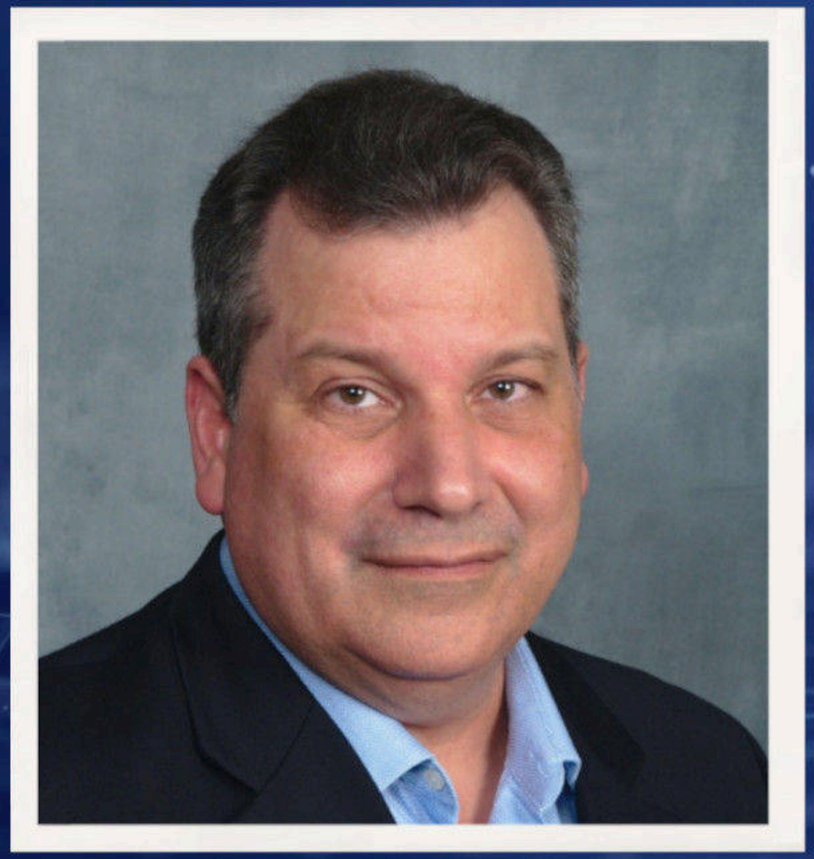
South Florida Tech Hub | SOFLO DEVCON 2023



South Florida Tech Hub Developer Conference 2023

soflodevcon.com

Software Engineering Management: Integrating Software Testing

Eugenio Alvarez

4:00 PM

SATURDAY | APRIL 15, 2023 | 7:30 AM - 7:30 PM | NSU Florida | NOVA SOUTHEASTERN UNIVERSITY

EUGENIO ALVAREZ

A South Florida software engineering professional. Experienced in organizational design, software design, construction, and deployment. Started working with Java in 1996. Unit test infected since 1999. An advocate for Agile Software Engineering methods using Kanban and Scrum.

www.linkedin.com/in/ealvarez

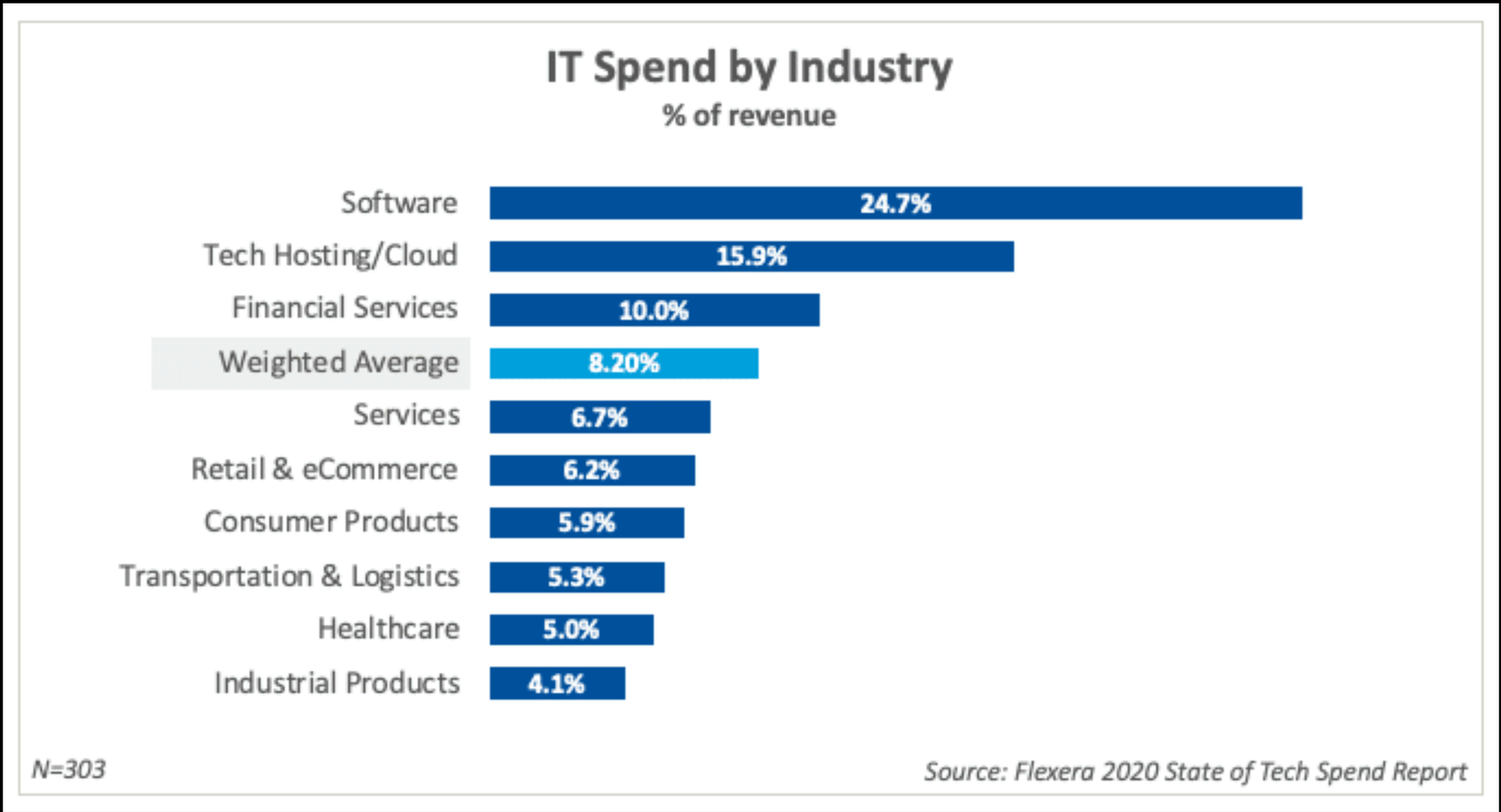
INTRODUCTION

- Business Context for Software Testing
- Tools to avoid bugs
- Testing levels
  - Unit testing
  - Integration testings
  - System testing
  - Production testing
- Other kinds of Testing
- Testing Anti-patterns

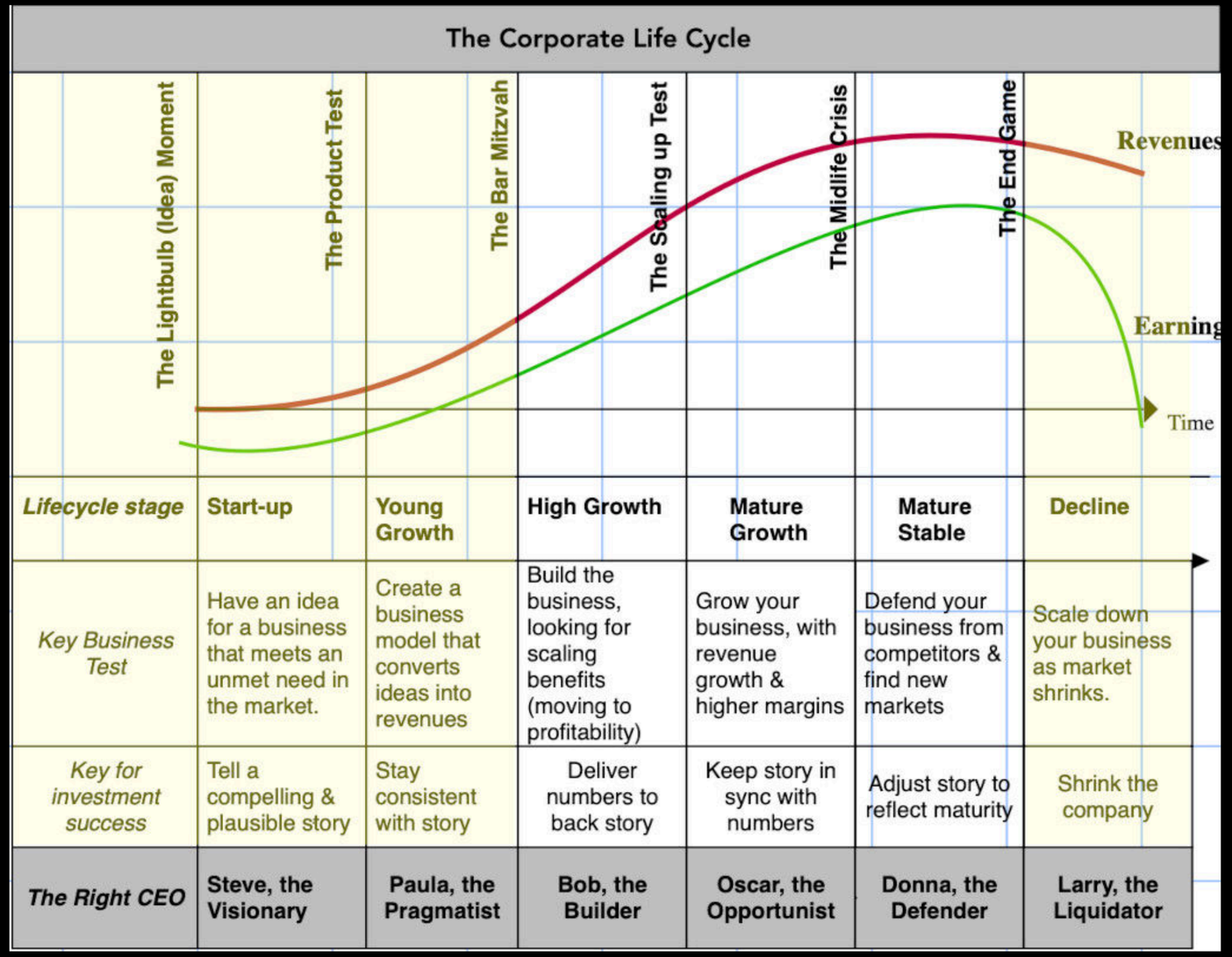
BUSINESS CONTEXT FOR SOFTWARE TESTING

- Considerations include:
  - IT Spend by Industry
  - Corporate Lifecycle
  - Product Lifecycle
  - Corporate organization
  - Software architecture

