Enterprise DevOps

Shift left with (test) automation

RAY BERNARDI

Ray Bernardi

Senior Solutions Consultant

Today's Speaker

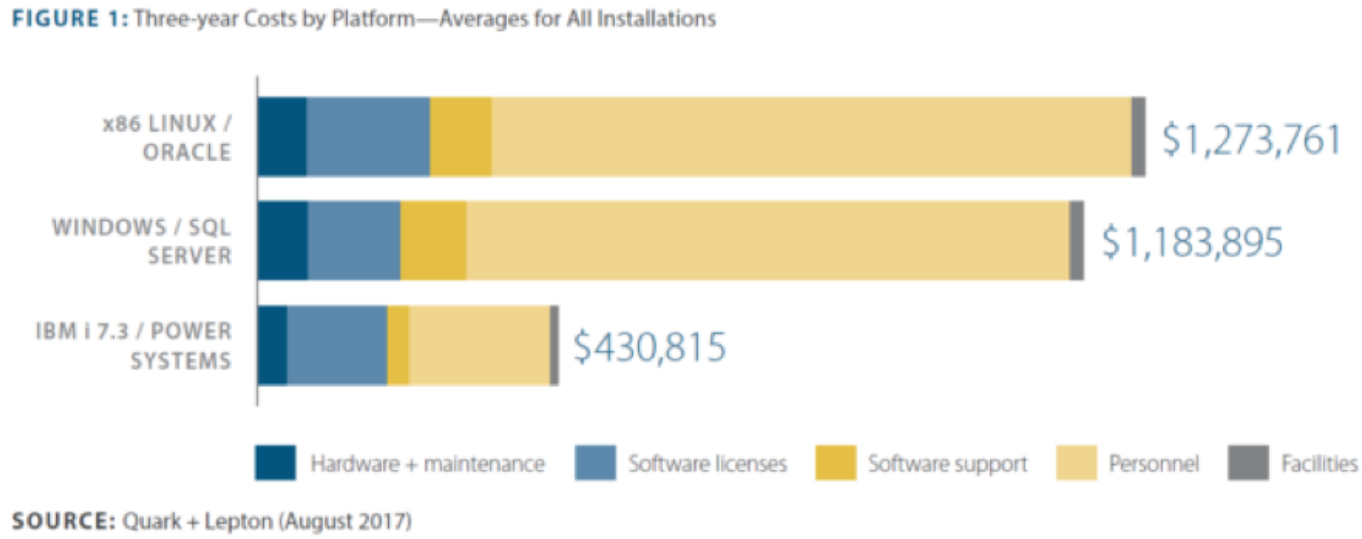
AGENDA

1. State of IBM i Development today
2. Challenges
3. DevOps to the Rescue! Shift Left!
4. Shift Left ROI
5. DevOps Tooling – Automated testing
6. DevOps Transition

Characteristics of IBM i

- Renowned, stable, secure, reliable environment
- Highly affordable (the best TCO in the world*)
- Running business-critical applications

*Quark & Lepton 2017

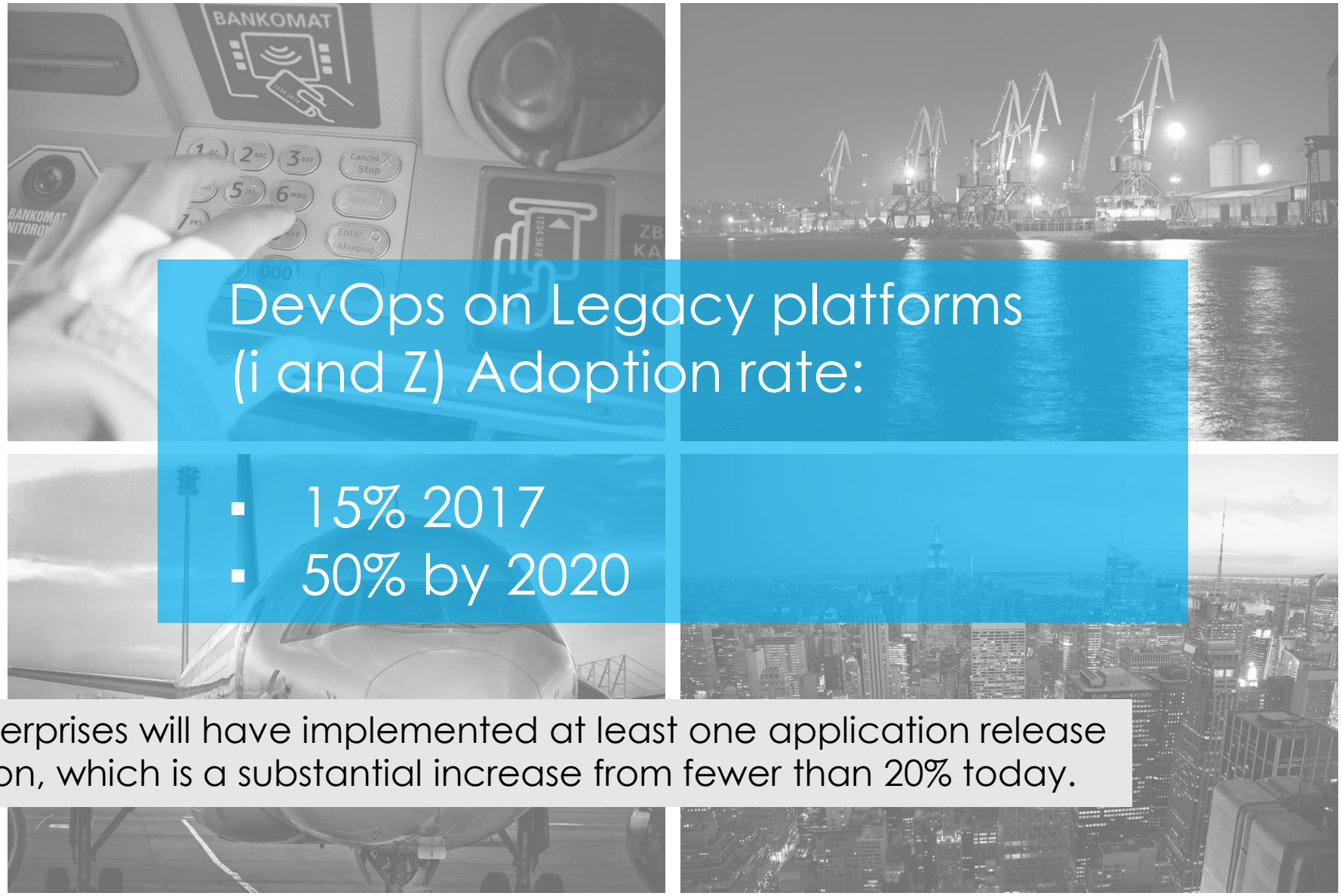


Facts

"Legacy" systems are responsible for >70% of the world's business transactions

Translation…the world runs on COBOL… and RPG…. and this will not change for the foreseeable future…will it be a bottleneck or an asset?

Facts



DevOps on Legacy platforms (i and Z) Adoption rate:

- 15% 2017
- 50% by 2020

By 2023, 75% of global enterprises will have implemented at least one application release orchestration (ARO) solution, which is a substantial increase from fewer than 20% today.

Source: Gartner 2018

Challenges today on IBM i

- Under pressure from digital transformation
  - Web front ends
  - Mobile apps connected to legacy systems
  - Complexity resulting from multiple technologies and tools

- The IT population on IBM i is getting older and older…
  - Existing teams retiring
  - New skills depletion

Bimodal IT

Larger enterprises often face challenges when extending DevOps enterprise-wide:

- Differences in technology cultures between “Systems of Engagement” (SoE) and “Systems of Record” (SoR) reduces DevOps effectiveness overall.
- Each culture has their own tool pipeline with little or no sharing of data.
- Delivery frequency and development speed is often radically different between distributed and legacy teams.

