How AI & Shift Left Can Help Transform Your Enterprise Testing

Today's Speakers

Guy Arieli

QA CTO Continuous Testing

Digital.ai

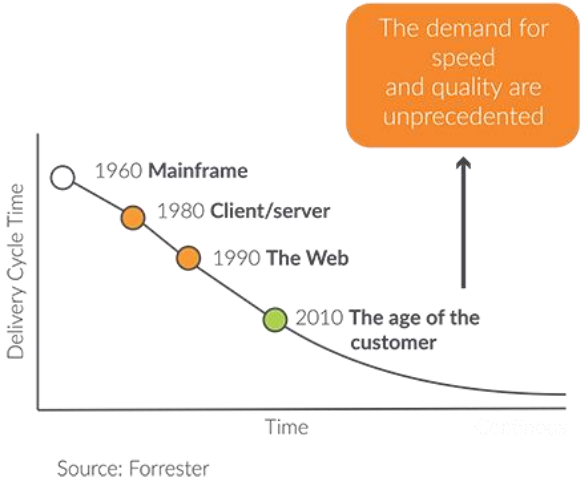
Juan Lugo

Product Marketing Manager

Digital.ai

The Shift Left Movement

The Evolution of Software Development



Source: Forrester

What's Driving Demand for Continuous & Automated Testing?

Test earlier, better, faster to keep pace with modern delivery and optimize digital experiences

Higher Release Velocity

As speed of delivery increases, manual testing approaches don't cut it: costly and time-consuming

Delayed Test Feedback

Testing at the end of the software lifecycle delays time-to-market, and increases risks, and costs

Inefficient Environments

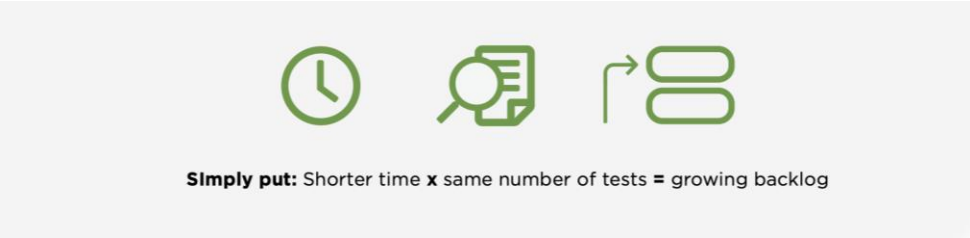
Existing environments unable to support expanding test surfaces and distributed team collaboration

> **Companies are trying to build better software faster to support digital acceleration/digital transformations, and that makes testing a first-class citizen in the realm of digital.**

–Diego Lo Giudice, VP Principal Analyst at Forrester–

Enterprise Testing Challenges

1. Strained collaboration and communication in remote working groups
2. Balancing process execution and new initiatives
3. Increased demand for continuous quality and continuous release
4. Delayed testing feedback
5. Delivery bottlenecks



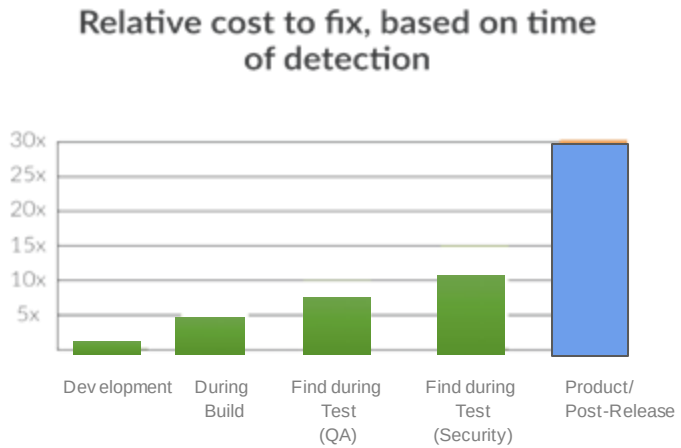
What's Driving the Shift Left Movement

The demand for faster software releases has created new challenges for those that fail to test earlier and more frequently in the Systems Development Life Cycle (SDLC).

The Cost of addressing bugs post-release include:

1. **Financial Setbacks**
2. **Brand Damage**
3. **Customer Retention**
4. **Regulatory Compliance**

Bugs found in early stages of SDLC cost on average $25-$80[1] to fix. When bugs are found post-release, this increases to $7,500-$16,000[2].