SOFTWARE TESTING AS PART OF IT SPEND BY INDUSTRY



SOFTWARE TESTING CORPORATE LIFECYCLE



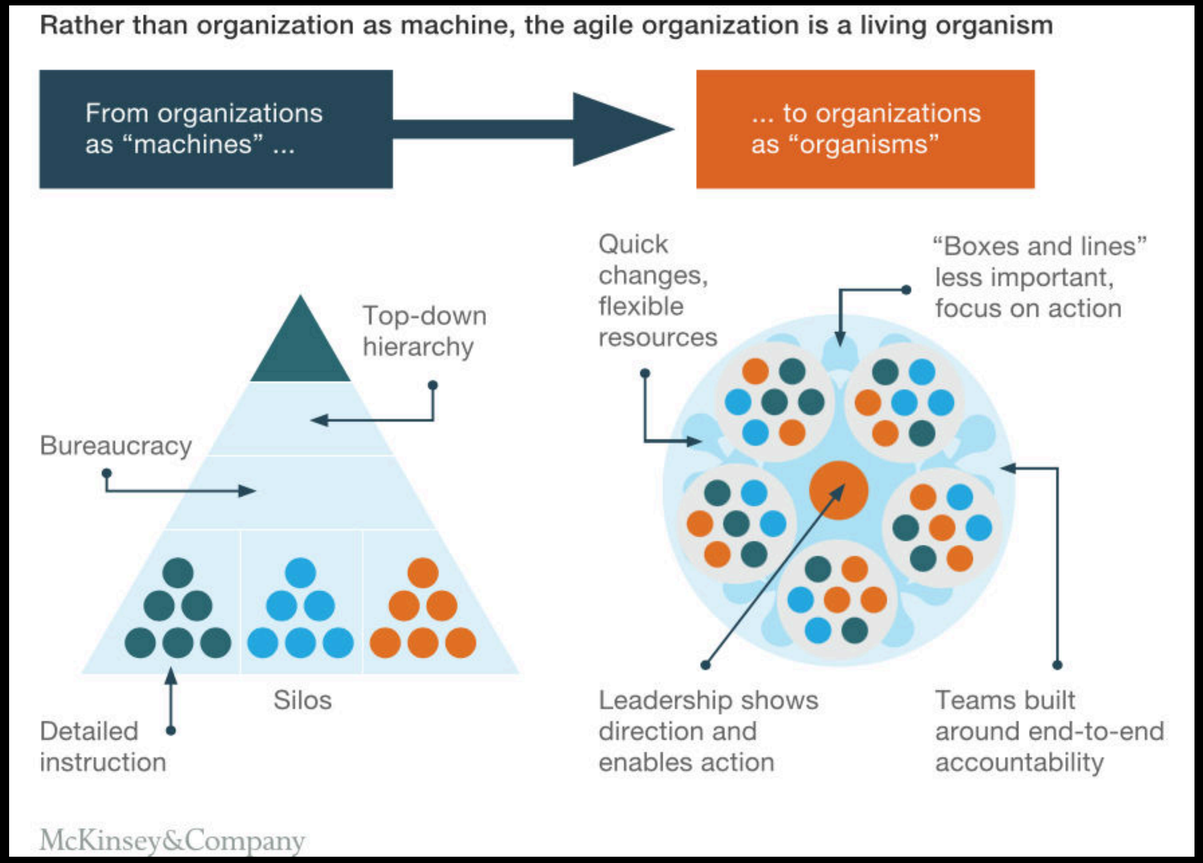
Lifecycle stage	Start-up	Young Growth	High Growth	Mature Growth	Mature Stable	Decline
Key Business Test	Have an idea for a business that meets an unmet need in the market.	Create a business model that converts ideas into revenues	Build the business, looking for scaling benefits (moving to profitability)	Grow your business, with revenue growth & higher margins	Defend your business from competitors & find new markets	Scale down your business as market shrinks.
Key for investment success	Tell a compelling & plausible story	Stay consistent with story	Deliver numbers to back story	Keep story in sync with numbers	Adjust story to reflect maturity	Shrink the company
The Right CEO	**Steve, the Visionary**	**Paula, the Pragmatist**	**Bob, the Builder**	**Oscar, the Opportunist**	**Donna, the Defender**	**Larry, the Liquidator**

Source: https://aswathdamodaran.blogspot.com/2021/12/managing-across-corporate-life-cycle.html

SOFTWARE TESTING PRODUCT LIFECYCLE

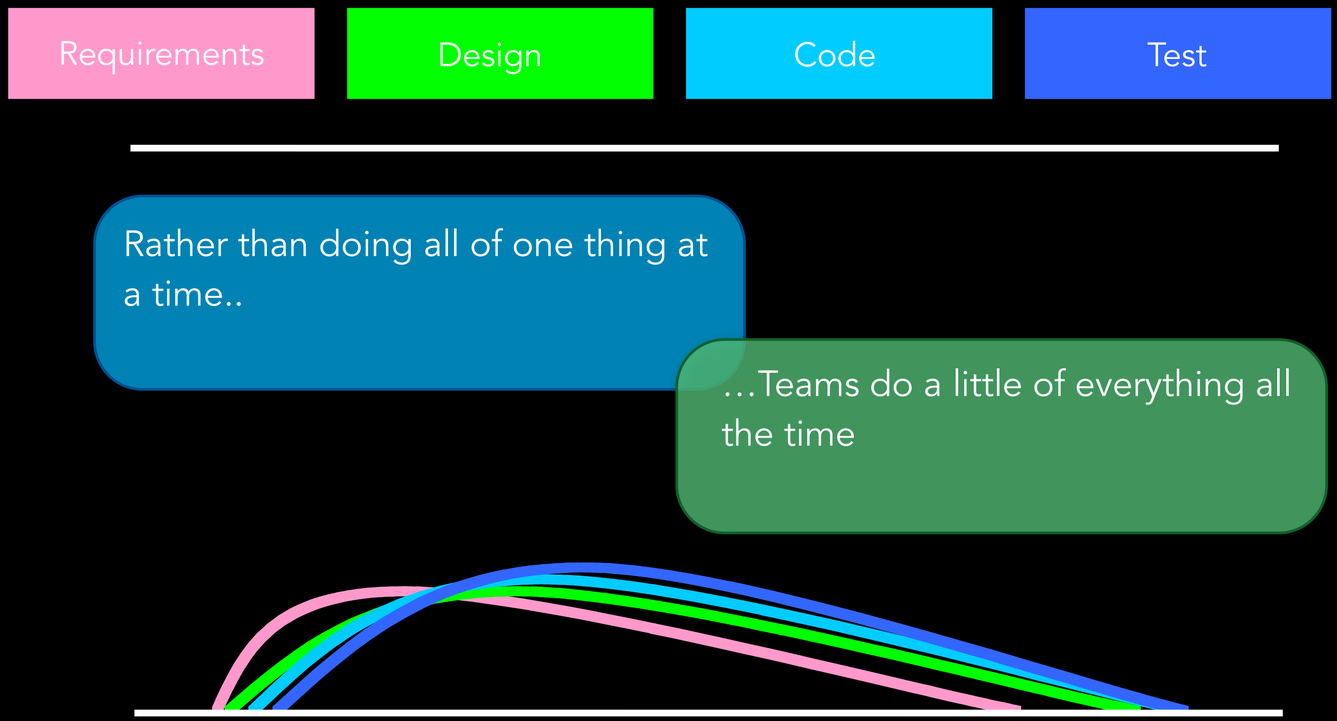
Source: 280 Group https://280group.com/product-management-blog/the-power-of-the-product-lifecycle-a-look-at-the-4-key-stages-of-the-plc/

SOFTWARE TESTING
IT ORGANIZATIONAL DESIGN



Source: McKinsey & Company

SOFTWARE TESTING METHODOLOGY / PROCESS DESIGN

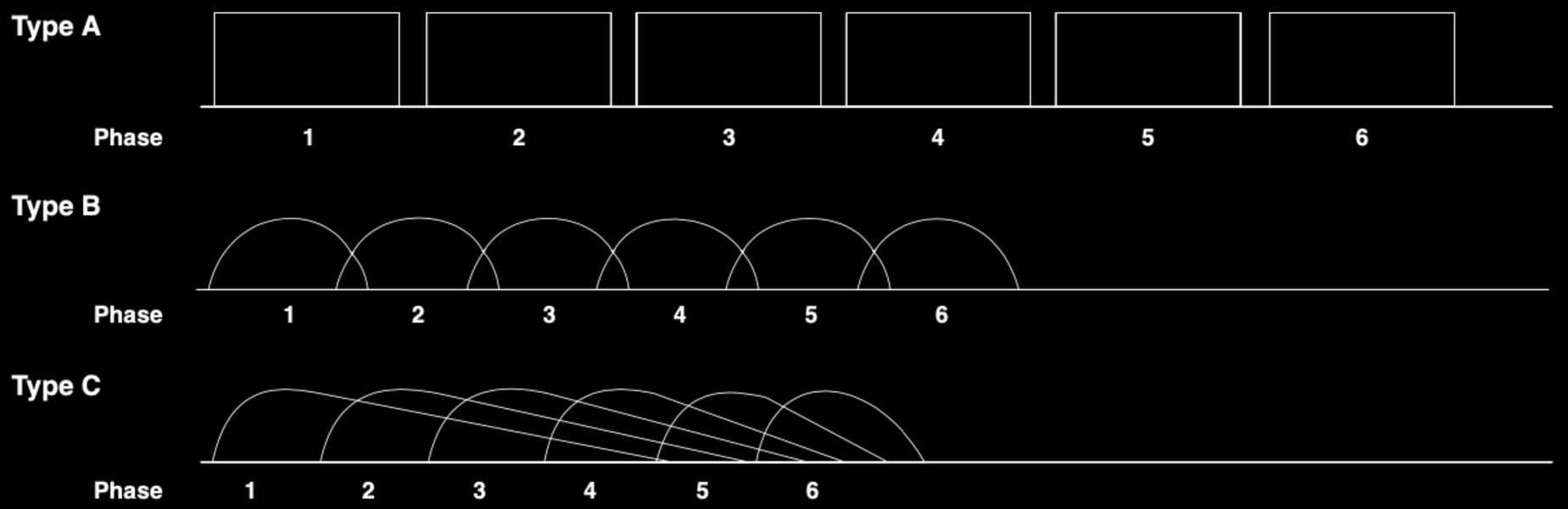


Graphics from: "Mike Cohn's Scrum Presentation" Mountain Goat Software, LLC

Source: "The New New Product Development Game" by Takeuchi and Nonaka. Harvard Business Review, January 1986