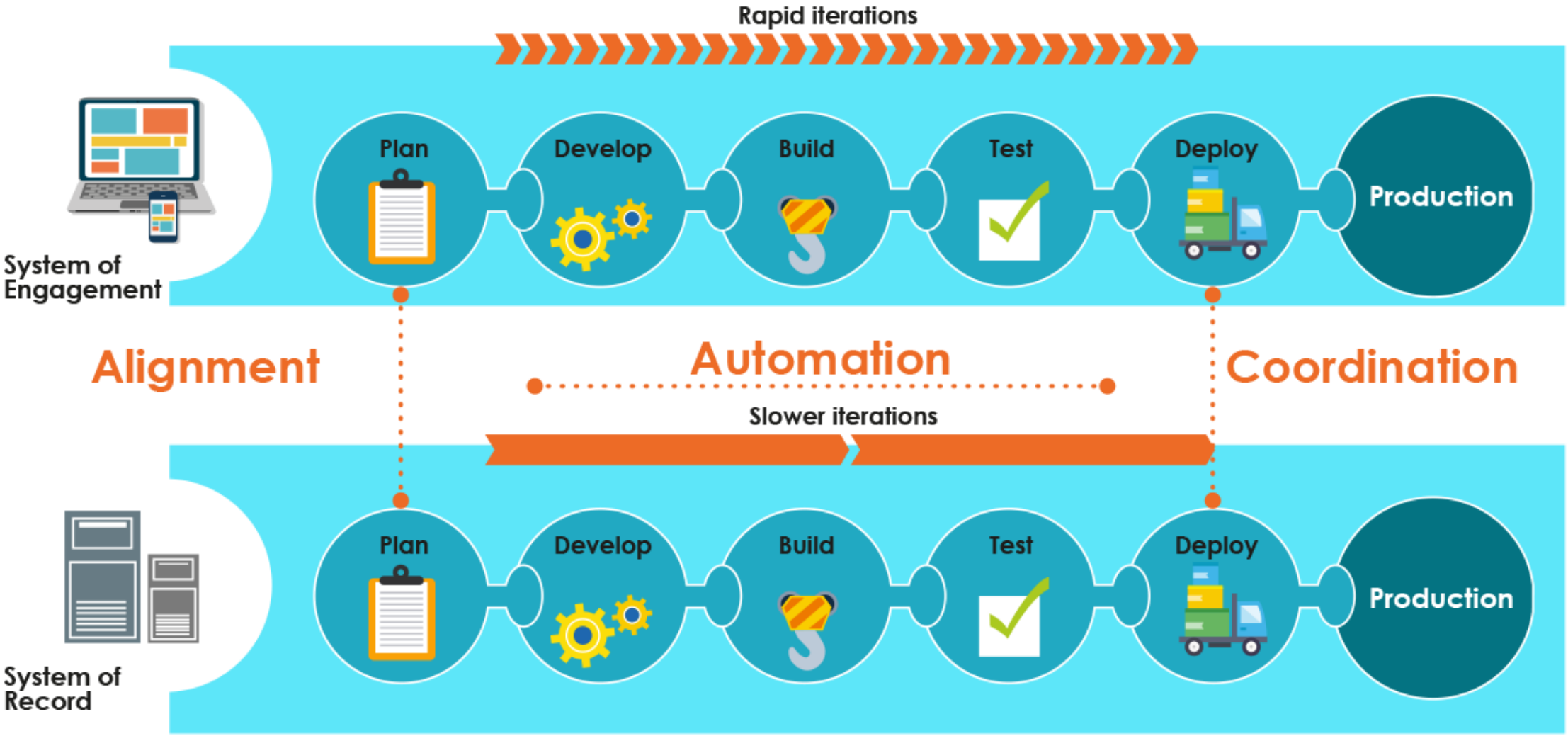
To avoid bottlenecks, DevOps tools must tie ALL these specific technologies together.

Development Meetings?

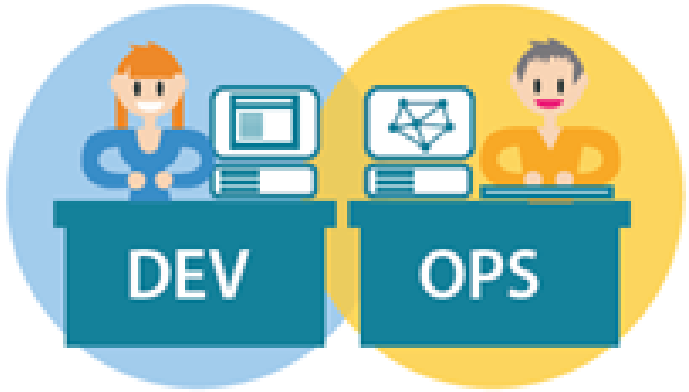
Bimodal IT

	Systems of Record	Systems of Engagement
Applications	Legacy/high volume	Modern/small
Speed of change	Slow	Rapid
Methodology	Waterfall	Agile
Skills	Specialized	"Jack of all trades"
Collaboration	Silos	Collaborative
	Managed by IT department	External ecosystem

Bimodal IT


Rapid iterations
Plan
Develop
Build
Test
Deploy
Production
System of Engagement
Alignment
Automation
Coordination
Slower iterations
Plan
Develop
Build
Test
Deploy
Production
System of Record

DevOps to the Rescue!



ACCELERATE software delivery

BALANCE speed, cost, quality and risk

REDUCE time to customer feedback

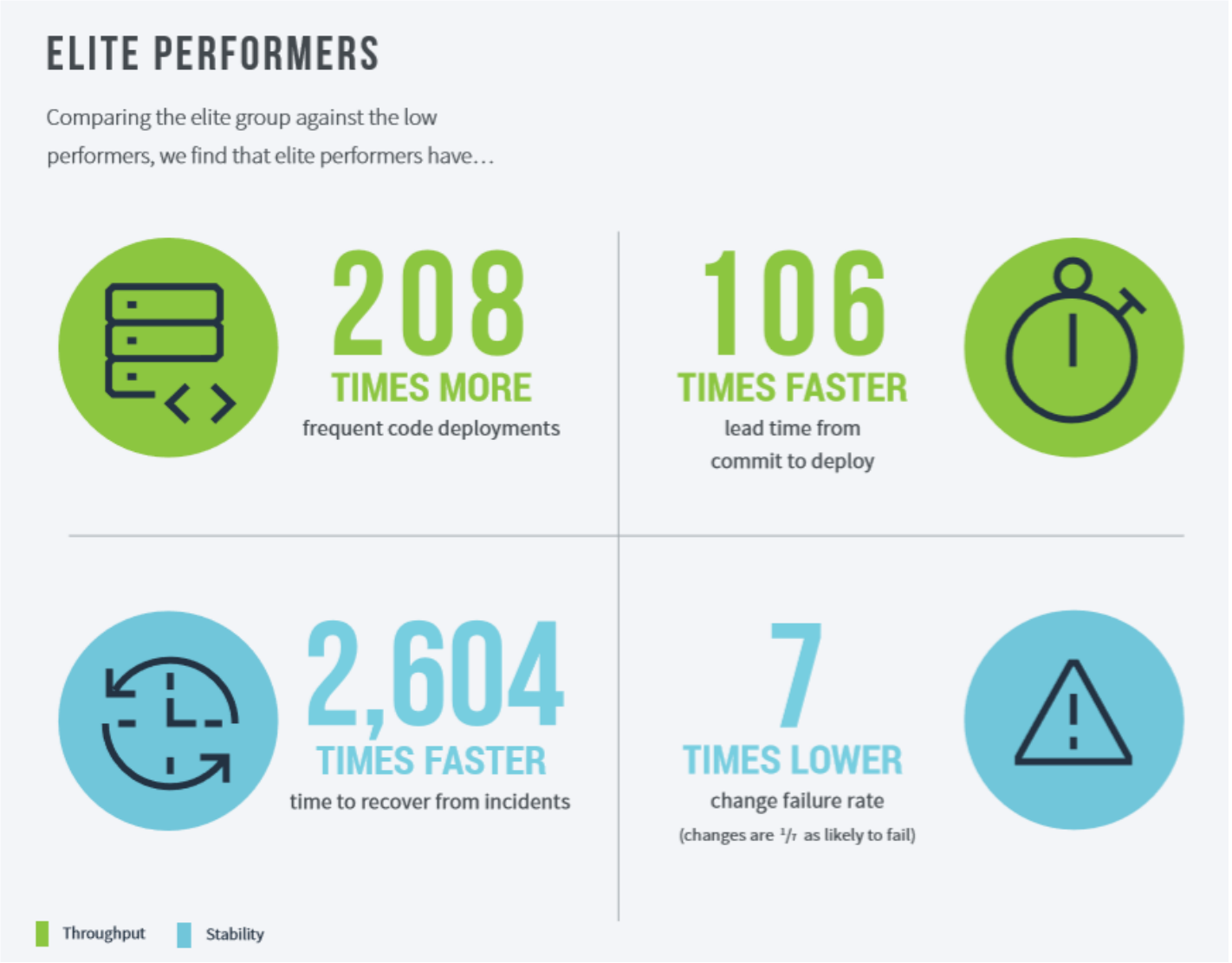
Do not under estimate the impact and value of DevOps

About DevOps Research and Assessment

DevOps Research and Assessment (DORA), founded by Dr. Nicole Forsgren, Jez Humble, and Gene Kim, conducts research into understanding high performance in the context of software development and the factors that predict it. DORA's research over four years and more than 30,000 data points serves as the basis for a set of evidence-based tools for evaluating and benchmarking technology organizations and identifying the key capabilities to accelerate their technology transformation journey.

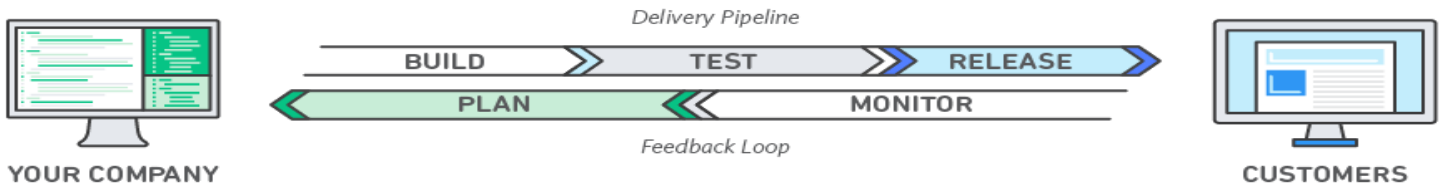
Learn more at devops-research.com.

DORA State of DevOps 2019



What is DevOps?

