

Salesforce Apex Hours



Automate UI Testing with AI and UTAM

Speakers

Christopher Ramm & Sebastiano Schwarz

Venue/Link

[Bit.ly/ApexHours](https://bit.ly/ApexHours)



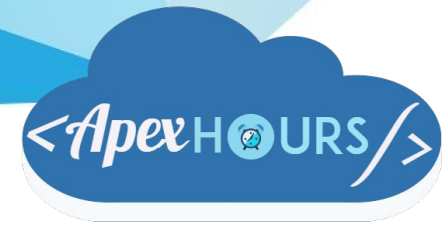
#SalesforceApexHours



www.apexhours.com

 **#SalesforceApexHours**

Our Speakers



Christopher Ramm

DCX CTO Germany |
Salesforce CTA & MVP
at Capgemini

christopher.ramm@capgemini.com

cramm.dev/



Sebastiano Schwarz

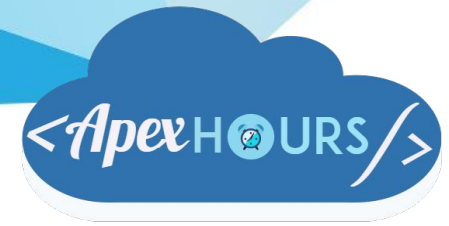
Salesforce CTO Germany |
Technical Architect
at Capgemini