SOFTWARE TESTING METHODOLOGY / PROCESS DESIGN

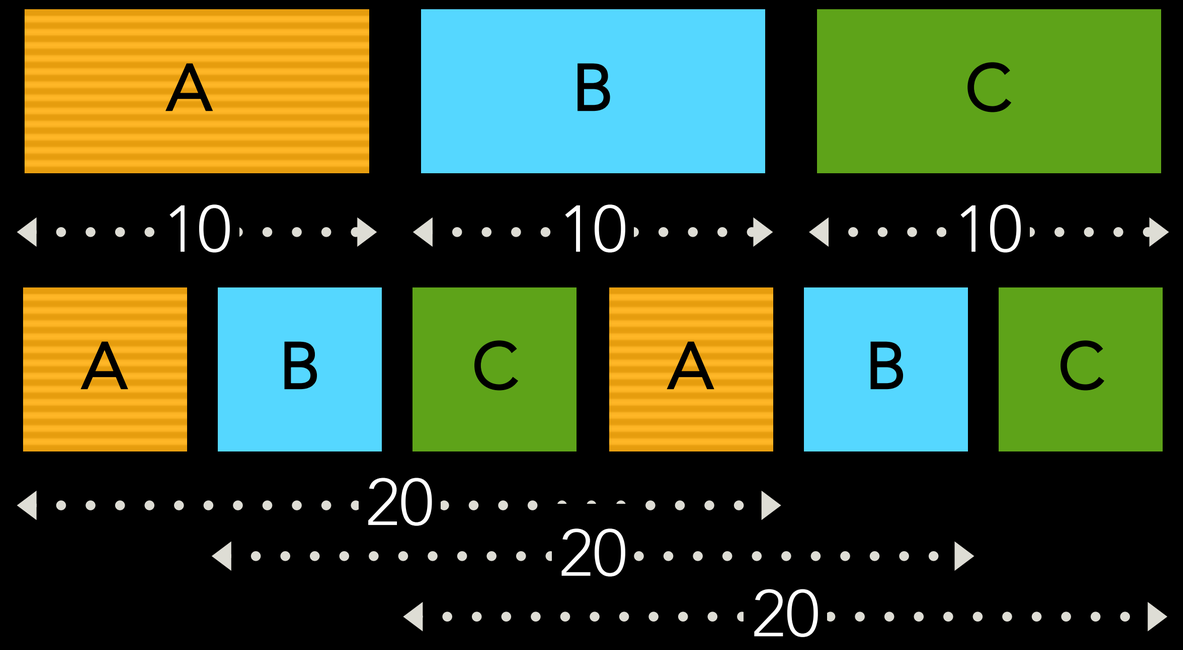
EXHIBIT 1
Sequential (A) vs. overlapping (B and C) phases of development



Source: "The New New Product Development Game" by Takeuchi and Nonaka. Harvard Business Review, January 1986

BUT DOESN'T THAT TAKE LONGER

"Do you realize what impact ***multi-tasking*** has on lead time?"



Source: Eliyahu M. Goldratt (1997), *Critical Chain*

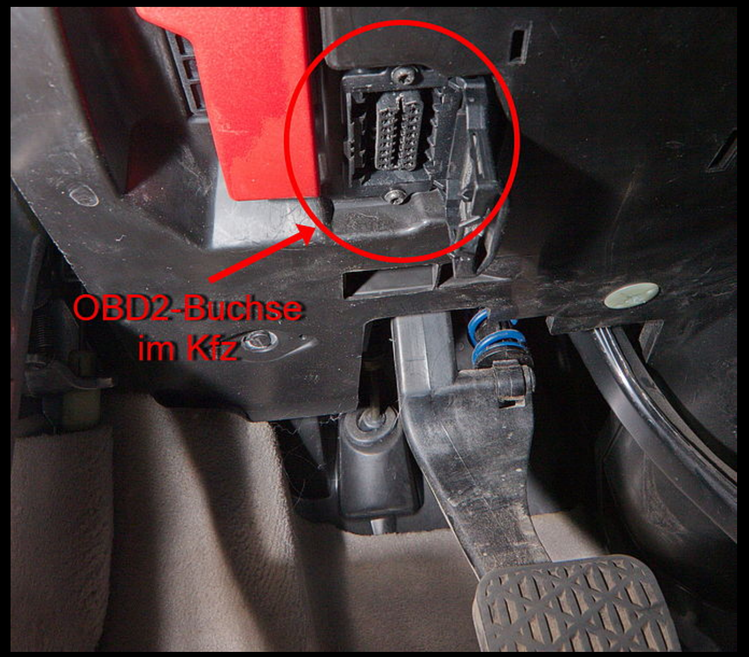
TESTABLE SOFTWARE DESIGNS PRODUCE BETTER SOFTWARE ARCHITECTURE

MONOLITHIC SOFTWARE IS LIKE MAKING A CAR WITHOUT A DIAGNOSTIC PORT

MODULAR DESIGN

INVERSION OF CONTROL DESIGN PATTERN

OBD2 DIAGNOSTIC PORT



Source: https://en.wikipedia.org/wiki/Inversion_of_control
Source: https://commons.wikimedia.org/wiki/File:OBD2-Buchse_am_Mercedes-Benz_IMG_1701.jpg

SOFTWARE TESTING NEWS

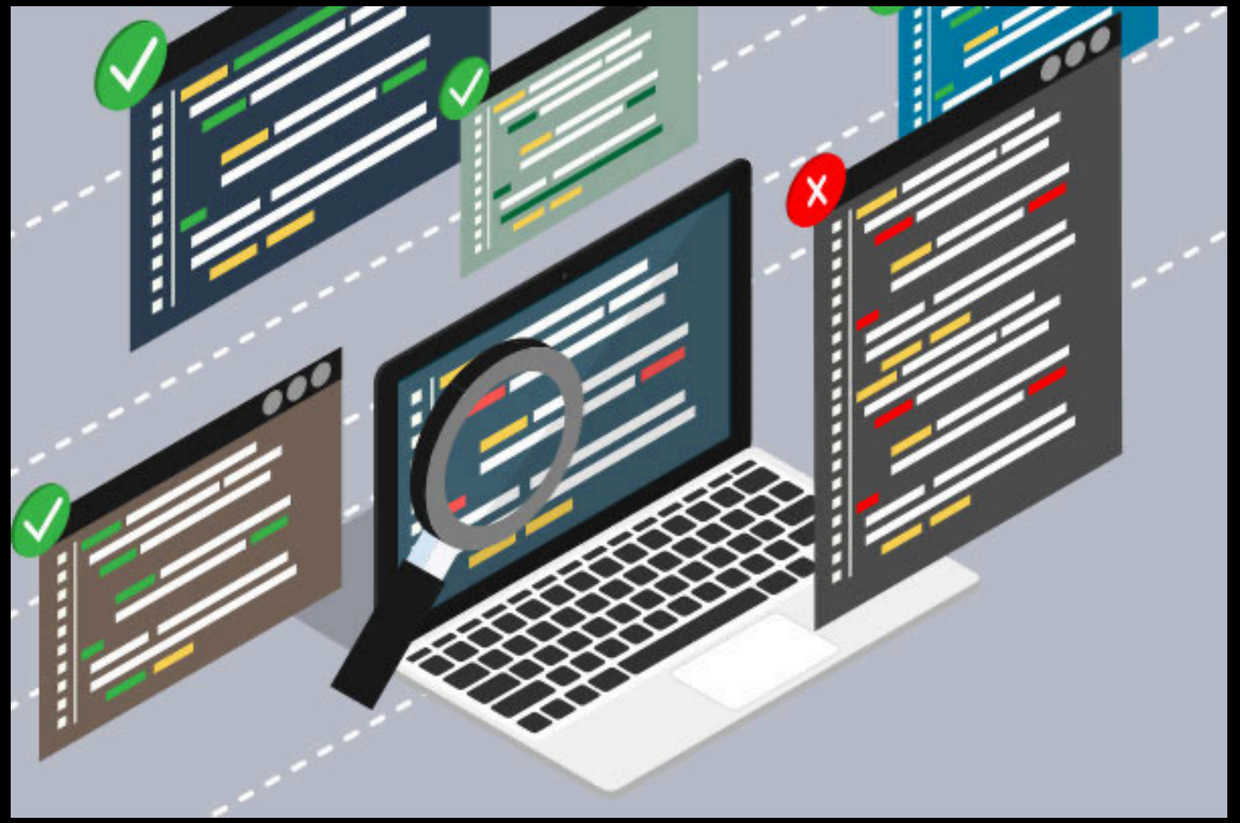
- No matter the Industry Trends, Corporate Lifecycle, Product Lifecycle, Company Culture or Software Architecture
- There is always room for improving Software Testing

Source: https://commons.wikimedia.org/wiki/File:Good_news_is_coming_%28Unsplash%29.jpg