DevOps is the combination of cultural philosophies, practices, and tools that increases an organization's ability to deliver applications and services at high velocity *{and with higher quality}*:



https://aws.amazon.com/devops/what-is-devops/

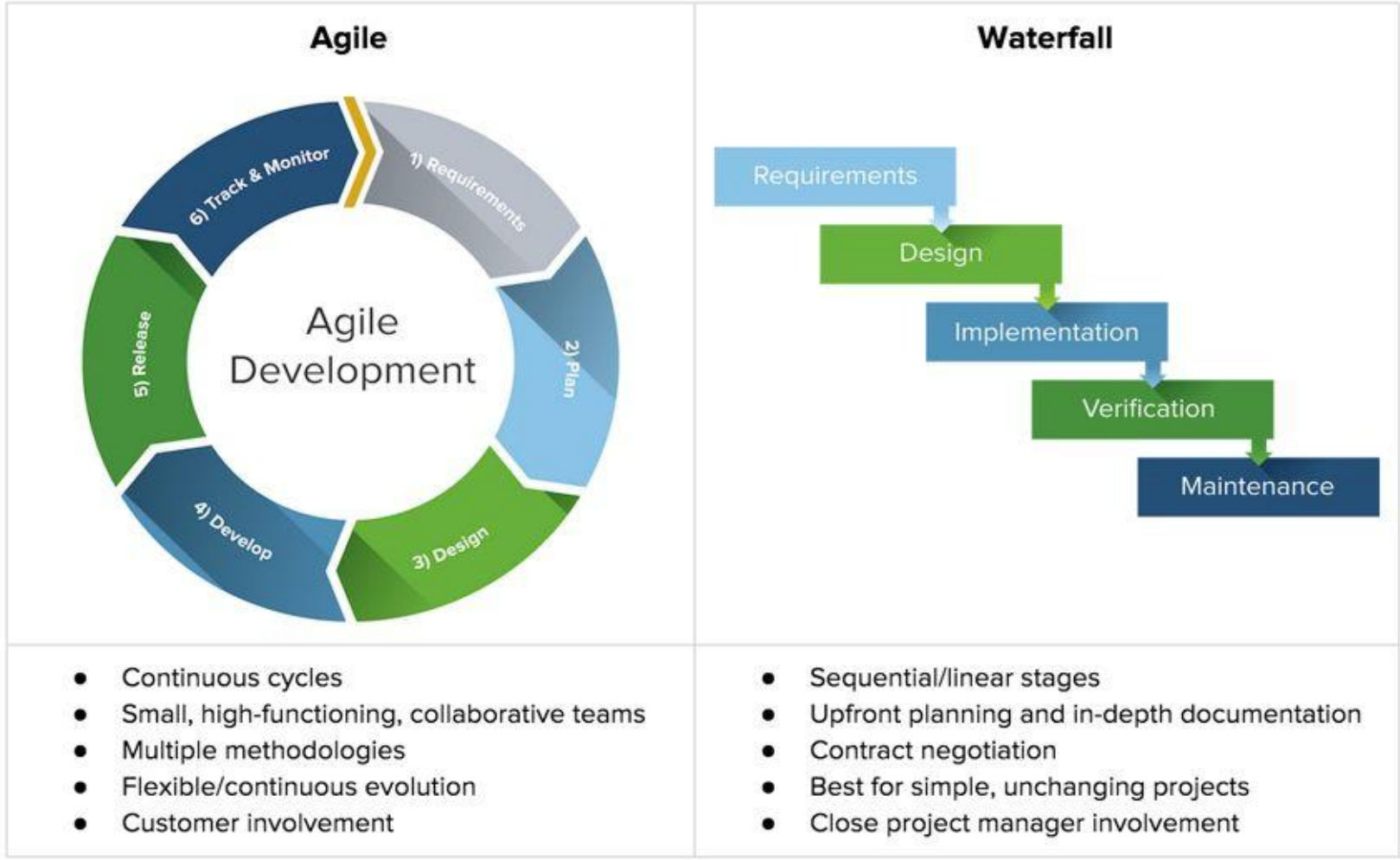
What is SDLC?

The software development life cycle (SDLC) is a framework defining tasks performed at each step in the software development process.

1. Requirements
2. Design
3. Implementation
4. Verification
5. Maintenance

https://www.techopedia.com/definition/22193/software-development-life-cycle-sdlc

SDLC Models

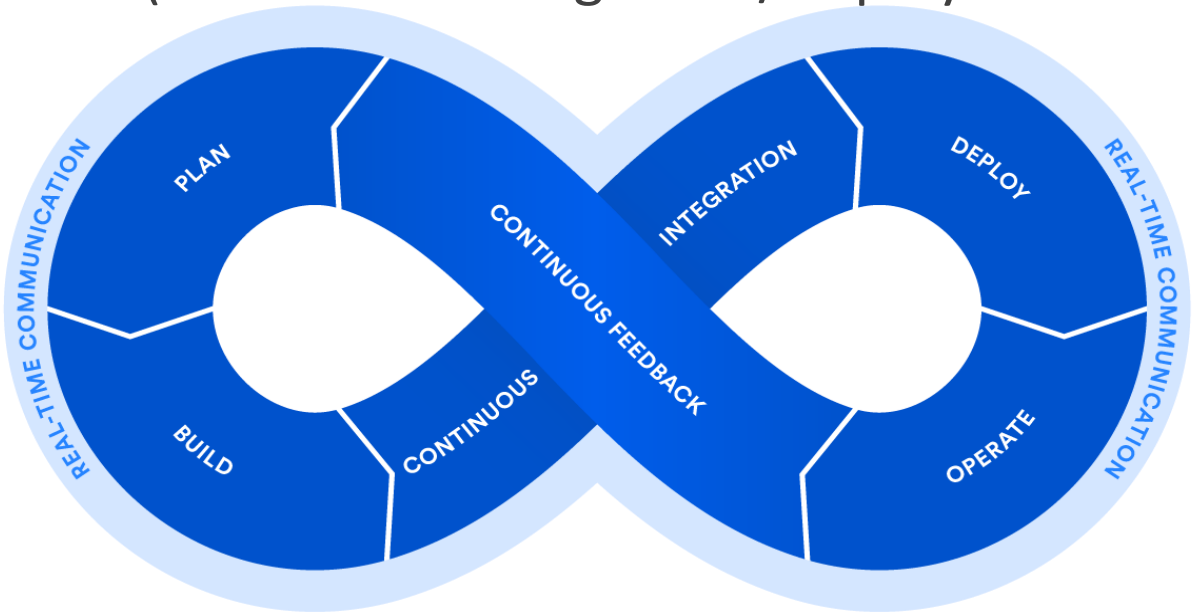


https://www.kovair.com/blog/shifting-left-going-beyond-agile-devops-in-sdlc/

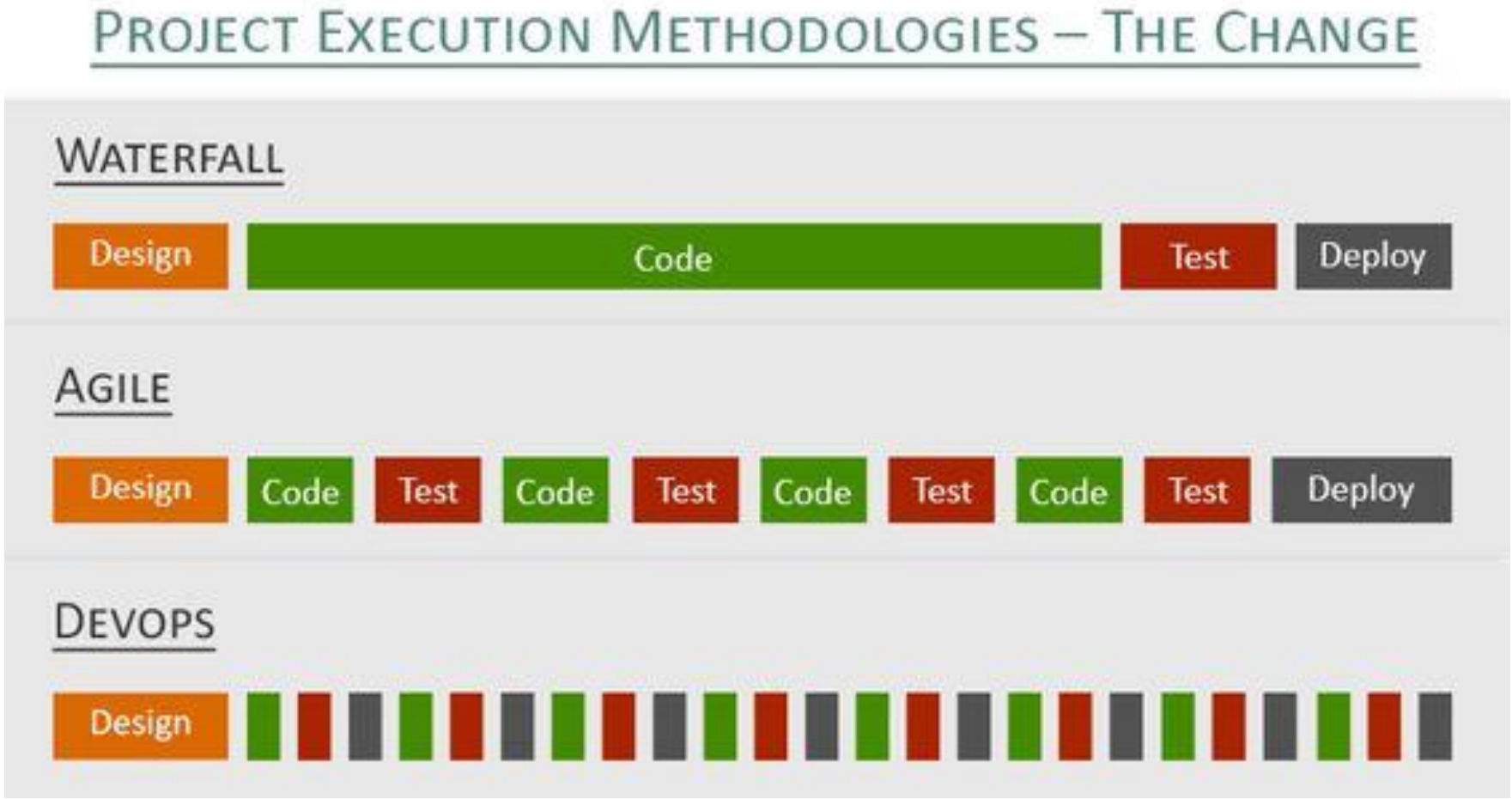
What is DevOps?