sebastiano.schwarz@capgemini.com

svierk.github.io/



Agenda



01 UI Test Automation
and its Challenges

02 UTAM - One Step in
the Right Direction

03 Efficiency Drains in
the Testing Lifecycle

04 How AI can Further
Improve the Test Process

05 Demo Time - Don't
Tell Us, Show Us

06 Our Learnings & Key
Takeaways



Unit tests alone are not enough

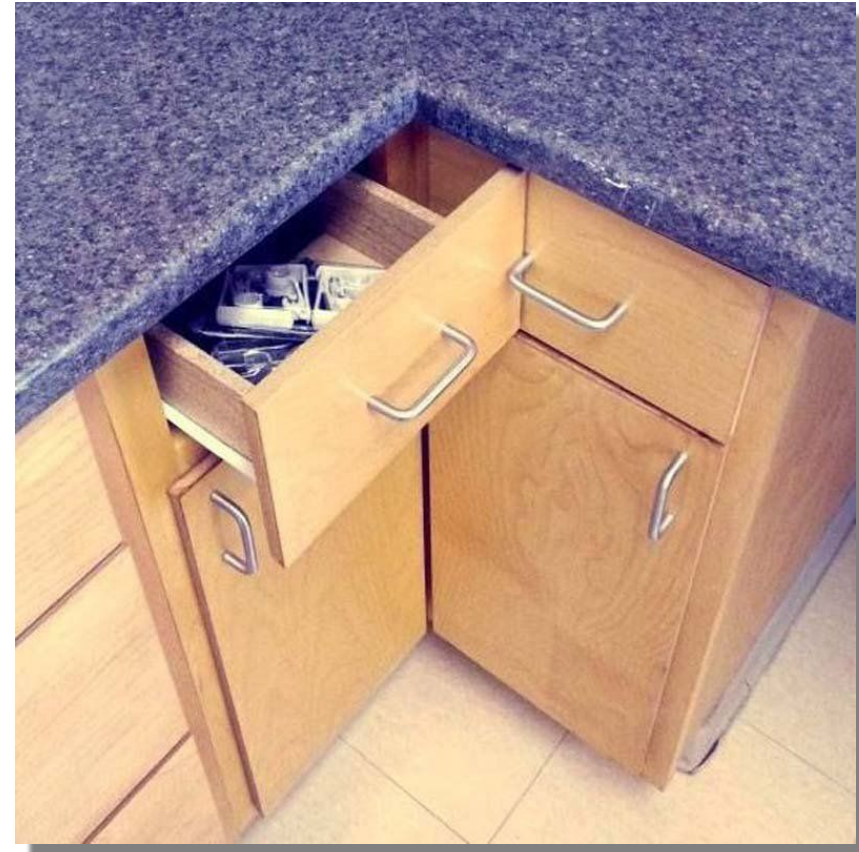


Why End-to-End Test Automation Matters

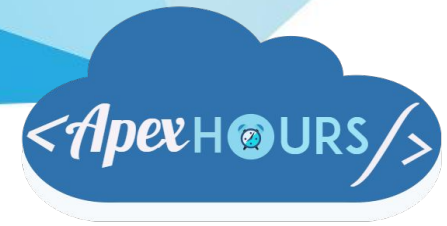


Unit Tests:
Isolated Components
Pass ✓

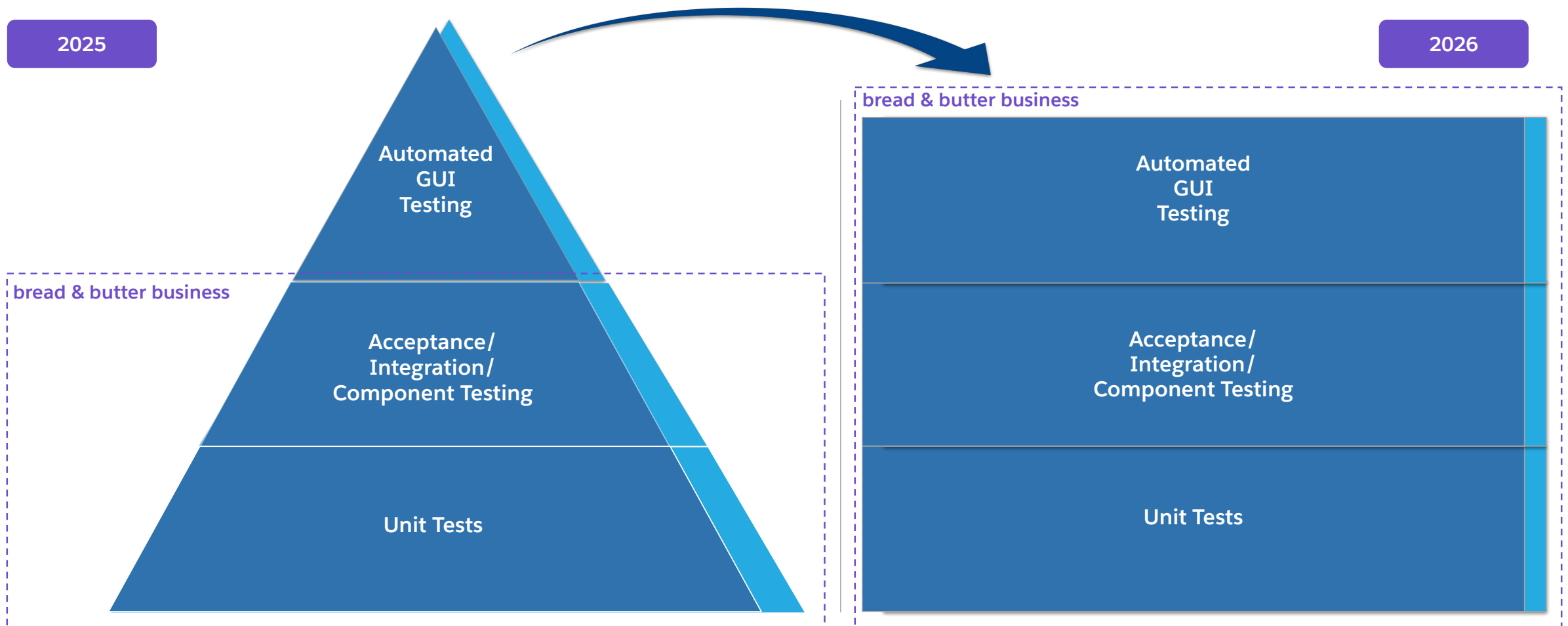
Integration
Failure



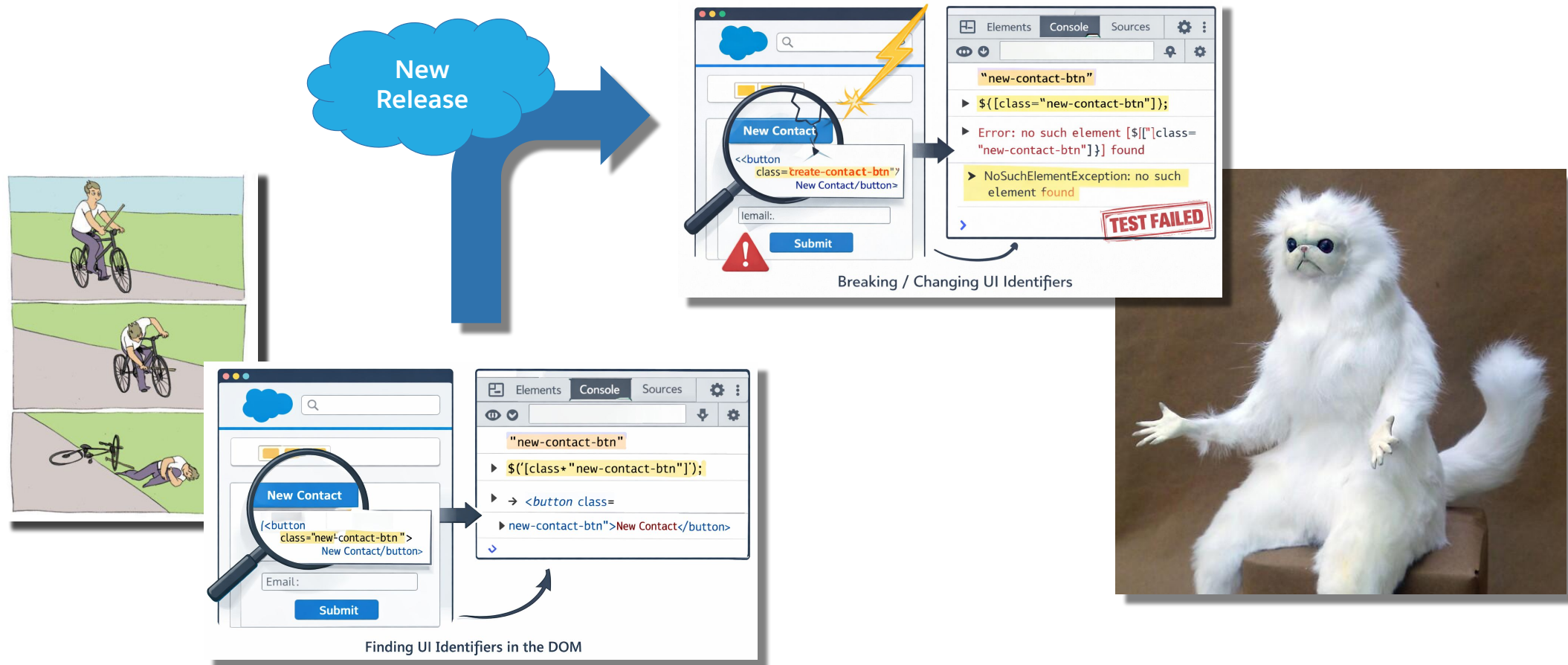
AI makes GUI testing economically viable