START WITH TRAINING FOR TESTING

Source: https://www.istqb.org/certifications/certification-list/

STATIC CODE ANALYSIS



Finding bugs and security issues before testing

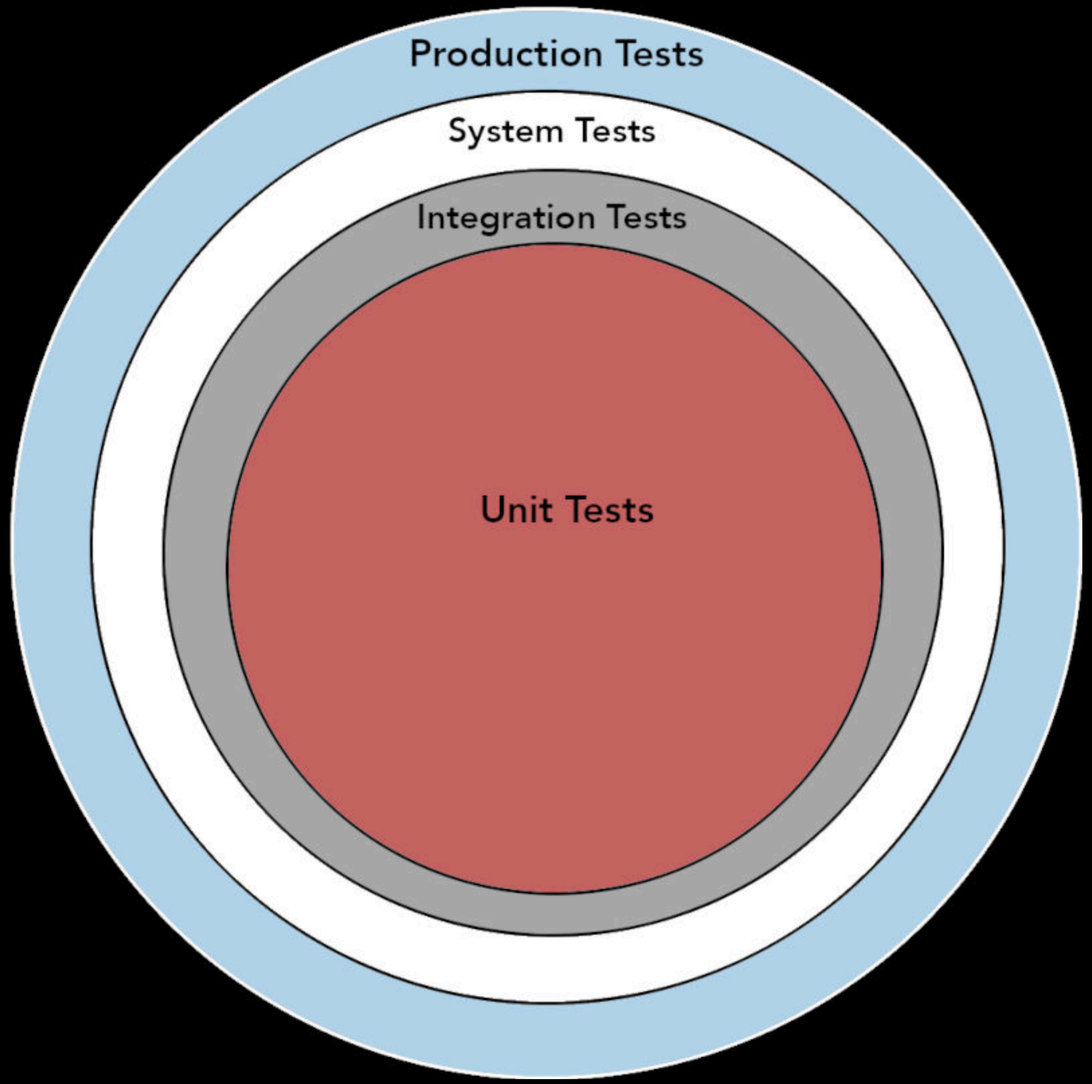
Source: https://www.perforce.com/blog/sca/what-static-analysis

SOFTWARE COMPOSITION ANALYSIS

Your bugs and security issues might be coming from the open source libraries

Source: https://fossa.com/blog/framework-for-evaluating-software-composition-analysis-tools/

TEST LEVELS

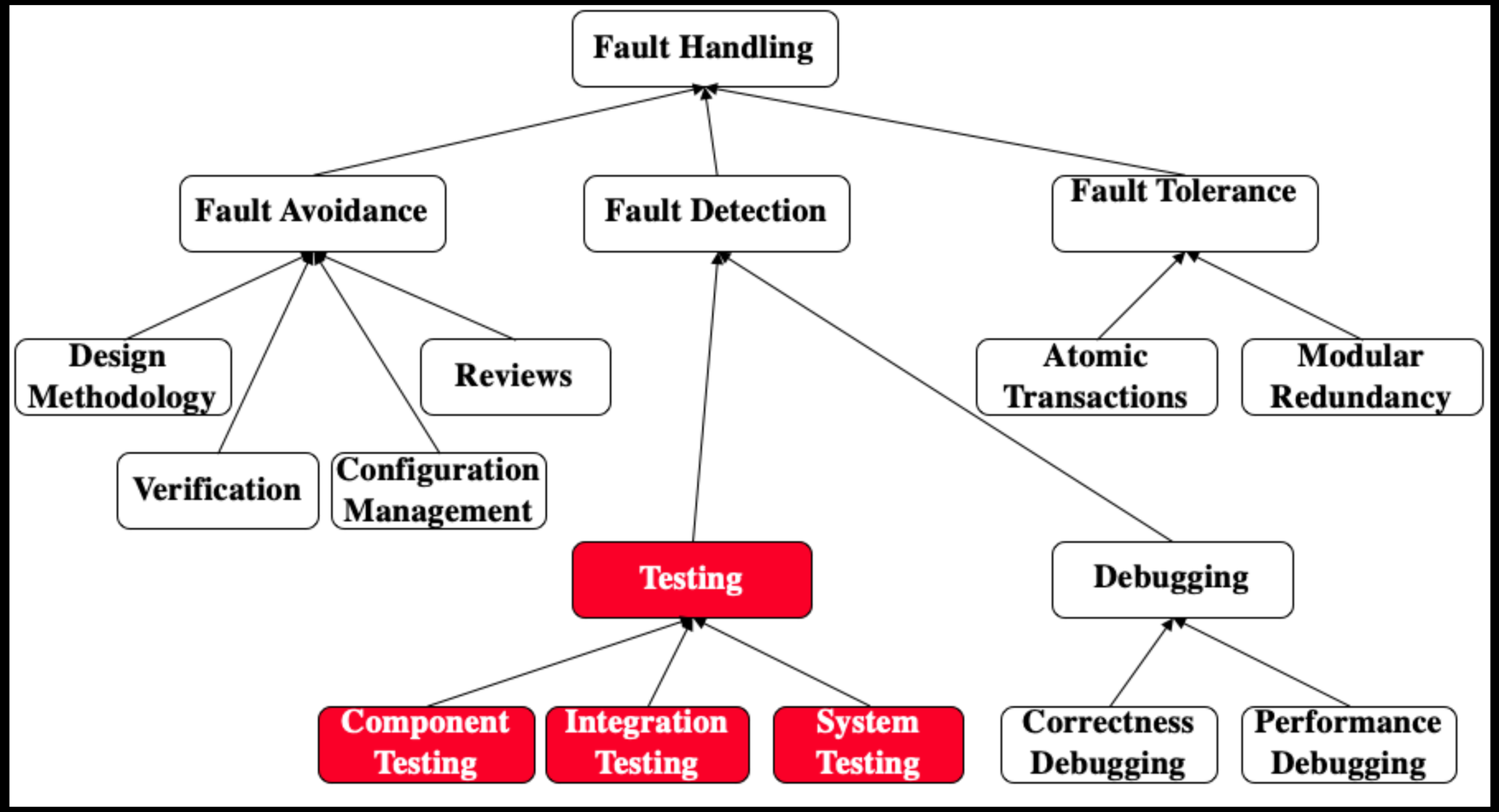

Production Tests
System Tests
Integration Tests
Unit Tests

OTHER KINDS OF TESTING

- Blackbox Testing
- Whitebox Testing
- Acceptance Testing
- Performance Testing
- Installation Testing

WHERE DOES TESTING FIT IN?

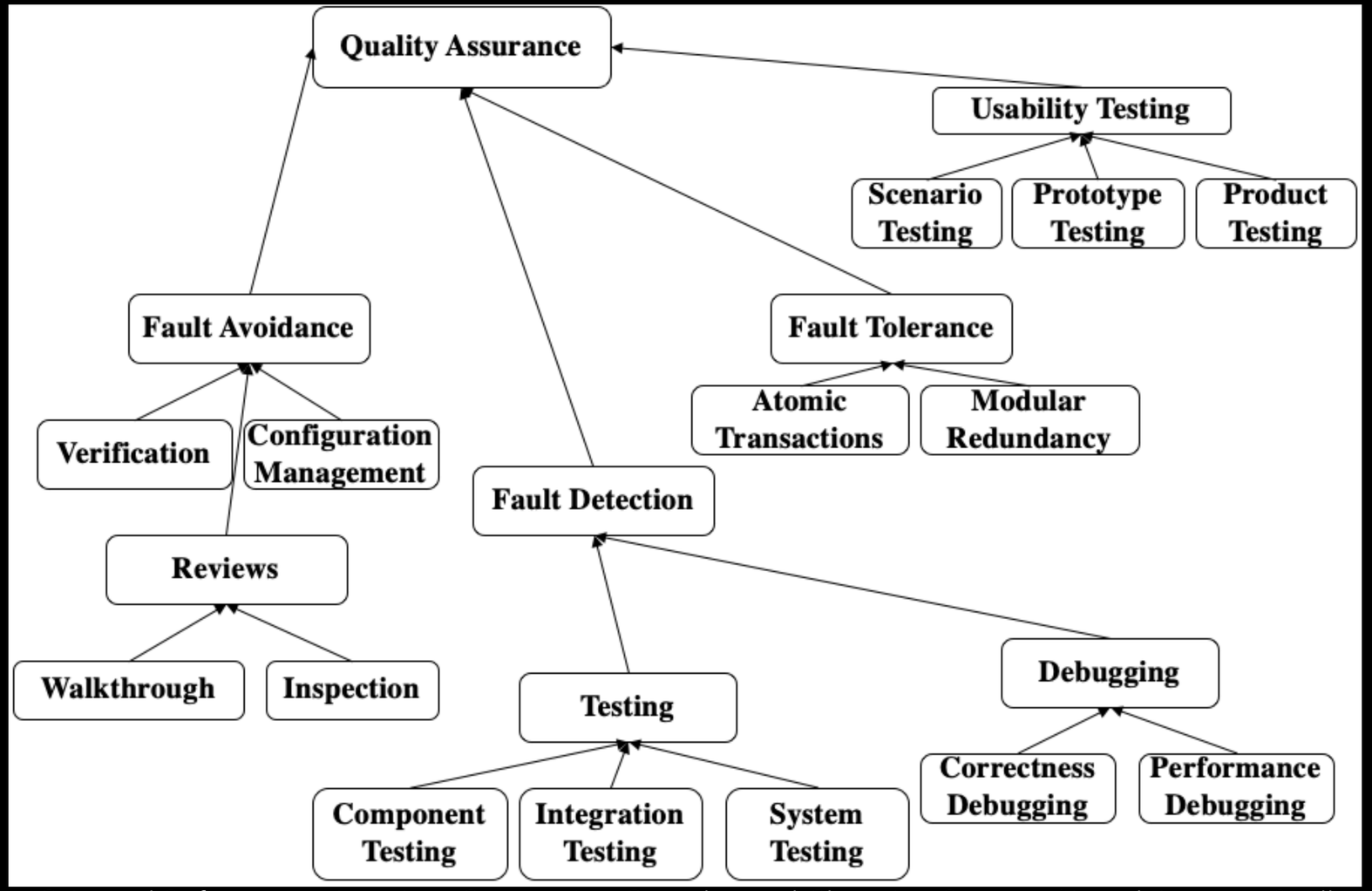
QUALITY ASSURANCE ENCOMPASSES TESTING



Source: Object-Oriented Software Engineering: Conquering Complex and Changing Systems (Bernd Bruegge & Allen H. Dutoit)