DevOps couples Development and Operations, minimizing or eliminating bottlenecks in Agile SDLC:

- ~~User Meetings~~ (Team Collaboration Software, Social Coding)
- ~~Manual processes~~ (Automated Testing)
- ~~Fixed Releases~~ (Continuous Integration/Deployment aka CI/CD)



Waterfall vs Agile vs DevOps

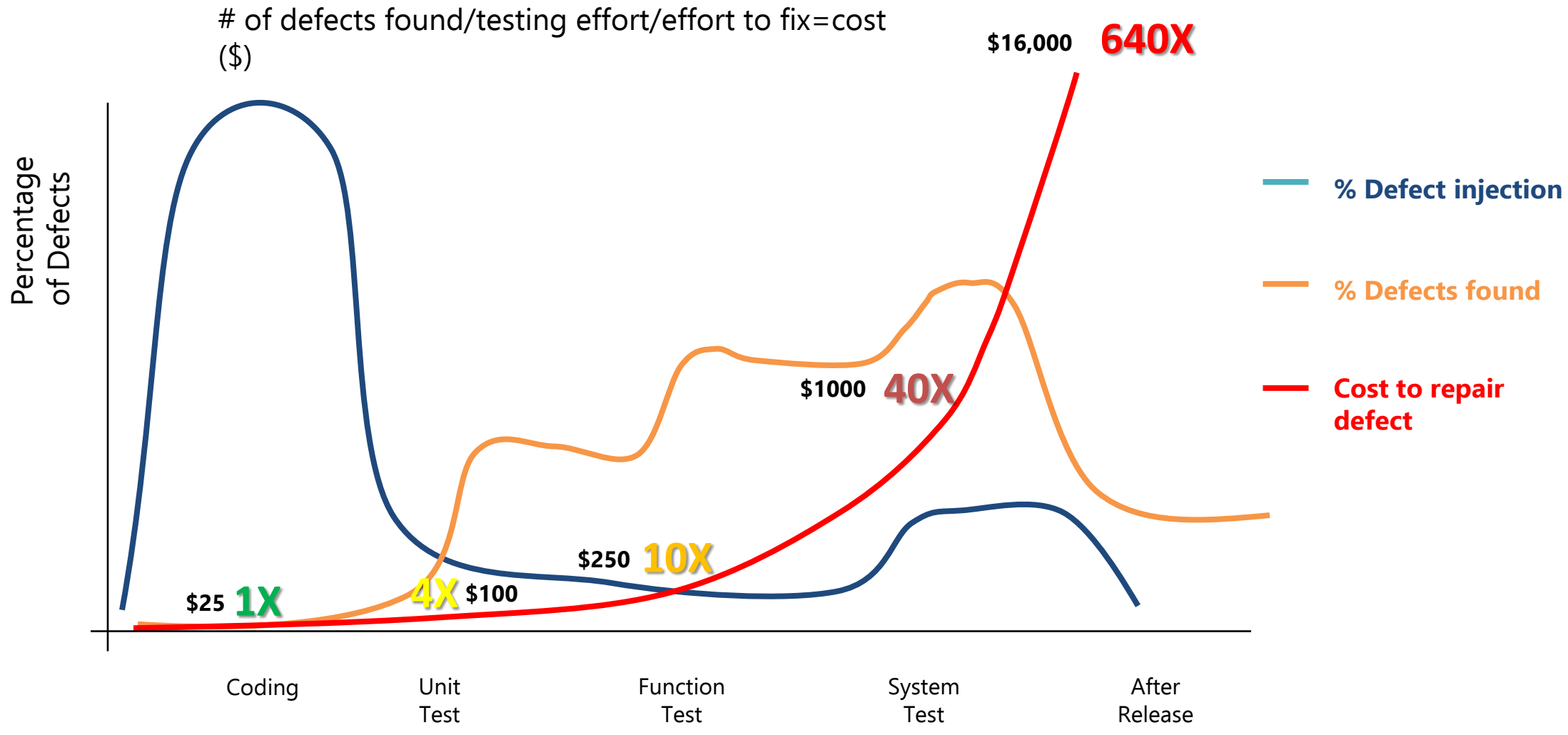


Challenges today on IBM i

- Under pressure from digital transformation
  - Web front ends
  - Mobile apps connected to legacy systems
  - Complexity resulting from multiple technologies and tools

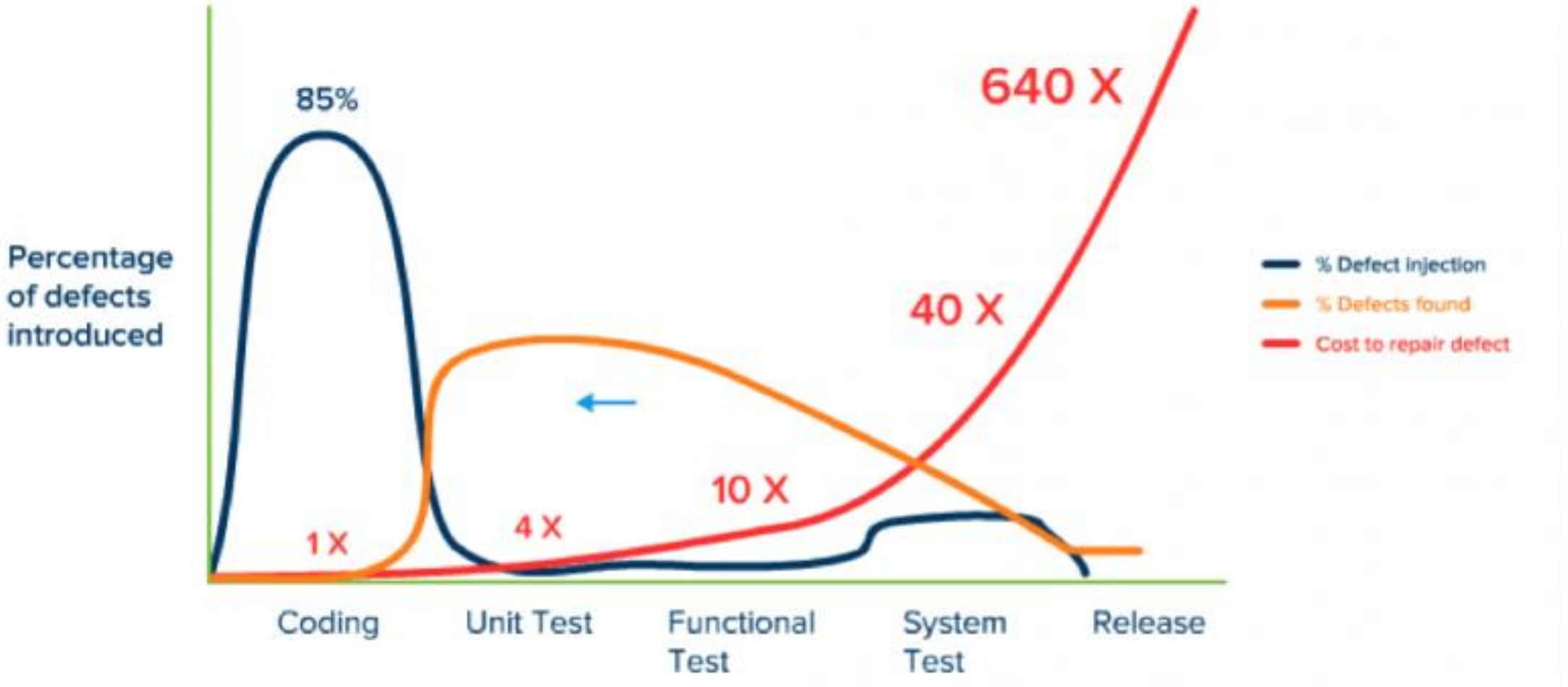
- The IT population on IBM i is getting older and older…
  - Existing teams retiring
  - New skills depletion

Defect resolution.



