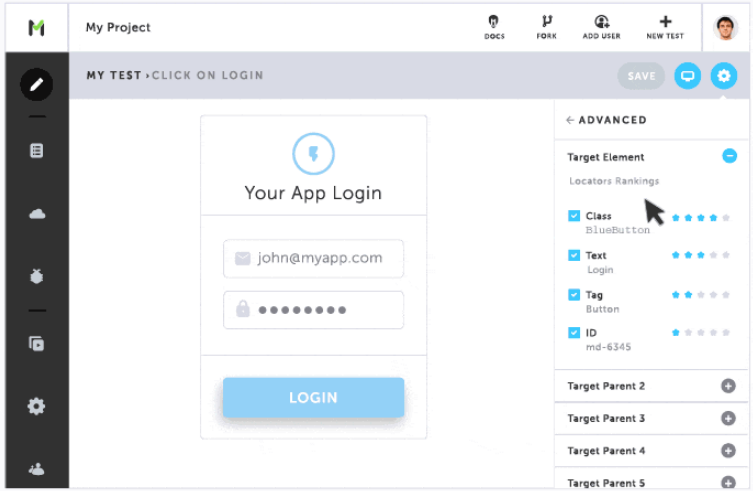
MAINTENANCE

Let’s start by understanding How test automation tools locate elements in your DOM


SELENIUM LOCATORS
ID: driver.FindElement(By.Id("menu"));
Name: driver.FindElement(By.Name("home"));
Link Text: driver.FindElement(By.LinkText("Read on Wikipedia"));
Partial Link Text: driver.FindElement(By.PartialLinkText("Wikipedia"));
Class Name: driver.FindElement(By.ClassName("container-top"));
Tag Name: driver.FindElement(By.TagName("a"));
CSS: driver.FindElement(By.CssSelector(".top-menu>li"));
XPath: driver.FindElement(By.XPath("//*[@id='top-menu']/li"));

HOW AI CAN SOLVE THIS CHALLENGE

AI Based Locating



- **MULTIPLE ATTRIBUTES VS. SINGLE ONE**

  Hundreds of attributes used to identify elements
  A few changes don't break the test

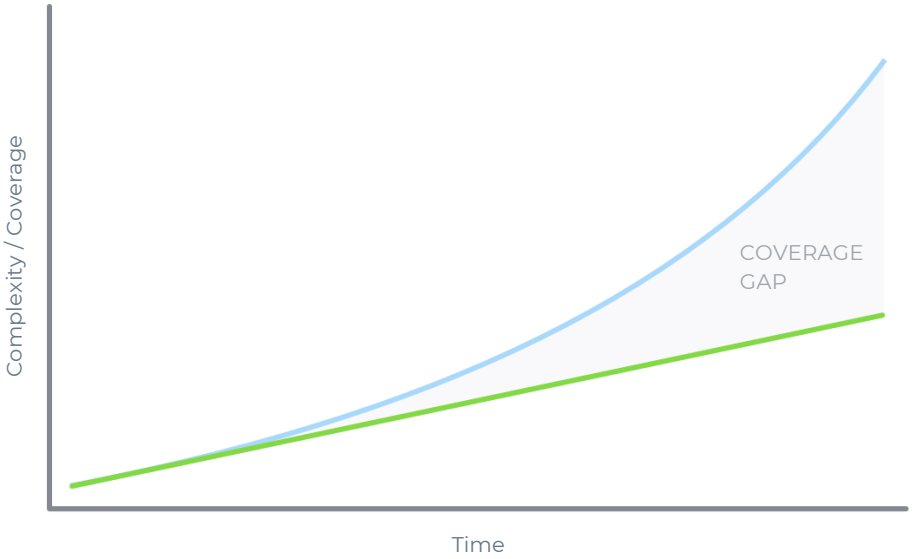
- **LOW MAINTENANCE**

  No need to (tediously) update locators

- **MORE STABLE**

  With each execution we learn the changes to the application and optimize scores

THE COVERAGE GAP IN SOFTWARE TESTING



FEATURES

Complexity increases as new features and states interact with existing features

TESTS

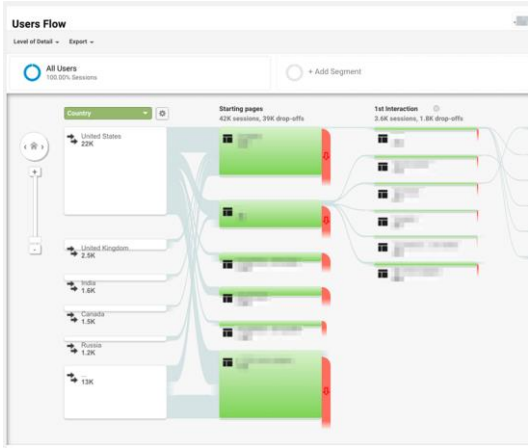
Test coverage grows linearly because they can only be added one at a time

Source: Jason Arbon

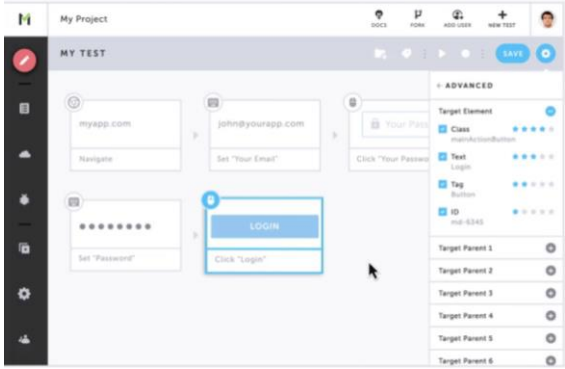
FUTURE OF AI

AUTONOMOUS TESTING

LEARN BY OBSERVATION (PRODUCTS)



AGGREGATE USER ACTIONS INTO FLOWS



TEST PRODUCED FROM FLOWS

TALK SUMMARY

Email: daniel@testim.io | Website: www.danielgold.net |

IT'S A WRAP