OTHER KINDS OF TESTING

FAULT HANDLING TECHNIQUES

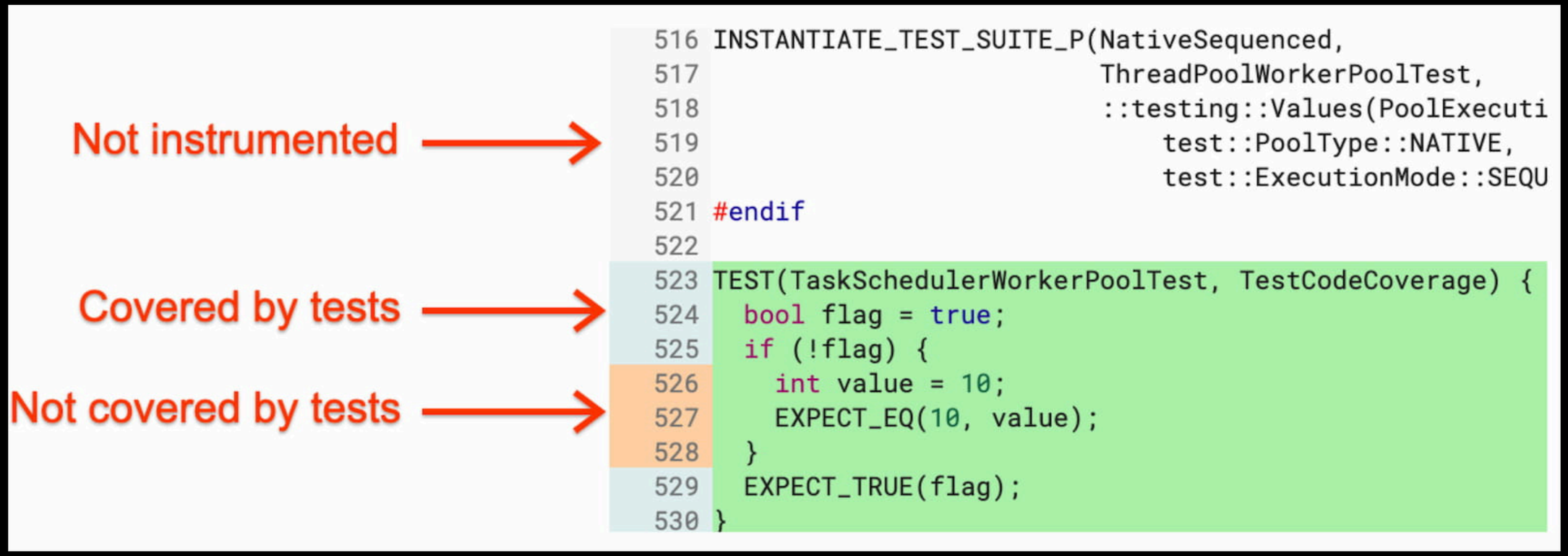


Source: Object-Oriented Software Engineering: Conquering Complex and Changing Systems (Bernd Bruegge & Allen H. Dutoit)

UNIT TESTS

- Designed to test individual components
- Cover boundary conditions
- Cover edge cases and error conditions
- Are small and designed to execute quickly
- Many unit test are execute together to create a suite
- Are usually created by developers
- Coverage Reports are typically used to review them

CODE COVERAGE REPORTING



Source: https://developer.chrome.com/blog/chromium-chronicle-3/

INTEGRATION TESTS

- Designed to test multiple components
- Can discover compatibility issues between components
- Cover edge cases and error conditions
- Larger test that can take longer to execute
- Created by developers and testing team
- Coverage Reports can also be used to review them
- Since Unit tests run fast they are run before integration tests

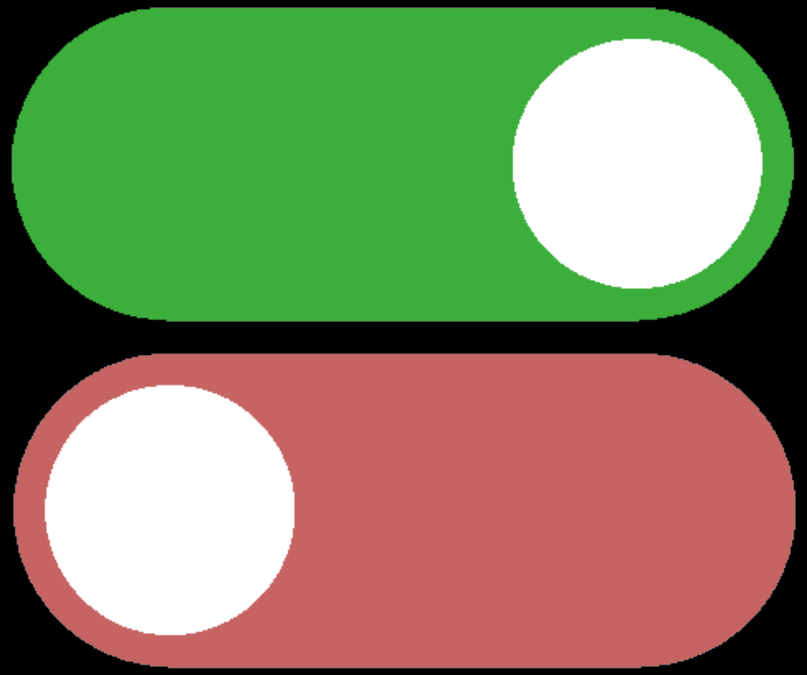
SYSTEM TESTS (E2E)

- Entired application is tested
- Covers functional and non-functional requirements
- End-to-end testing to find component interoperability
- Usually after integration testing
- Takes the longest amount of time to complete
- Usually created by Testing team

PRODUCTION TESTS

- Testing in production environment
- Feature toggle used enable/disable new feature
- Usually subset of users or environments are tested
- Any issues identified disable new feature
- Reduces risk and need to rollback
- Designed by developers as part of the feature

SOFTWARE FEATURE TOGGLE



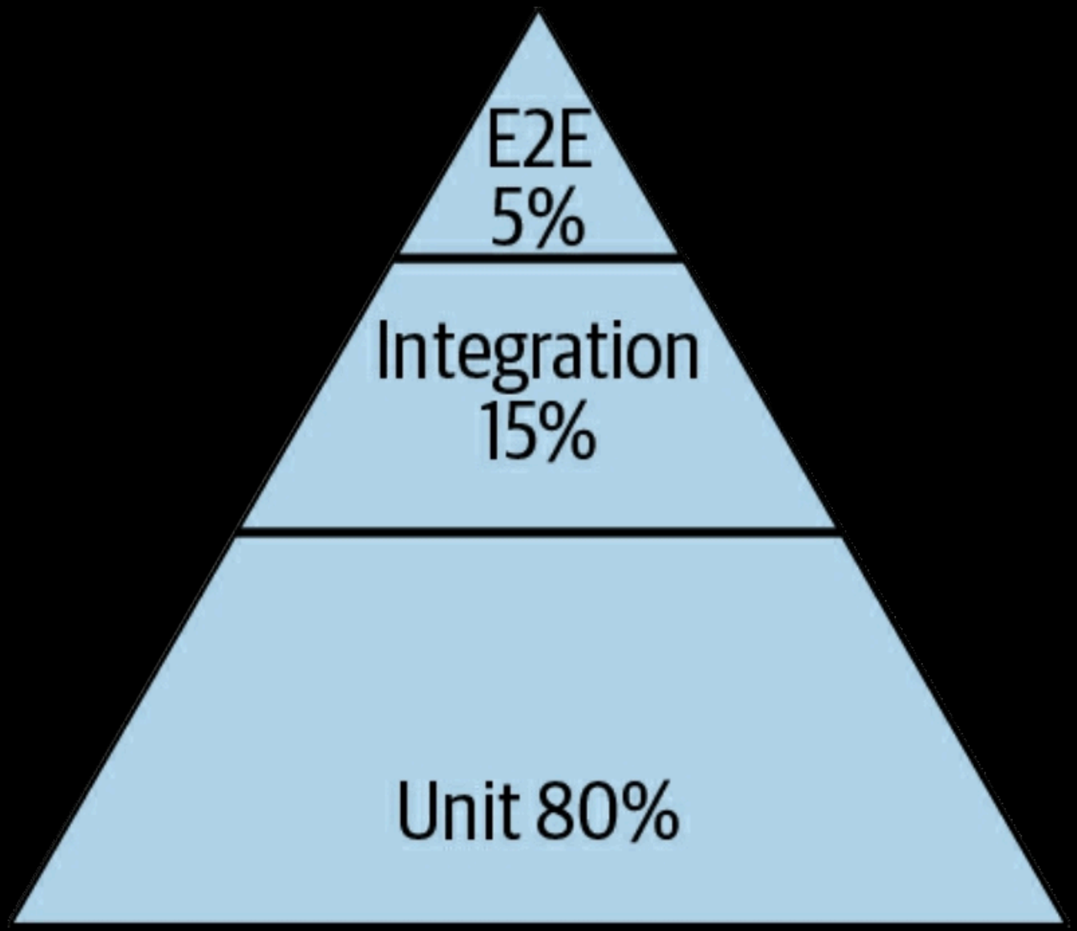
TYPICALLY USED FOR
PRODUCTION TESTING

SOFTWARE TESTING ANTI-PATTERNS

- Ideal Testing Pyramid
- Hourglass Anti-Pattern
- Hexagonal Anti-Pattern
- Ice Cream Cone Anti-Pattern

Reference: https://en.wikipedia.org/wiki/Anti-pattern

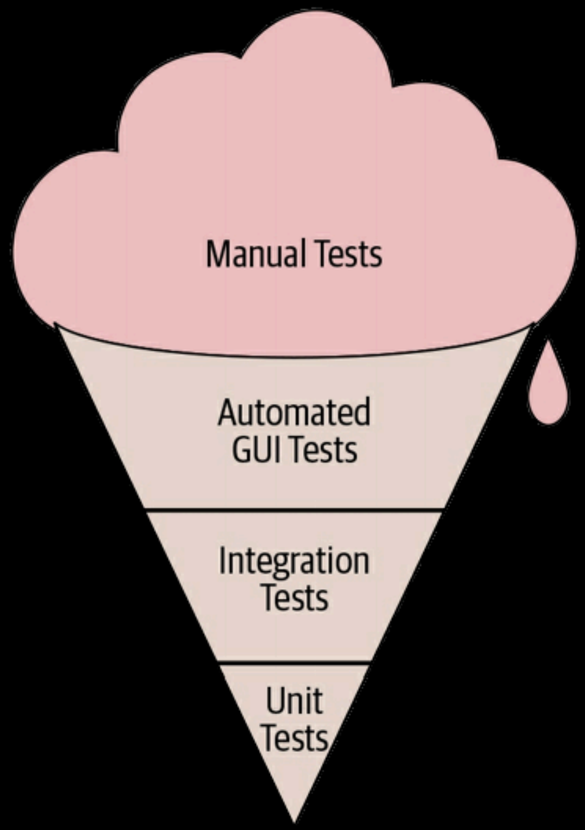
IDEAL SOFTWARE TESTING PYRAMID



Source: Winters, Titus; Manshreck, Tom; Wright, Hyrum. Software Engineering at Google. O'Reilly Media, 2020