Jones, Capers. *Applied Software Measurement: Global Analysis of Productivity and Quality*

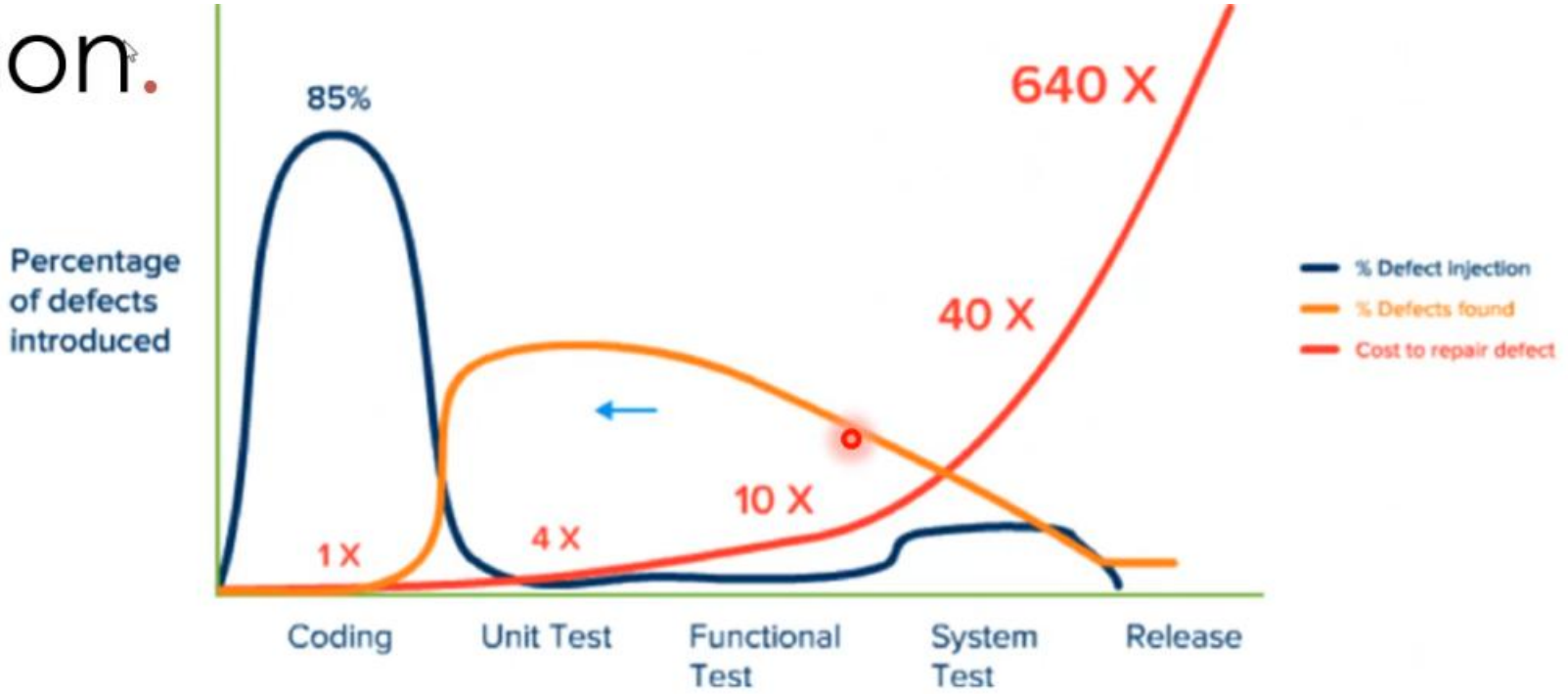
“Shift Left” effect.



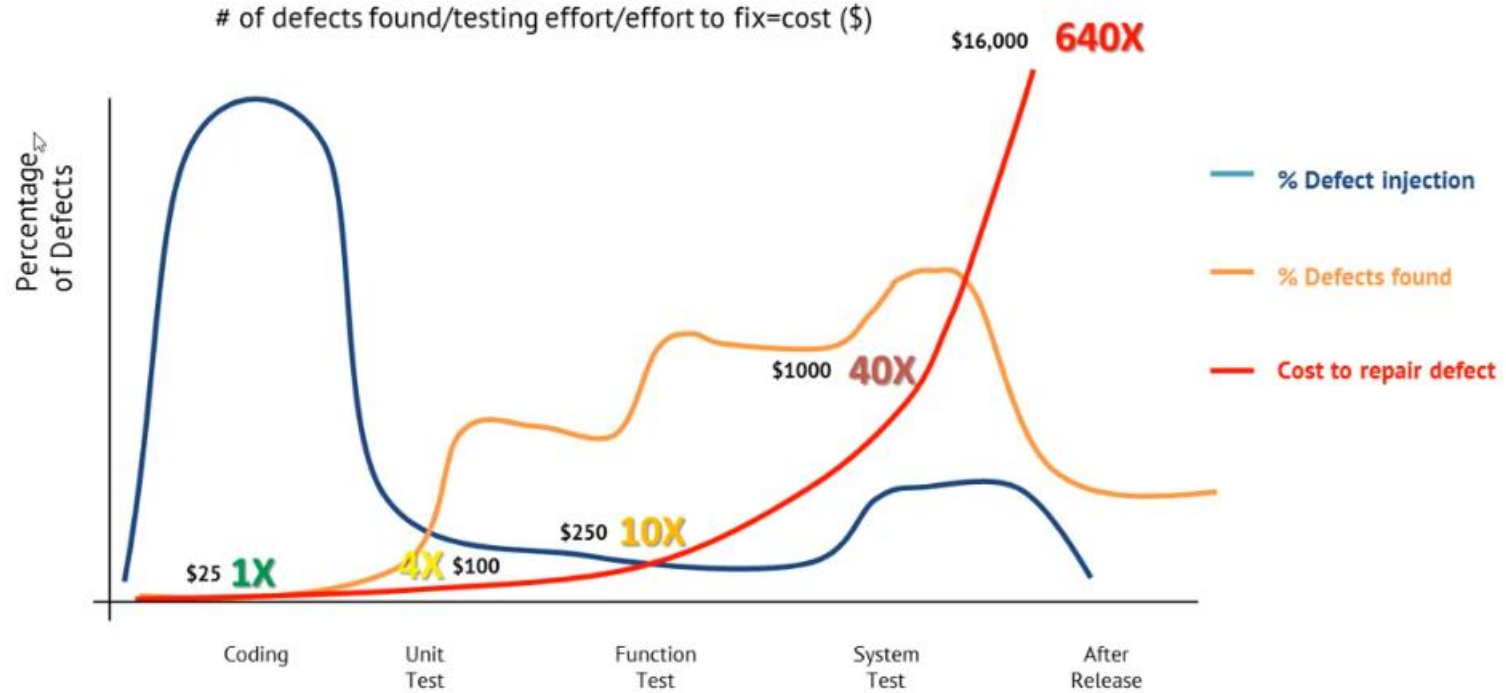
Jones, Capers. *Applied Software Measurement: Global Analysis of Productivity and Quality.*

Defect Comparison.

With DevOps



Without DevOps



Defect Cost Example (per Developer) without DevOps

D = Developer Cost to fix bug

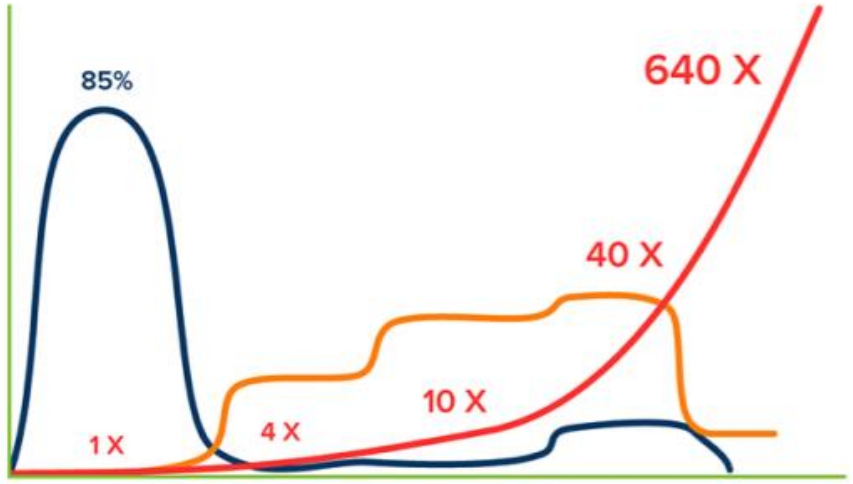
B = Total Bugs/Developer

U ≈ **B** × .20 = Unit Test Bugs

F ≈ **B** × .30 = Function Test Bugs

S ≈ **B** × .40 = System Test Bugs

R ≈ **B** × .10 = Release Bugs



Total Cost = **U**×**D**×4 + **F**×**D**×10 + **S**×**D**×40 + **R**×**D**×640

25

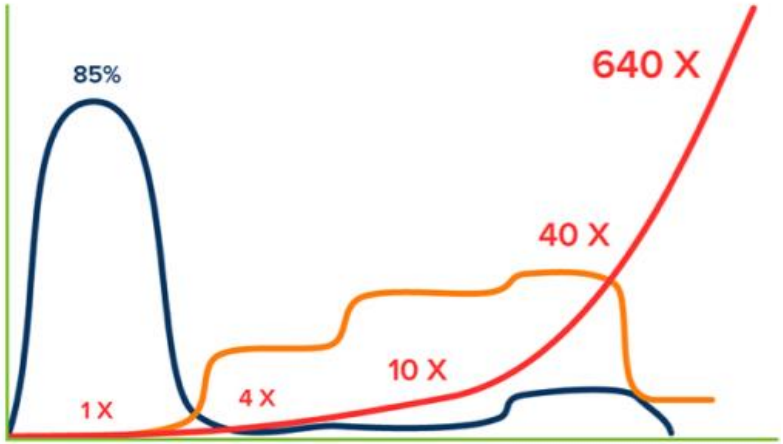
Defect Cost Example (without DevOps)