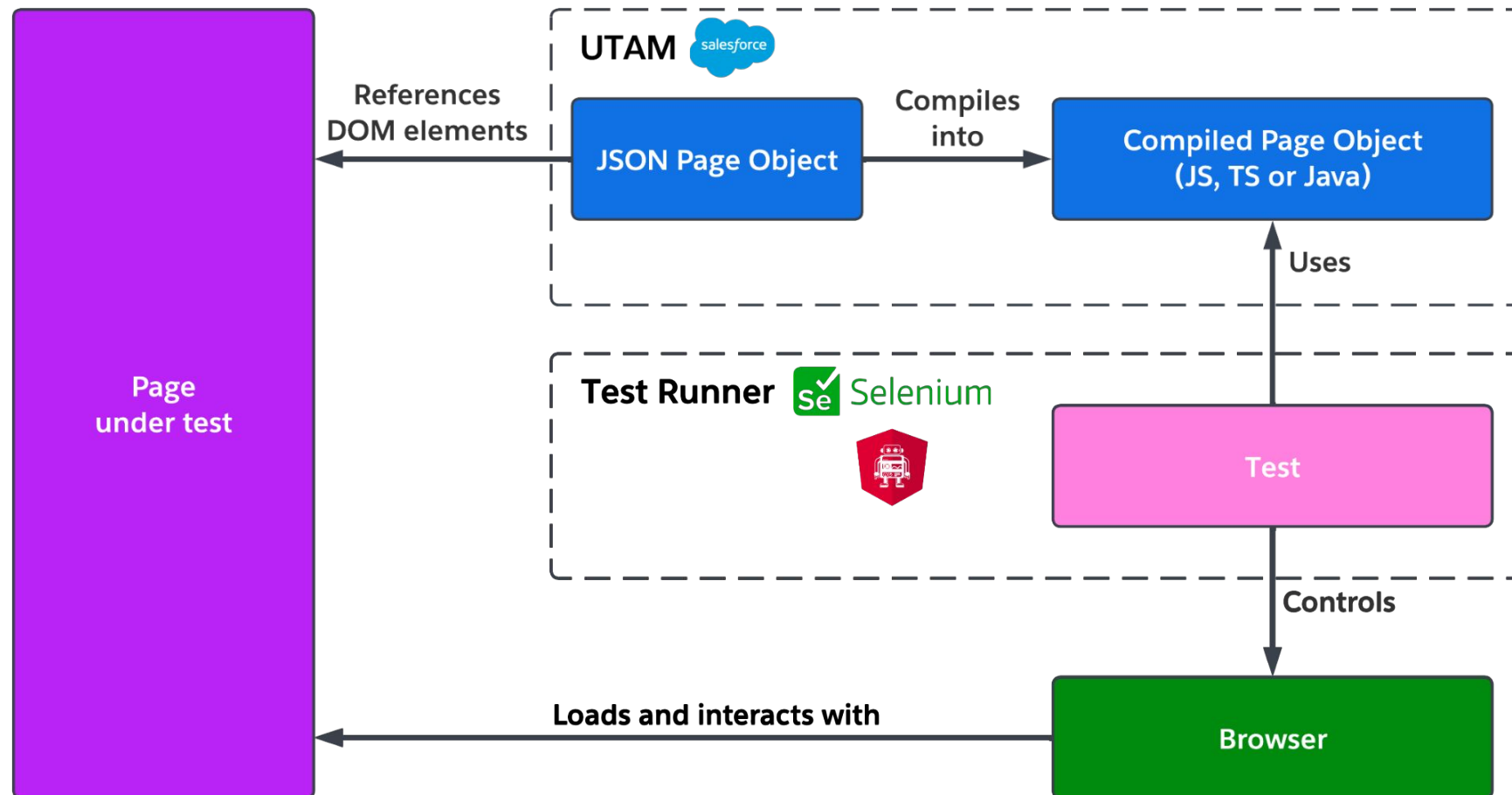
From Pyramid to Platform



Why UI Tests Break With Every Release



UTAM - UI Test Automation Model



Benefits that Make UTAM an Accelerator

Improved Test Stability

UTAM abstracts different UI elements into page objects with stable locators, decoupling tests from the DOM. This reduces the amount of flaky tests.

Agnostic Test Design

UTAM tests are decoupled from the test framework, allowing Salesforce to evolve UI technologies without breaking existing tests.