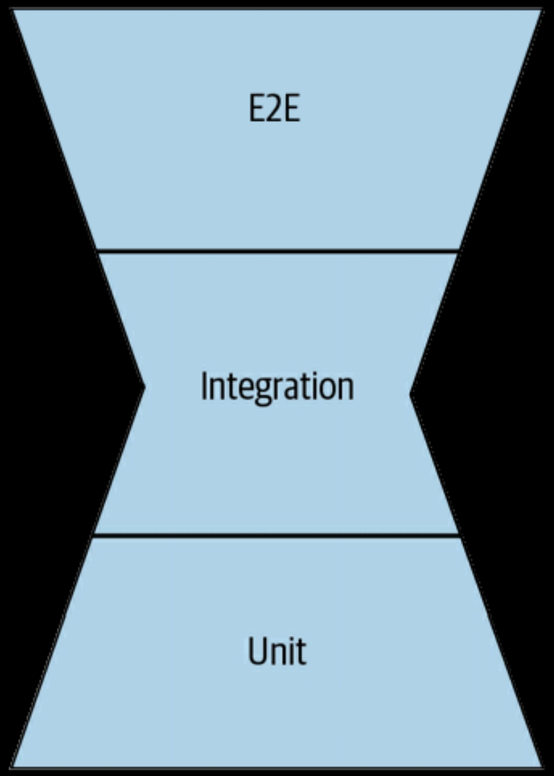
Source: Mike Cohn, Succeeding with Agile: Software Development Using Scrum (New York: Addison-Wesley Professional, 2009

SOFTWARE TESTING
"ICE CREAM CONE" ANTI-PATTERN



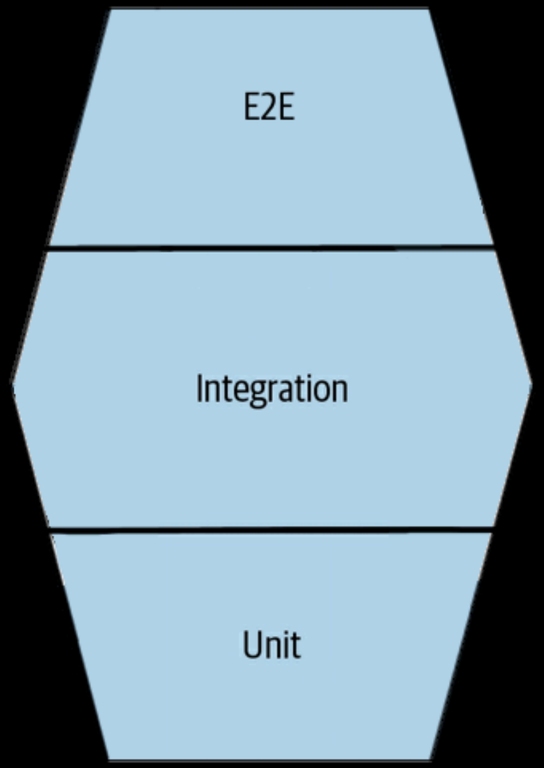
Source: Winters, Titus; Manshreck, Tom; Wright, Hyrum. Software Engineering at Google. O'Reilly Media, 2020

SOFTWARE TESTING "HOURGLASS" ANTI-PATTERN



Source: Winters, Titus; Manshreck, Tom; Wright, Hyrum. Software Engineering at Google. O'Reilly Media, 2020

SOFTWARE TESTING
"HEXAGONAL" ANTI-PATTERN



Source: Eugenio Alvarez, Software Engineering: Integrating Software Testing, SoFloDevCon 2023

UNIT TESTING

- My friend works at a large social media company says that they do not write unit test
- Large social media company is less concerned about errors in production
- Feature toggles used extensively in production
- Multiple deploys for per day to allow for quick fixes

IF IT AIN'T BROKE DON'T FIX IT

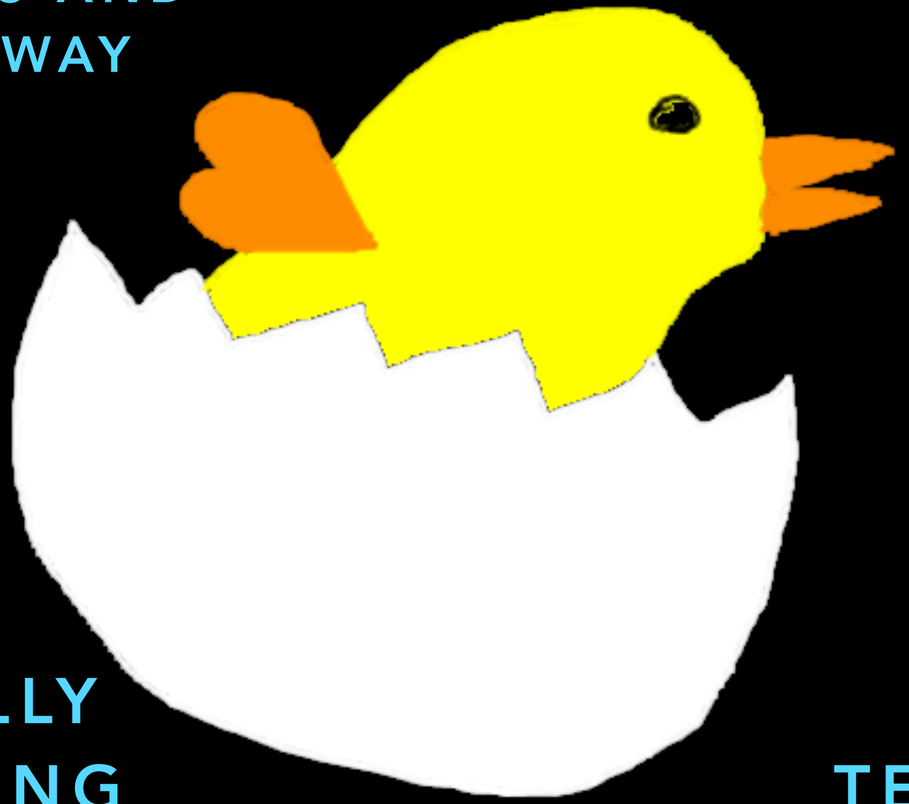
THIS IS HOW WE WOULD GET AROUND IF WE FOLLOWED THE SAYING

Source: *https://commons.wikimedia.org/wiki/File:Horse_(Cleveland_Bay)_Drawn_Clarence_(Brougham)_Carriage_%26_Victoria_Memorial,_Buckingham_Palace,_Westminster,_London_(3795290693).jpg*