D=25

1 Hour/Fix

Average Developer salary = $52,000/year

Developer bug cost = **52000÷52÷40 = 25**



B=50

60 Developers

3000 total bugs/year

Each developer fixed **50** bugs/year (**3000÷60 = 50**)

U=10, **F=15**, **S=20**, **R=5**

U=50×.2=10, F=50×.3=15, S=50×.4=20, R=50×.1=5

Total Cost = **U×D×4 + F×D×10 + S×D×40 + R×D×640**

Total Cost = **1000 + 3750 + 20000 + 80000**

Total Defect Cost per Developer = $104,750

26

Defect Cost Example (per Developer) with DevOps

D = Developer Cost

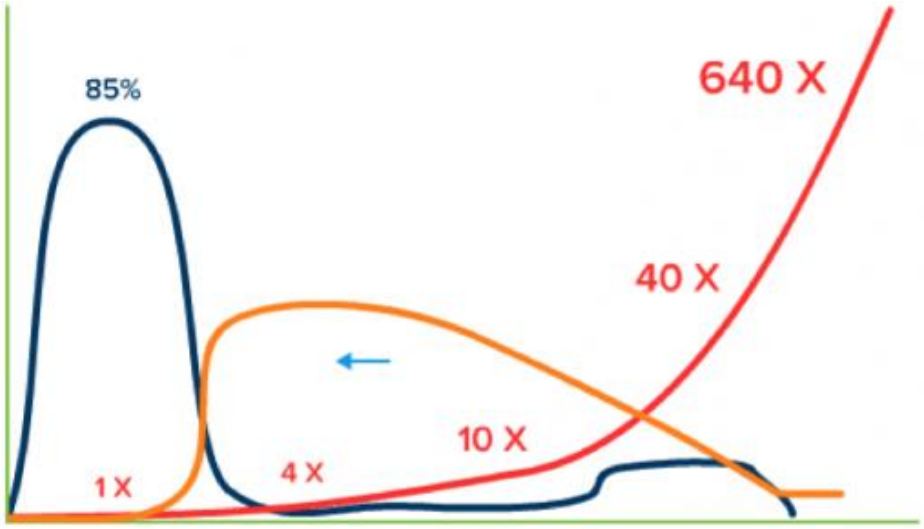
B = Total Bugs per Developer

U ≈ **B** × .60 = Unit Test Bugs

F ≈ **B** × .30 = Function Test Bugs

S ≈ **B** × .15 = System Test Bugs

R ≈ **B** × .05 = Release Bugs



Total Cost = **U**×D×4 + **F**×D×10 + **S**×D×40 + **R**×D×640

27

Defect Cost Example (with DevOps)

D=25

1 Hour/Fix

Average Developer salary = $52,000/year

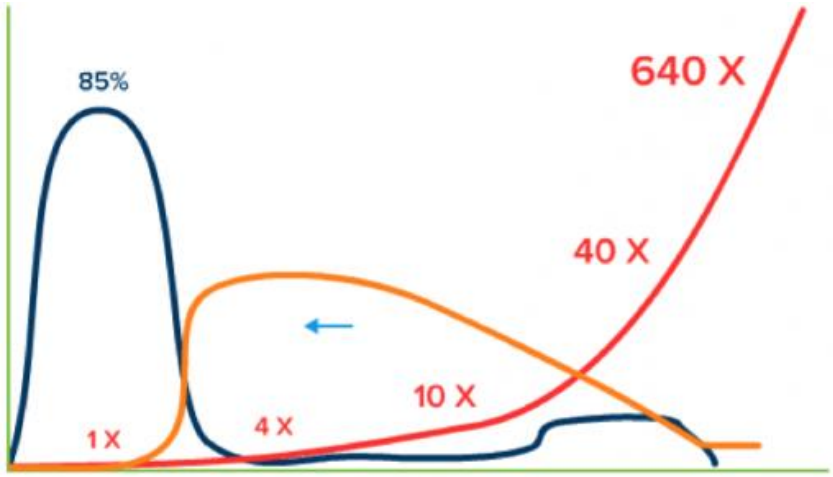
Developer bug cost = **52000÷52÷40 = 25**

B=50

60 Developers

3000 total bugs/year

Each developer fixed **50** bugs/year (**3000÷60 = 50**)



U=30, **F=12.5**, **S=7.5**, **R=2.5**

U=50×.6=30, F=50×.25=12.5, S=50×.15=7.5, R=50×.05=2.5

Defect Cost = **U×D×4 + F×D×10 + S×D×40 + R×D×640**

Defect Cost = **3000 + 3125 + 7500 + 40000**

Total Defect Cost per Developer = $53,625

28

DevOps ROI

Defect Savings = $DefectCost_{wo} - DefectCost_{DO}$
= 104750-53625
= 51125

Total Defect Saving = 51125×60 Developers

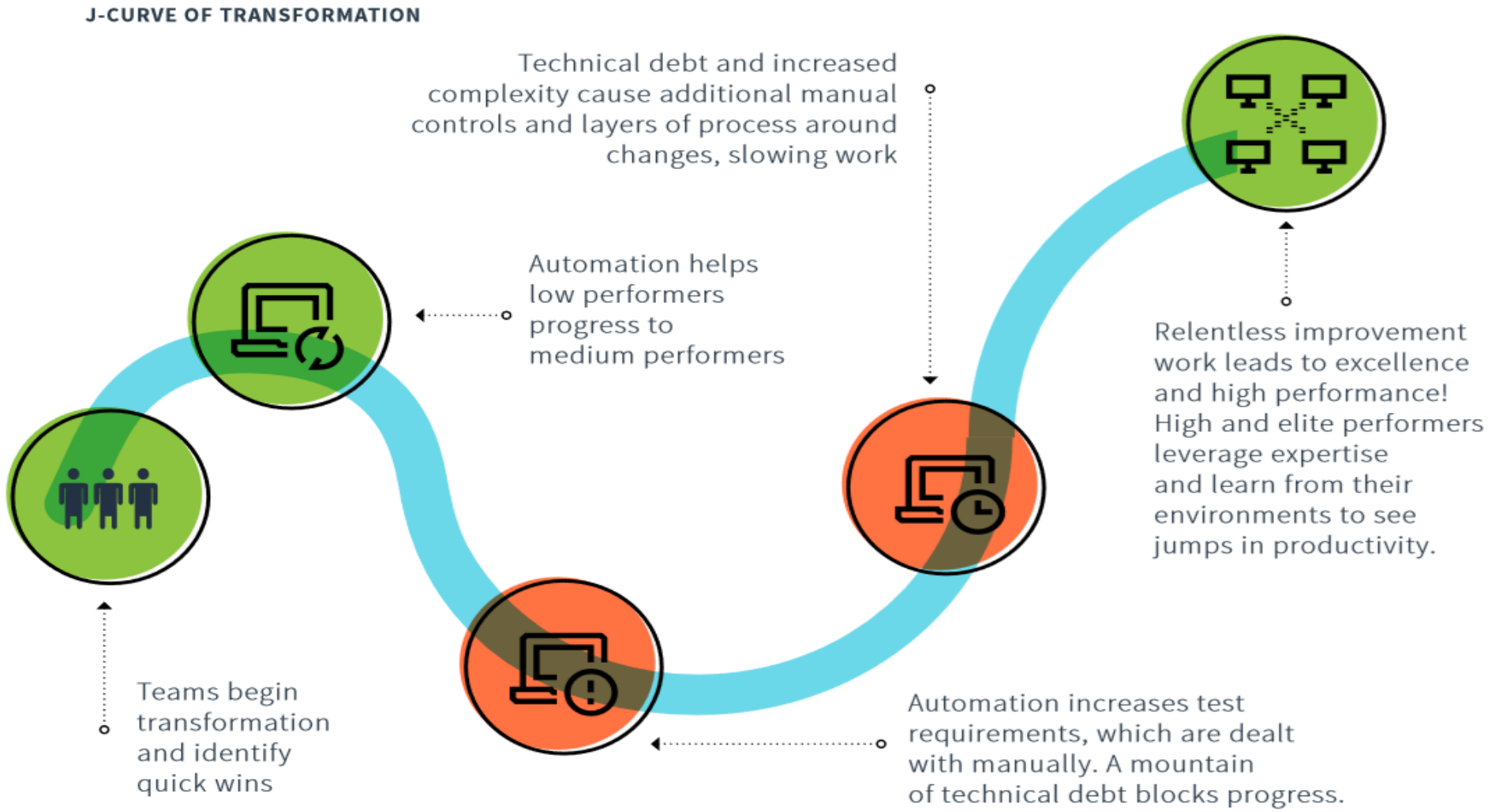
Total Defect Saving = $3,067,500

ROI = Total Defect Savings - Infrastructure - Implementation

!!! EVERY YEAR !!!

29

DevOps J-Curve



DevOps requirements

DevOps is not just tooling...

DevOps is a changed mind set
How can we quickly, and safely, deliver features to end-users

Developers, Operations and end-users are in constant communication

"Big Bang" (Waterfall) Versions are archaic
"Feature releases" (Agile) instead...

DevOps require Application ~~Modernization~~ Modularization
Microservices
TCD (Test Centric Development) i.e. Scriptable test cases

...but, tools do make things so much easier...