1. https://www.bmc.com/blogs/what-is-shift-left-shift-left-testing-explained

Challenges & Benefits of Shifting Left

Shift Left Challenges

- Teams are usually required to update (rework) legacy code
- Hard to introduce to projects with big technical debt
- Varying levels of coding Knowledge/capability between R&D and QA teams
- Highly dependent on the team culture

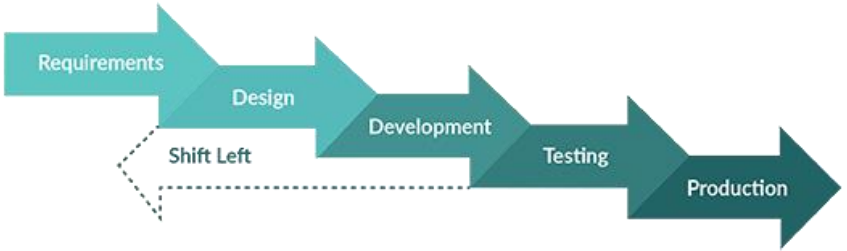
Shift Left Benefits

- Organization Efficiency
- Team Productivity
- Release Cycles
- Bug Mitigation (Post Deployment)

Shift Left in Practice

Introducing quality and automation early to the development process requires some changes including:

- Roles and responsibilities
- Skill sets
- Processes
- Infrastructure and tools



The Shift Left Continuum

Testing Personas

Role	Key Change
Developer Tester	Does not like testing, but if she/he has to automate -- loves doing so through programming in her/his IDE. No record/replay or script-less. (coding)
Technical Tester	A testing SME who understands code but does not program; Works with and understands technology. Uses scripting languages fluently, but also script-less, record/replay for automating. (low code)
Business Tester	Has no technical skills, no scripting or coding skills, prefers totally codeless tools for testing. (no code)

Continuous Testing is also for the Bots

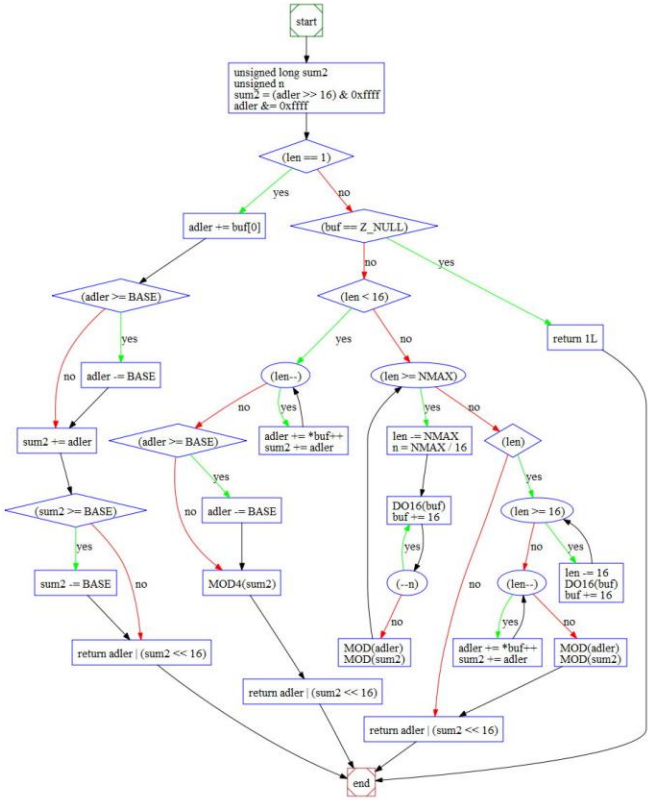
- Modern technology foundation
- Nurture all personas
- AI augmentation
- Low-code/no code

Infusing AI in Continuous Testing

20 Years Old Code Visualization Examples

login ✕ Start Page

Item	Operation	Value	Documentation
Login			
Welcome: Mercury Tours	Navigate	"http://newtours.de...	Navigate to "http://newtours.demoaut...
Welcome: Mercury Tours			
userName	Set	"Mercury"	Enter "Mercury" in the "userName" ed...
password	SetSecure	"4fb8e4adb99e85e...	Enter the encrypted password in the "...
Sign-In	Click	"20,1"	<No documentation summary is avail...
Find a Flight: Mercury			
Find a Flight: Mercury			
fromPort	Select	"New York"	Select the "New York" item from the "...
fromMonth	Select	"Mar"	Select the "Dec" item from the "from...
fromDay	Select	"10"	Select the "29" item from the "fromDa...
toPort	Select	"San Fancisco"	Select the "San Fancisco" item from t...
toMonth	Select	"Mar"	Select the "Dec" item from the "toMon...