TECHNOLOGY HATCHLING IMPRINTING

I HAVE WORKED AT ENTERPRISE COMPANIES AND THEY ALL DO IT THIS WAY

WE HAVE ALWAYS DONE IT THIS WAY



NO ONE ACTUALLY DOES UNIT TESTING

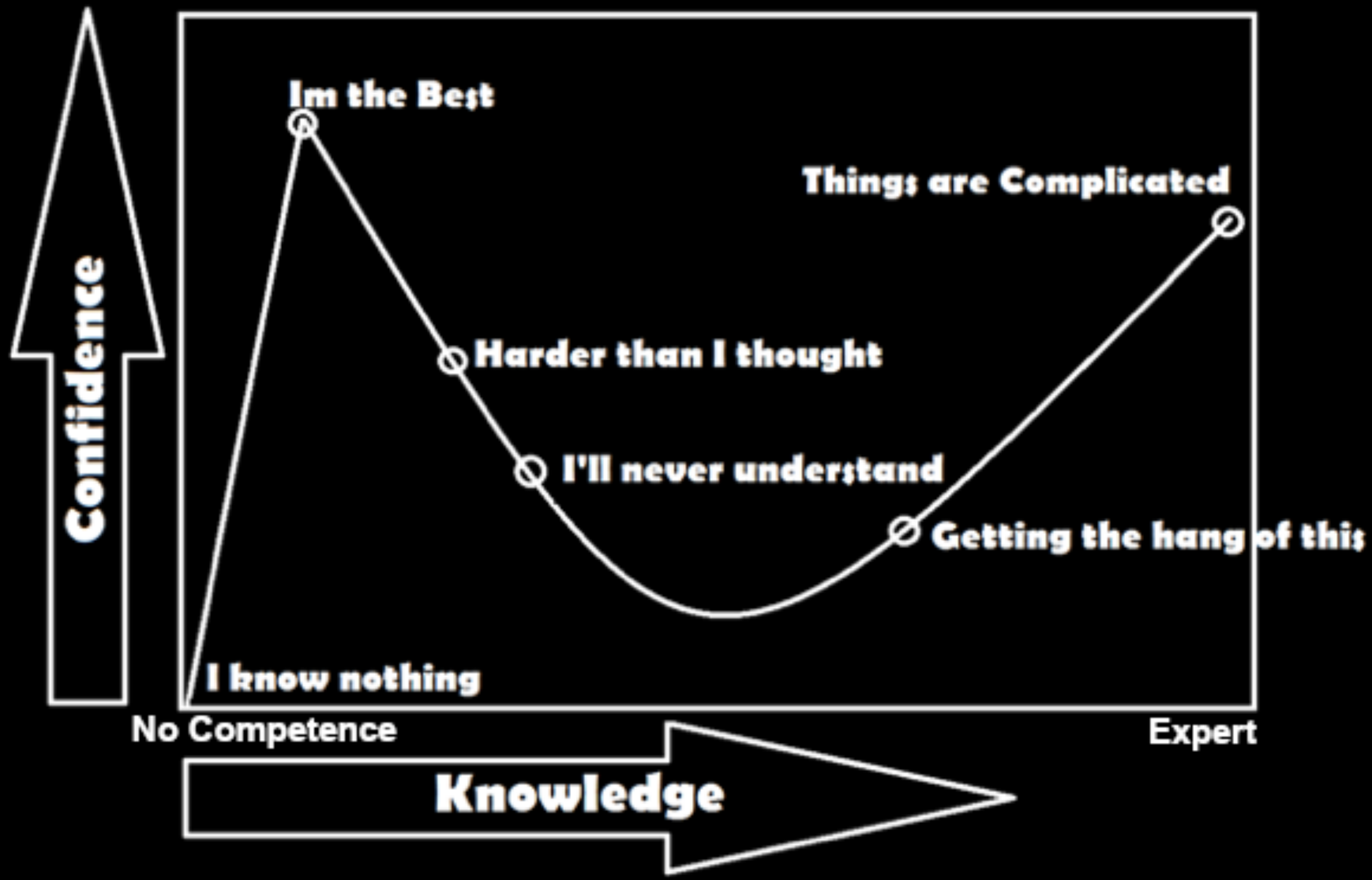
NOBODY DOES TESTING THIS WAY

DEFAULT DESIRE PATH

THE BEATEN PATH IS EASIER

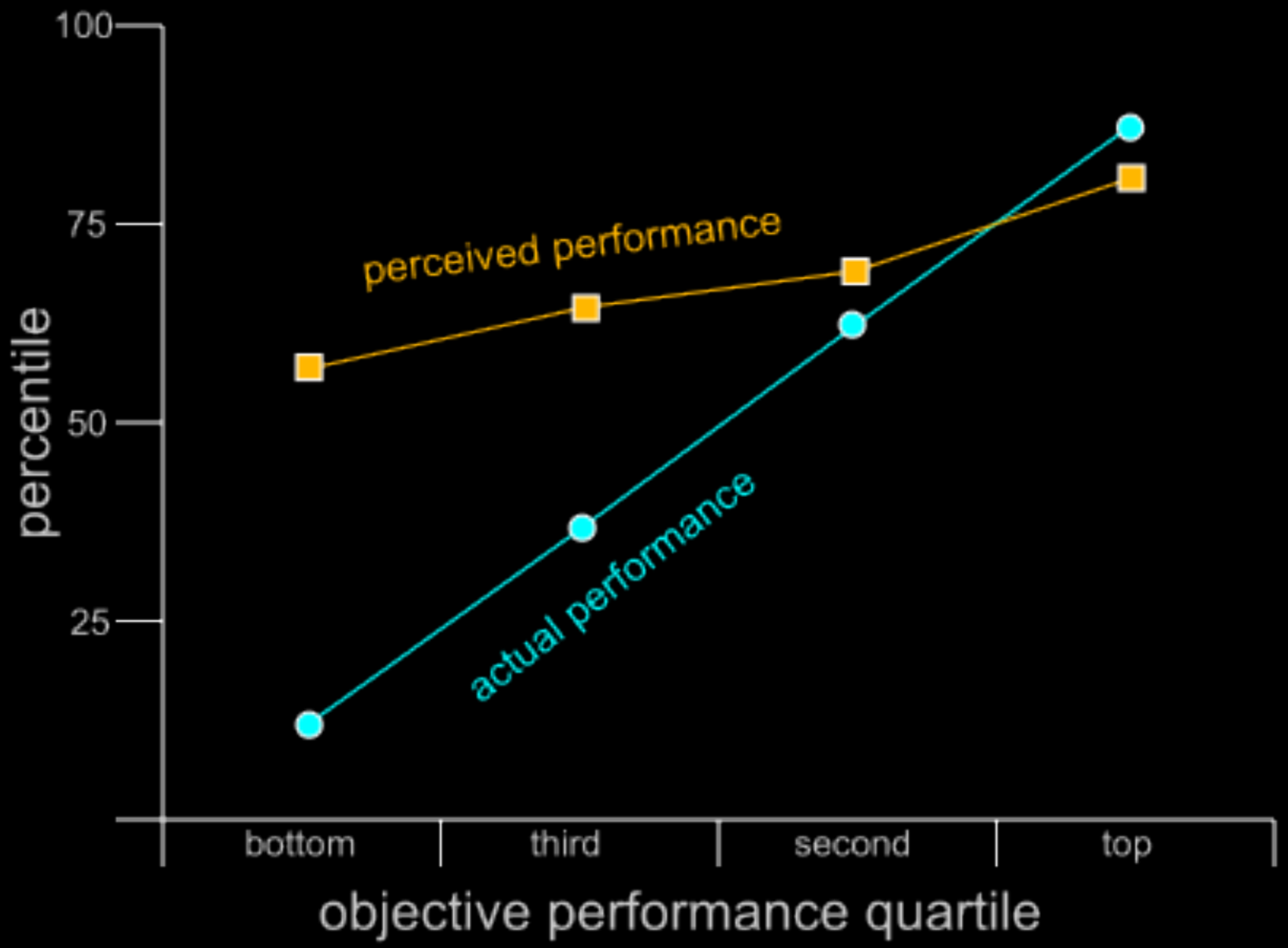
https://en.wikipedia.org/wiki/Desire_path

DUNNING-KRUGER EFFECT



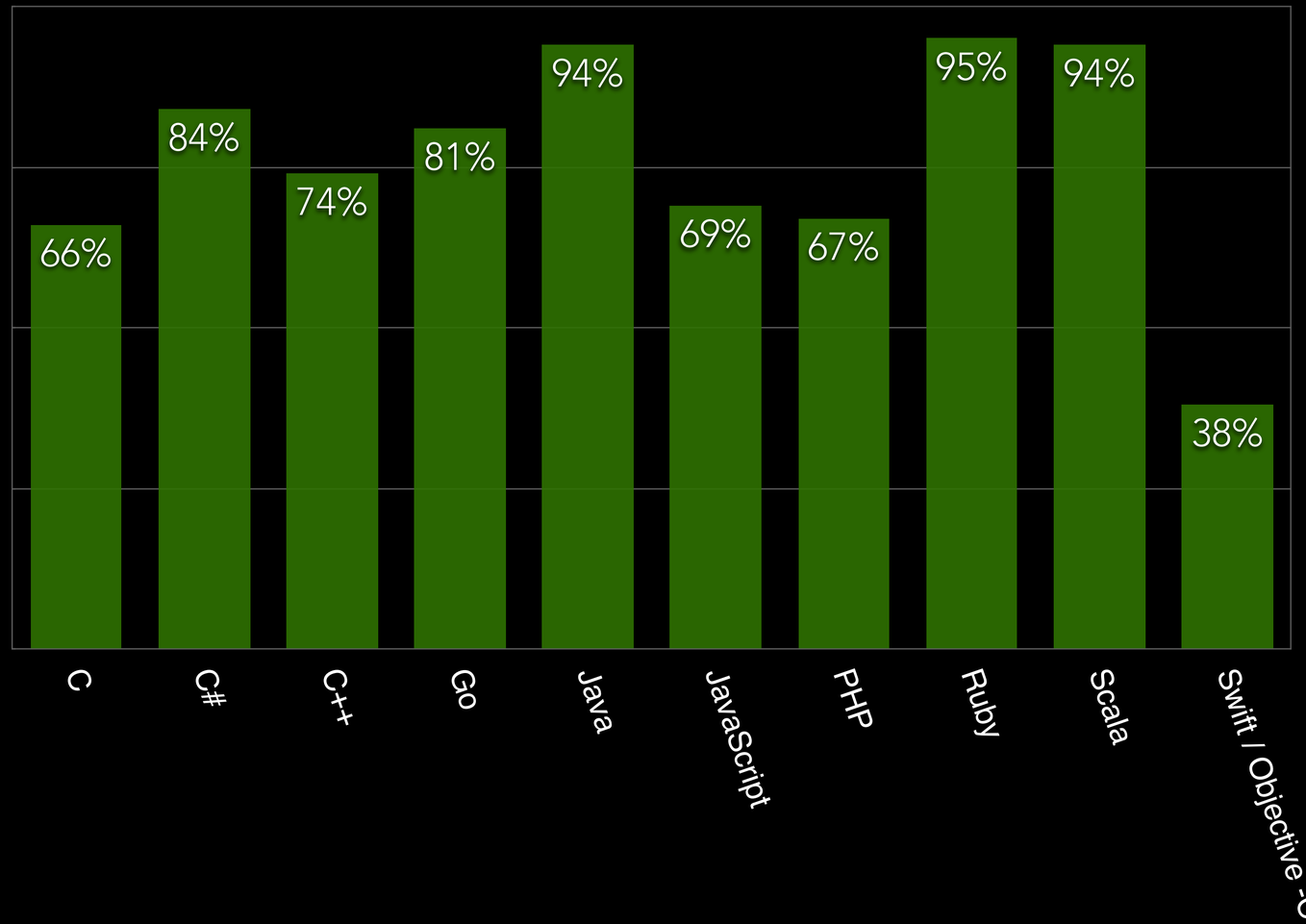
Reference: https://en.wikipedia.org/wiki/Dunning%E2%80%93Kruger_effect

DUNNING-KRUGER EFFECT



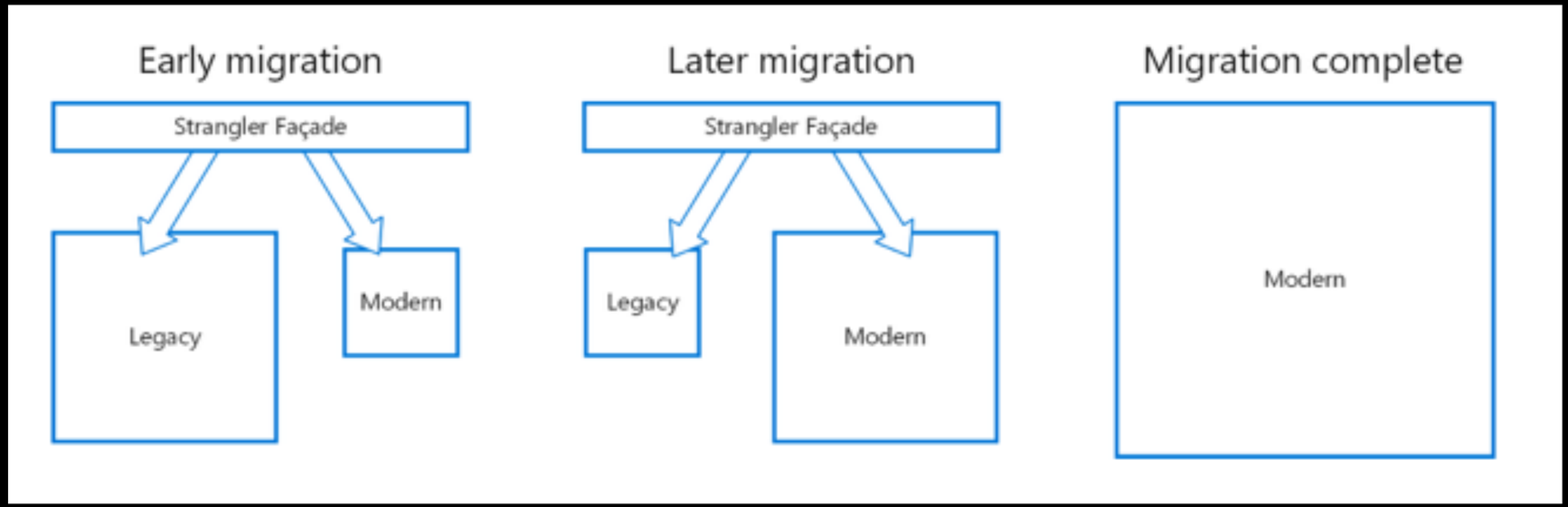
Source: https://commons.wikimedia.org/wiki/File:Dunning-kruger_effect_-_percentile.svg