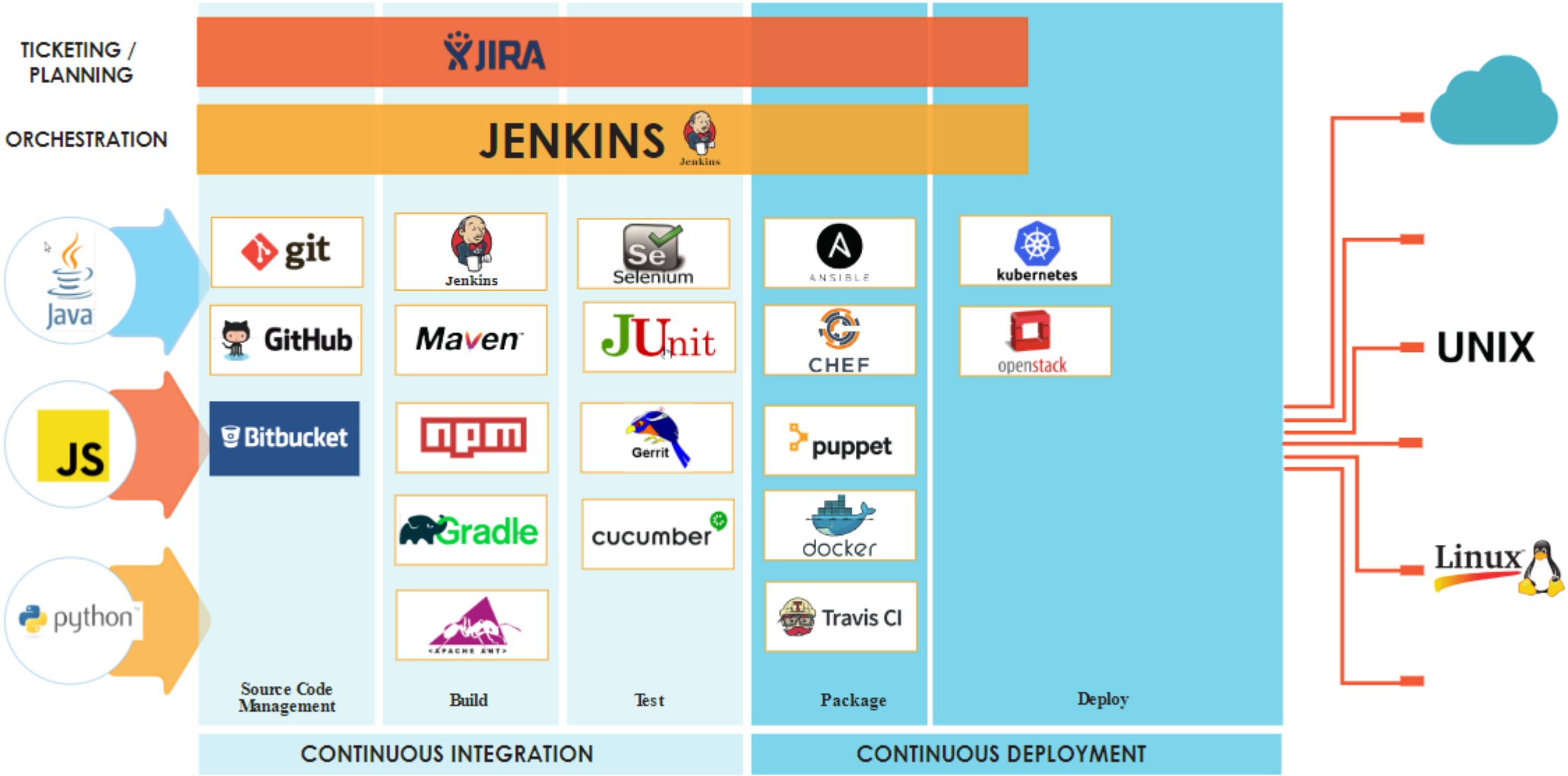
DevOps tooling

Standardization

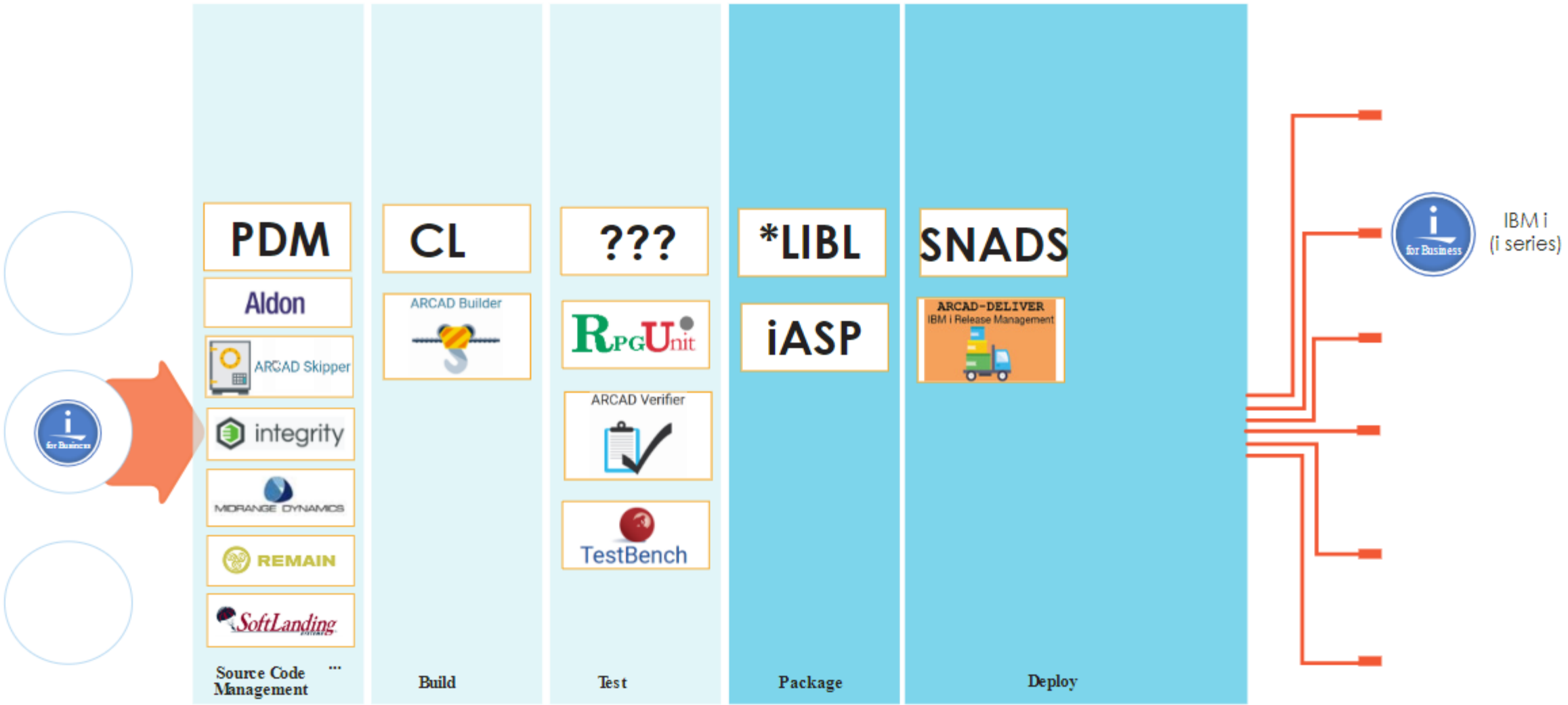
- Easier to onboard and acquire new talent
- Less software license and maintenance costs moving forward
- Shorter lifecycles
- Quicker feedback loops
- More responsive to the business stakeholders.

The end business goal is to drive more business or services with interaction through interfaces that people wish to use.