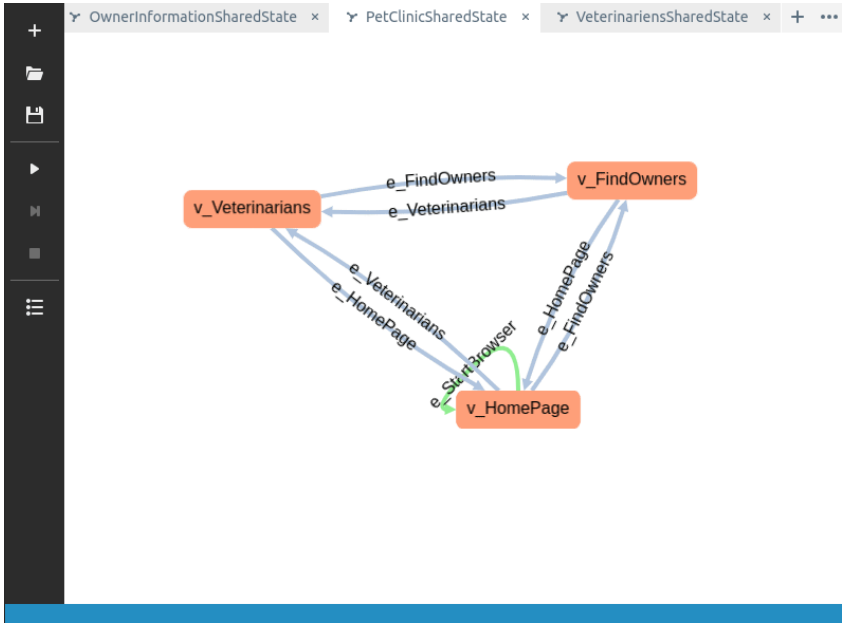


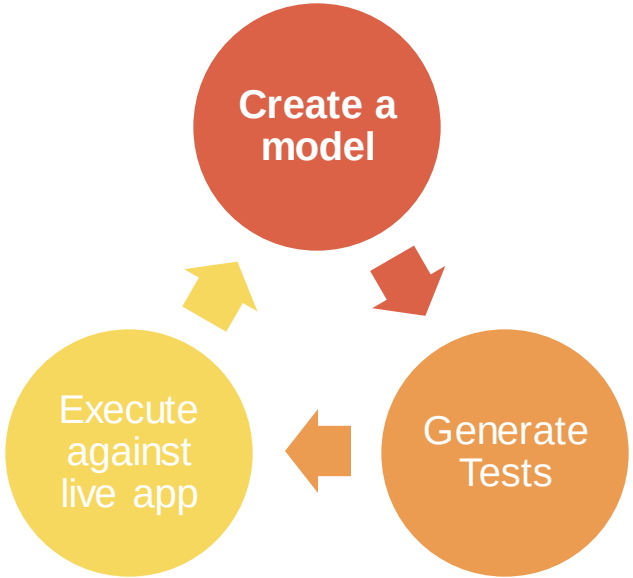
Why Code Visualization Failed

- It did not reduce the complexity
- It introduced an alternative, but utilized the same logic
- Complex logic usually results in complex presentations
- Did not meet the requirements of all stakeholders
- Did not scale at the level enterprises required

Traditional Model Based Testing

Software testing technique where tests are derived from a model of the functional aspects of the system being tested. Models include offline and online testing.





Digital.ai Model Based Testing

Organizations that leverage the Digital.ai Continuous Testing Solution are enabled to:

- Automate testing models to accelerate release
- Gain full visibility of the changes occurring within their testing cycles
- Save resources that would otherwise be spent executing endless numbers of manual tests

Demo

Optimize Your Customers Experience

Increase Efficiency

- Provide immediate test feedback
- Shorten investigation times with automated root-cause analysis
- Accelerate testing cycles
- Improve developer-tester collaboration and productivity

Accelerate Agility

- Increase release velocity
- Empower non-technical users to create and run tests
- Enable developers to easily create accurate tests right from their IDE
- Harness distributed testing teams

Reduce Risk

- Predict users' reactions to new releases to avoid customer experience issues
- Ensure the software release candidate is within acceptable business risk using QA analytics

Scale Test Operations With 24/7 Continuous Testing Environment

United Airlines saved 400 hours of effort each month and more than $24M in 2019

Challenge

20+
B2B and B2C Apps
160M+ Customers

100+
Devices
running iOS and Android

Outcome

400
Hours of Effort
By executing large-scale automated tests

$24M
Annual Savings

Digital.ai – The Enterprise Solution for Software-Driven Value

- Automate and improve processes
- Integrate the entire software lifecycle
- Scale best practices enterprise wide
- Make data-driven decisions
- Drive innovation and meet evolving customer needs
- No rip & replace

Thank you