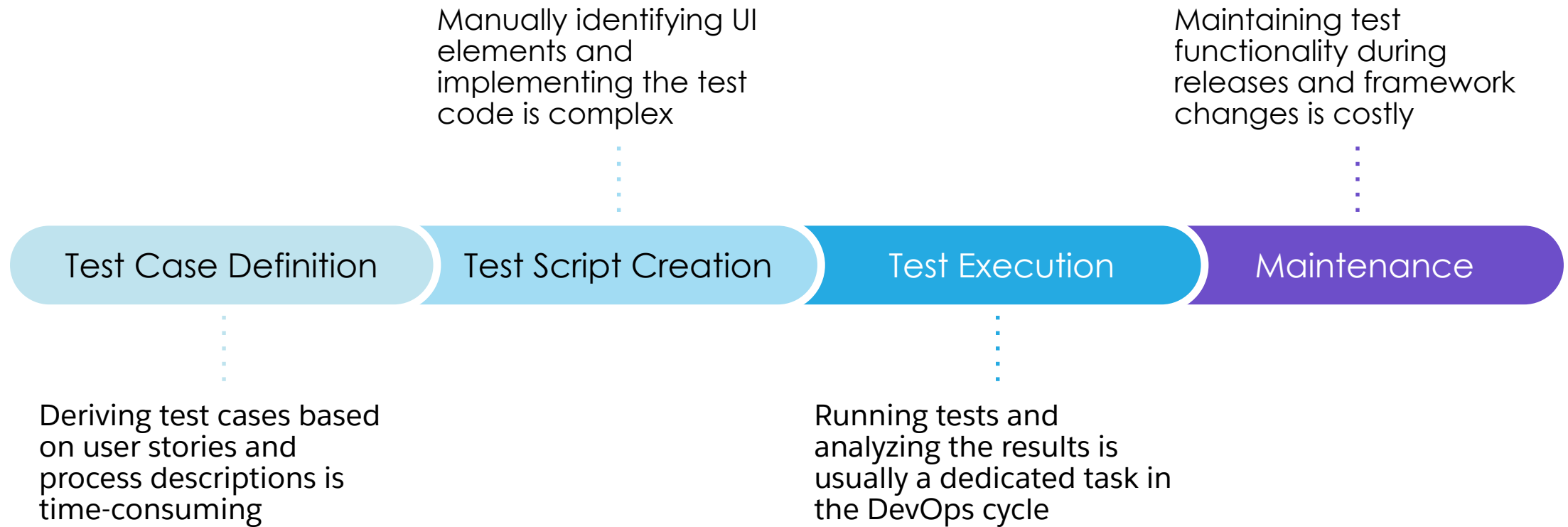
Prebuilt, Reusable Page Objects

UTAM uses standardized, page object definitions for reuse, lower maintenance, and less duplication across teams and projects.

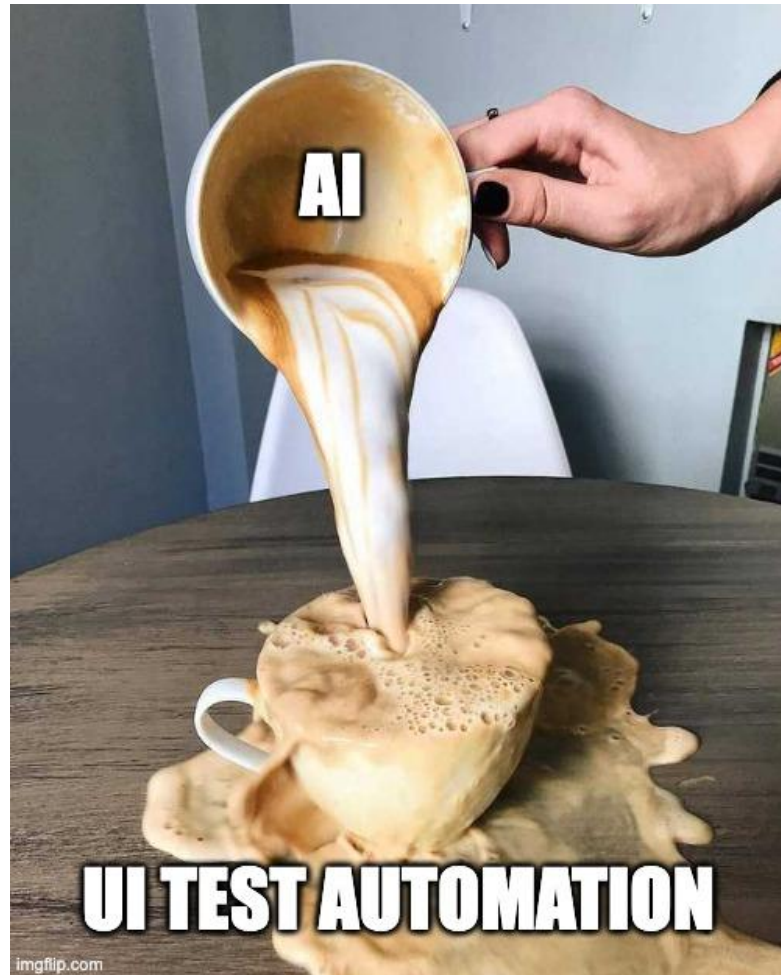
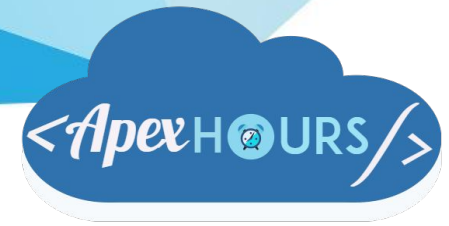
Cross-Platform-Support