Open Source DevOps

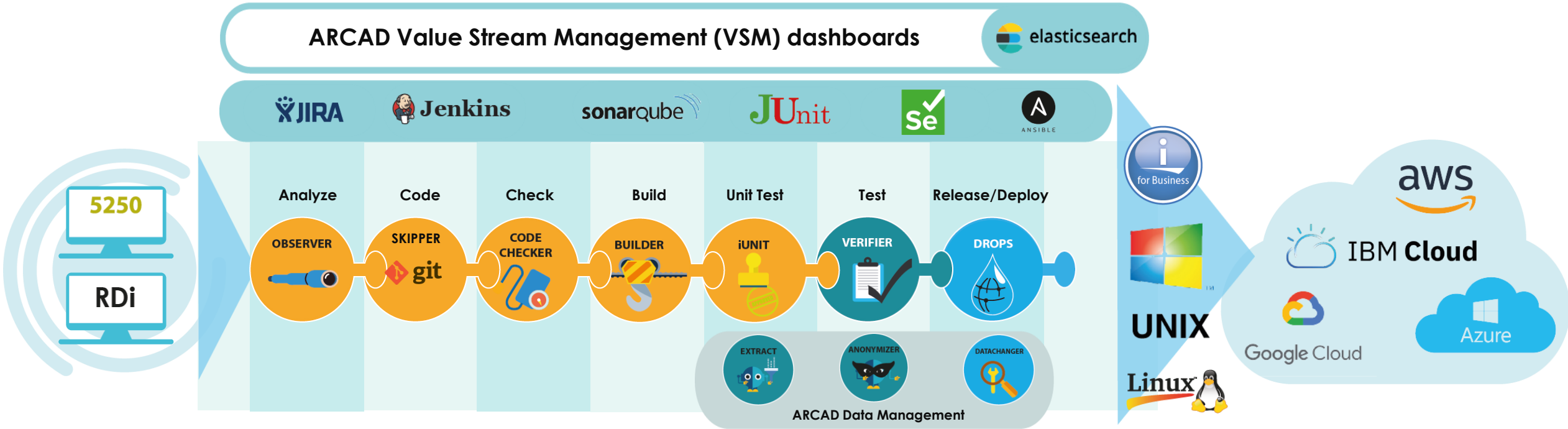


"Legacy" IBM i Development

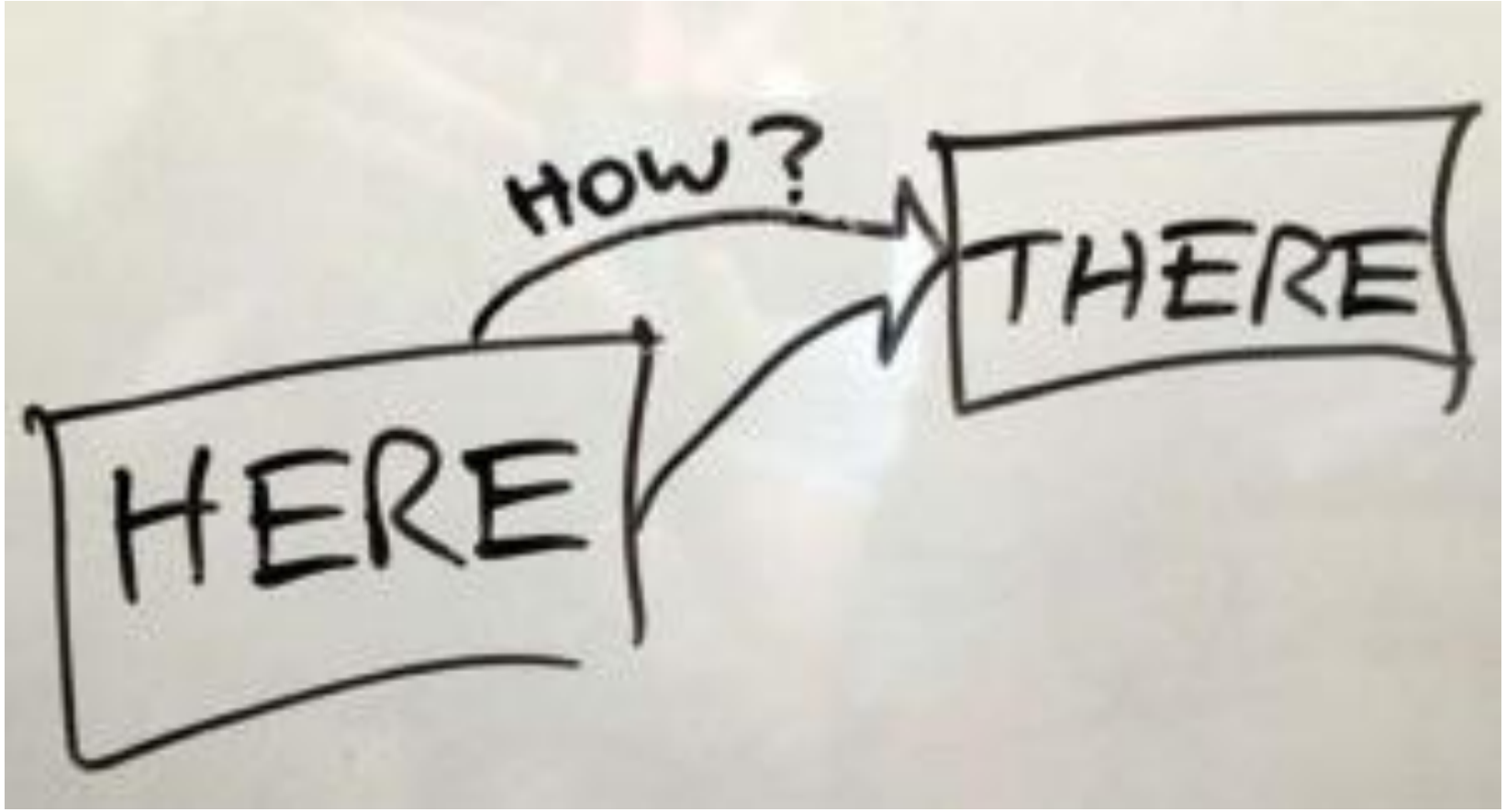


Enterprise DevOps

ARCAD for DevOps:
Continuous software delivery on IBM i.



The $M question...

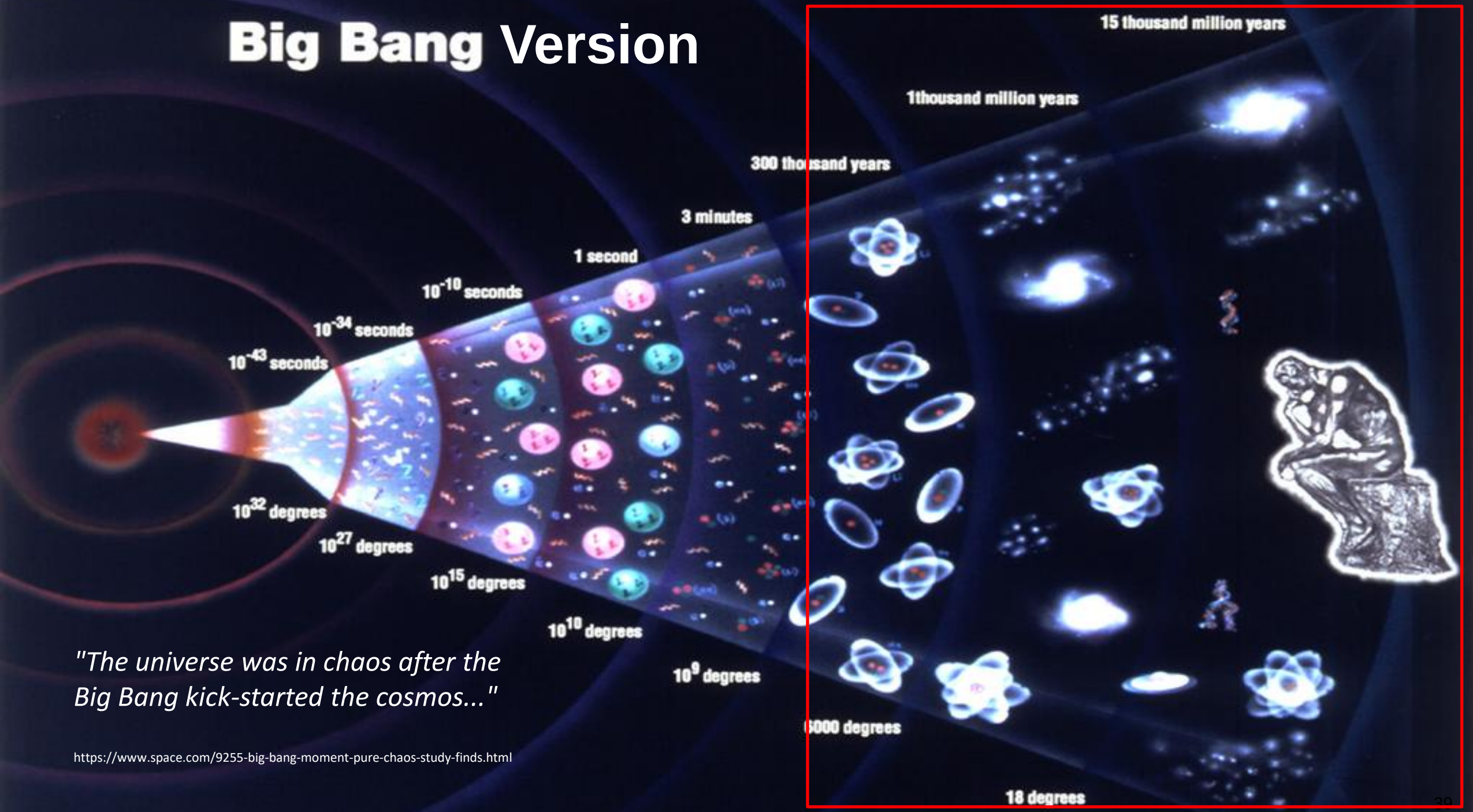


http://cultivatemarketing.co.uk/content-strategy-2/

DevOps Transition

- Lay the Foundation
  - Analyze your existing resources
    - Skillset, methodology, existing tooling
    - Identify "Champions"
  - Pick tools that match your needs and experience
  - Document
    - **5733-AO1 Observer for IBM i**
  - Begin to install DevOps tooling
- Make incremental Development changes
  - Start using RDi (if you aren't already)
  - Convert RPGIII to RPGLE
    - **5733-AC1 RPG Converter for IBM i**
  - Social Coding
  - Microservices
  - Test Centric Development
  - "Big Bang DevOps" Version isn't necessary

Big Bang Version



"The universe was in chaos after the Big Bang kick-started the cosmos..."

https://www.space.com/9255-big-bang-moment-pure-chaos-study-finds.html

SUMMARY

1. Significant Development Challenges exist today
2. DevOps ROI is Proven, Measurable
3. DevOps is a Journey
4. DevOps is (or should be Strategic


Questions?.

Thank You!

- Ray Bernardi
- Senior Solutions Consultant