

SESSION



TestingUy

Shift Left your Performance Testing

HARI KRISHNAN

harikrishnan83@gmail.com

@harikrishnan83

August 2-7, 2021

testinguy.org

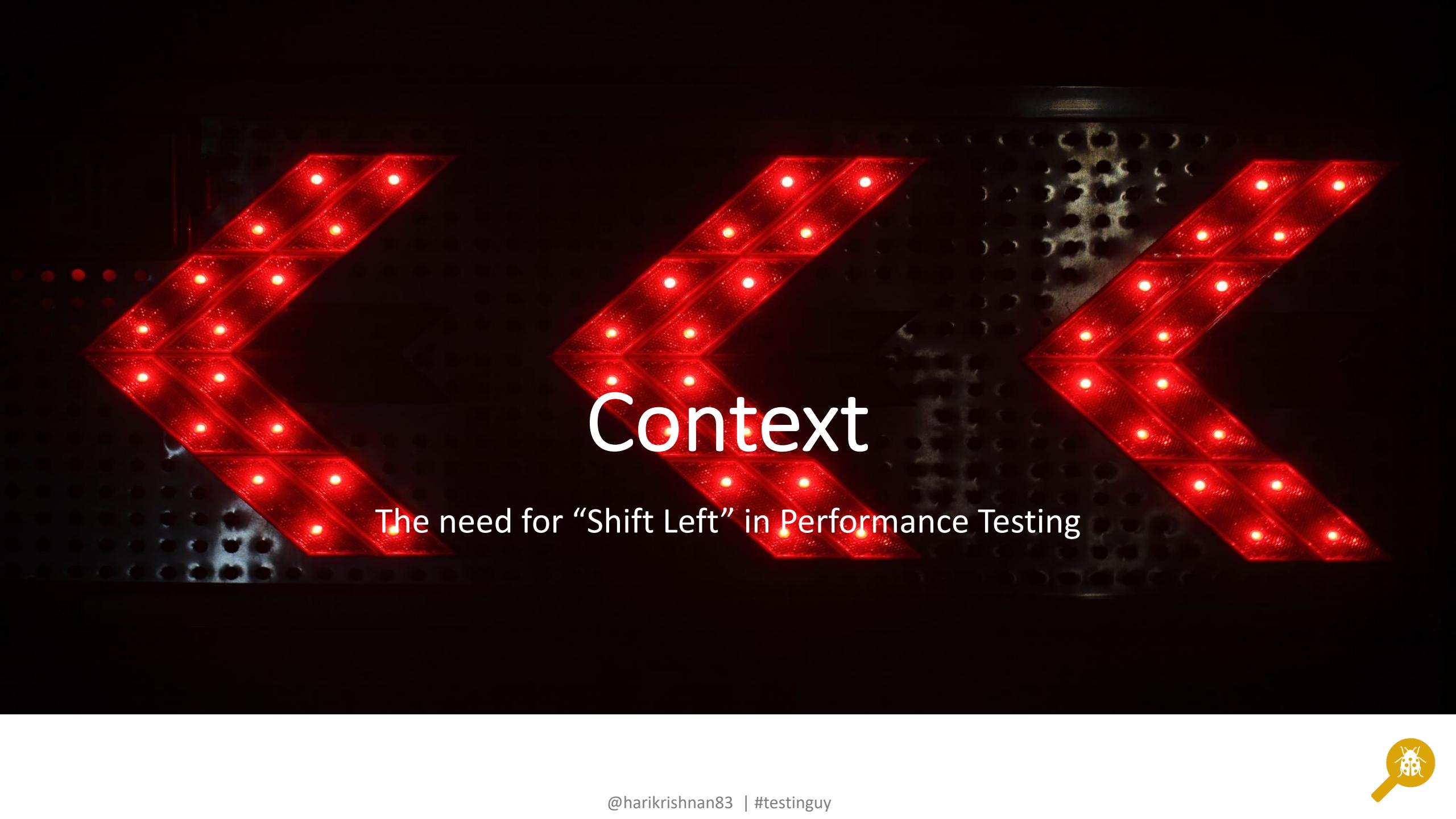
@testinguy | #testinguy

About me

Consultant and Coach

- Cloud Transformation
- Extreme Programming