UTAM supports mobile and non-Salesforce apps, enabling consistent and scalable UI test automation in large enterprises.

Efficiency Drains in the Testing Lifecycle



Just Add AI - What Could Go Wrong?



How AI Is Transforming the Test Process



Intelligent test generation

AI generates test cases for meaningful coverage from requirements and user stories



Self-healing test automation

uses AI to automatically adapt locators at runtime when UI changes



Risk-based test prioritization

Run the most critical tests first instead of waiting for full regression



Validation of AI-generated code

Testers must validate that AI-produced code meets production standards earlier in the process

Demo

Don't tell us, show us.



***UTAM stabilizes, AI
accelerates** - together they
enable sustainable
Salesforce test automation.*

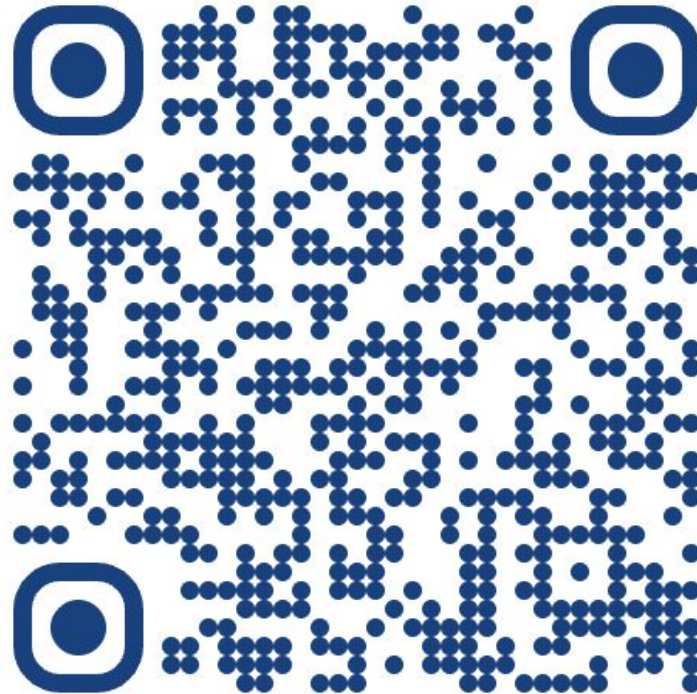
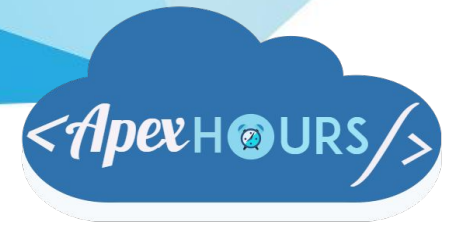




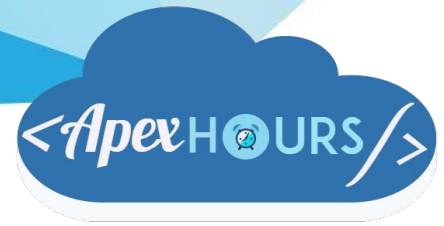
THANK
YOU



Resources



Follow us



#SalesforceApexHours @ApexHours



www.ApexHours.com

<https://www.linkedin.com/company/apexhours/>



Bit.ly/ApexHours



www.ApexHours.com

 @ApexHours