UNIT TEST USAGE VARIES BY LANGUAGE



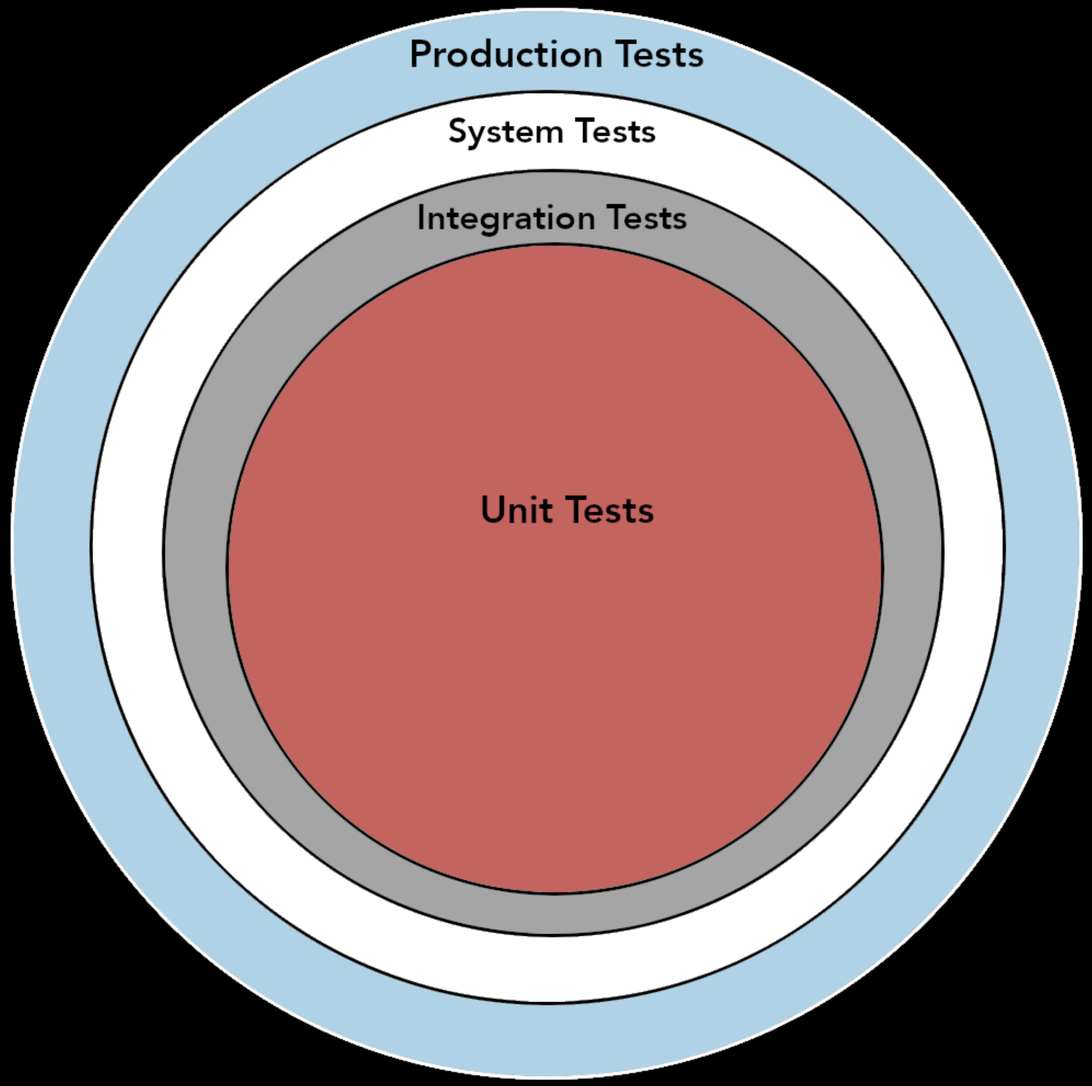
Source: JetBrain's State of Developer Ecosystem Report 2022

HOW TO GET TO A BEST STATE OF TESTING STRANGLER MIGRATION PATTERN

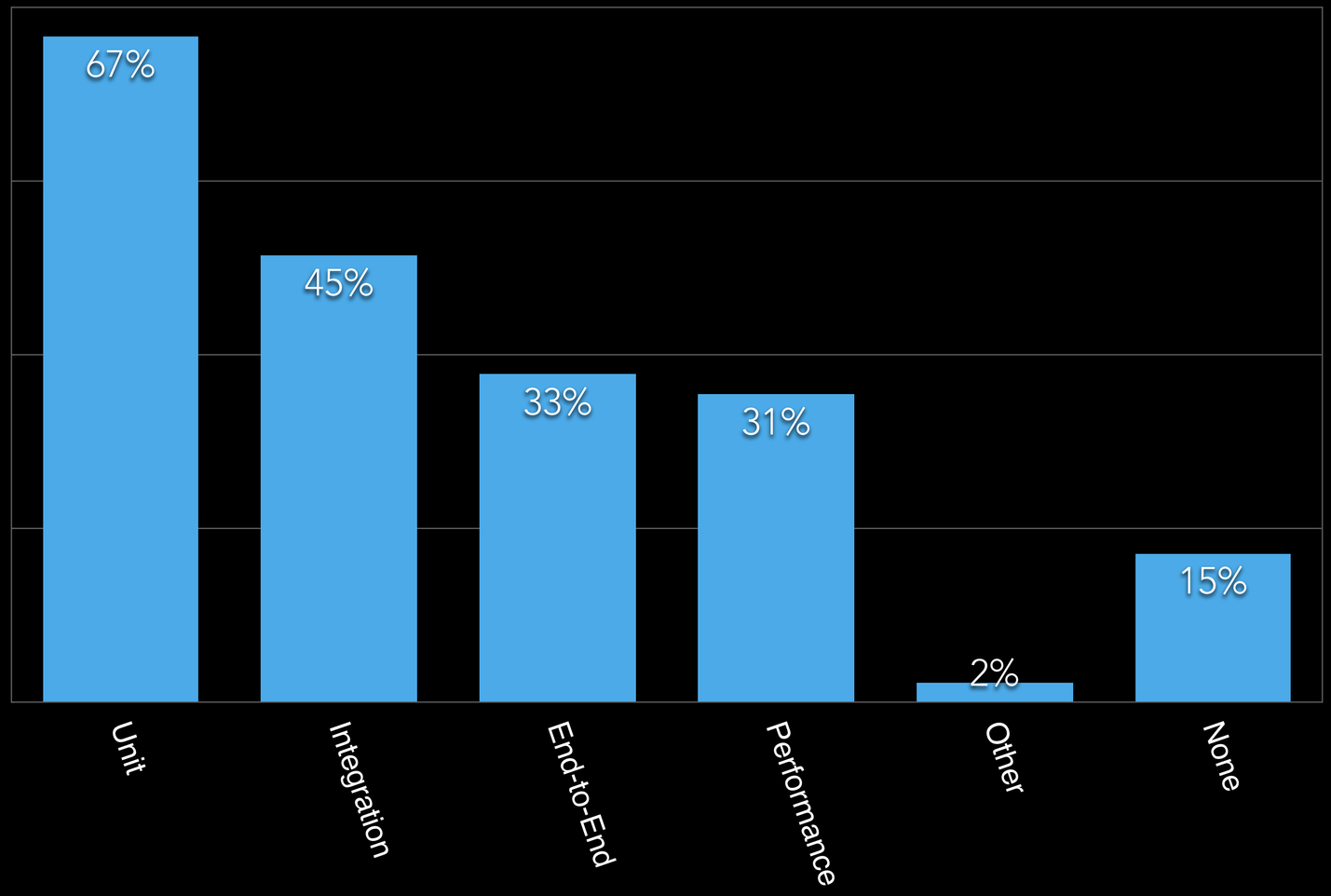


https://learn.microsoft.com/en-us/azure/architecture/patterns/strangler-fig

USING TEST LEVELS AND THE STRANGLER PATTERN



2022 SURVEY TESTS PERFORMED



Source: JetBrain's State of Developer Ecosystem Report 2022

INEXPERIENCED UNIT TESTING

- Hire Interns to do our unit testing
- Have a separate development team do unit tests
- Do unit test in separate sprint/cycle
- Unit test generate tons of output to log
  - Example: Hello from Unit test so and so.
- Running unit test with full debug enabled

SOFTWARE ENGINEERING MANAGEMENT: INTEGRATING SOFTWARE TESTING

South Florida Developer Conference

https://techhubsouthflorida.org/meetups/soflodevcon/

Thank You

www.linkedin.com/in/ealvarez

REFERENCES

- Software Engineering at Google: Lessons Learned from Programming Over TIme (2020)
- https://www.amazon.com/Software-Engineering-Google-Lessons-Programming/dp/1492082791

- The five trademarks of agile organizations (McKinsey & Company)
- https://www.mckinsey.com/capabilities/people-and-organizational-performance/our-insights/the-five-trademarks-of-agile-organizations

- The New New Product Development Game. Harvard Business Review Hirotaka Takeuchi & Ikujiro Nonaka. (1986)
- http://hbr.org/1986/01/the-new-new-product-development-game/

- Critical Chain Eliyahu M. Goldratt. (1997)
- http://www.amazon.com/Critical-Chain-Eliyahu-M-Goldratt/dp/0884271536/

- Object-Oriented Software Engineering: Conquering Complex and Changing Systems (Bernd Bruegge & Allen H. Dutoit)
- https://www.amazon.com/Object-Oriented-Software-Engineering-Conquering-Changing/dp/0134897250

- Flexera State of Tech Spend Report
- https://info.flexera.com/FLX1-REPORT-State-of-Tech-Spend

REFERENCES

- The “Right” CEO: A Corporate Life Cycle Perspective (Aswath Damodaran)
- https://aswathdamodaran.blogspot.com/2021/12/managing-across-corporate-life-cycle.html

- Product Life Cycle by the 280 Group
- https://280group.com/product-management-blog/the-power-of-the-product-lifecycle-a-look-at-the-4-key-stages-of-the-plc/

- Inversion of control software design pattern
- https://en.wikipedia.org/wiki/Inversion_of_control

- ISTQB Software Testing Certification
- https://www.istqb.org/certifications/certification-list/

- Software Static Analysis
- https://www.perforce.com/blog/sca/what-static-analysis

- Software Composition Analysis
- https://fossa.com/blog/framework-for-evaluating-software-composition-analysis-tools/