- Agile and Lean





Context

The need for “Shift Left” in Performance Testing

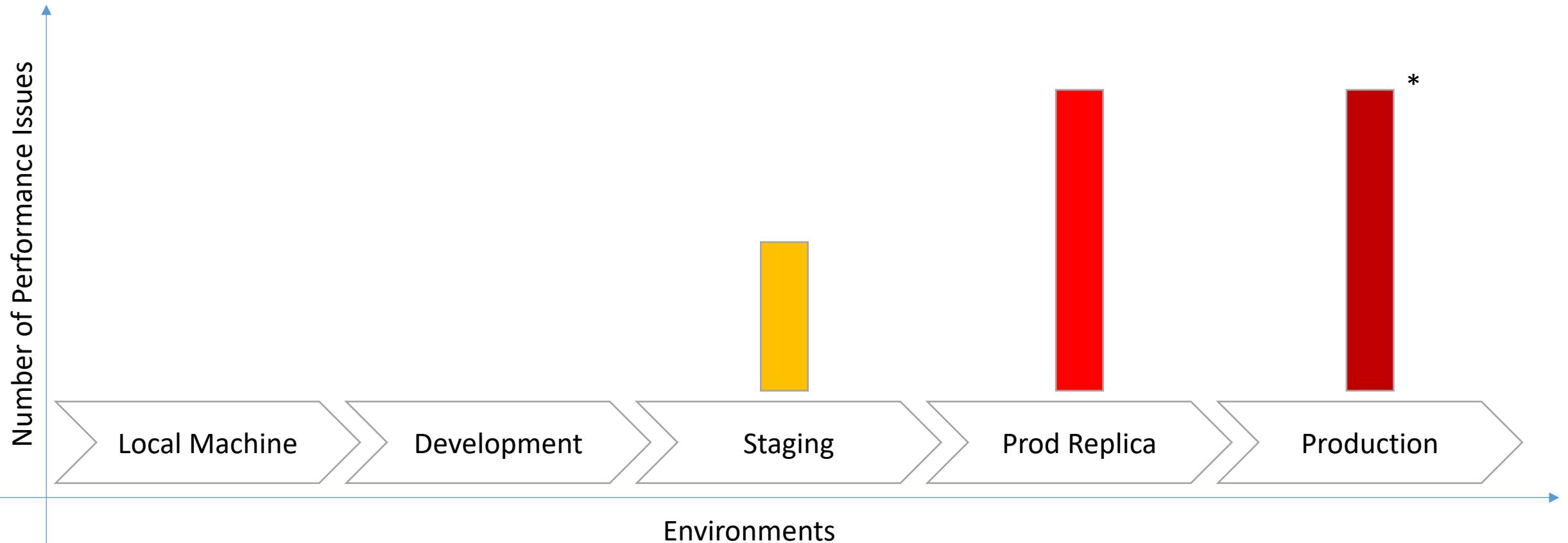


Show of hands

In which Environment do you identify Performance Issues?



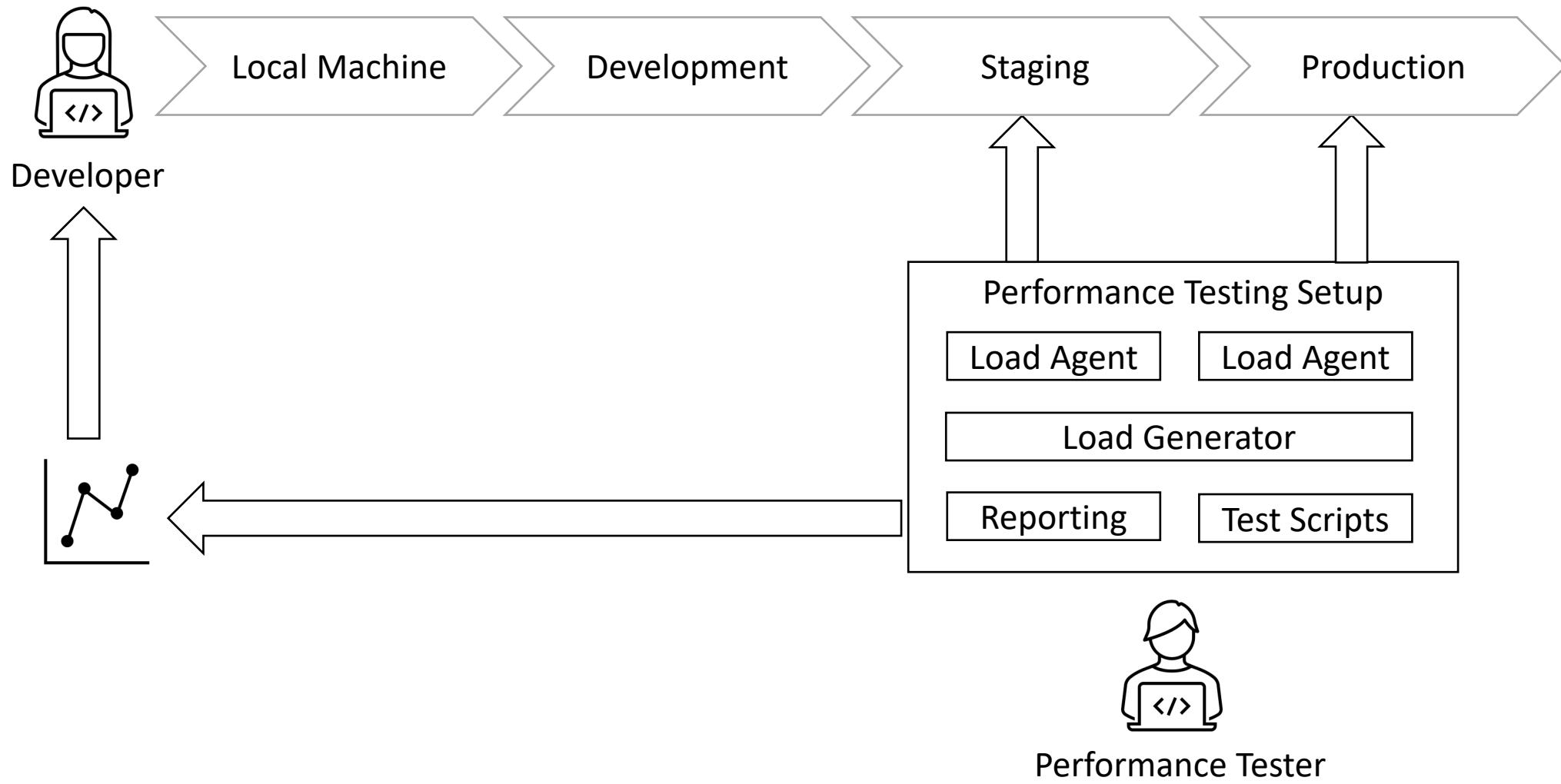
In which Environment do you identify Performance Issues?



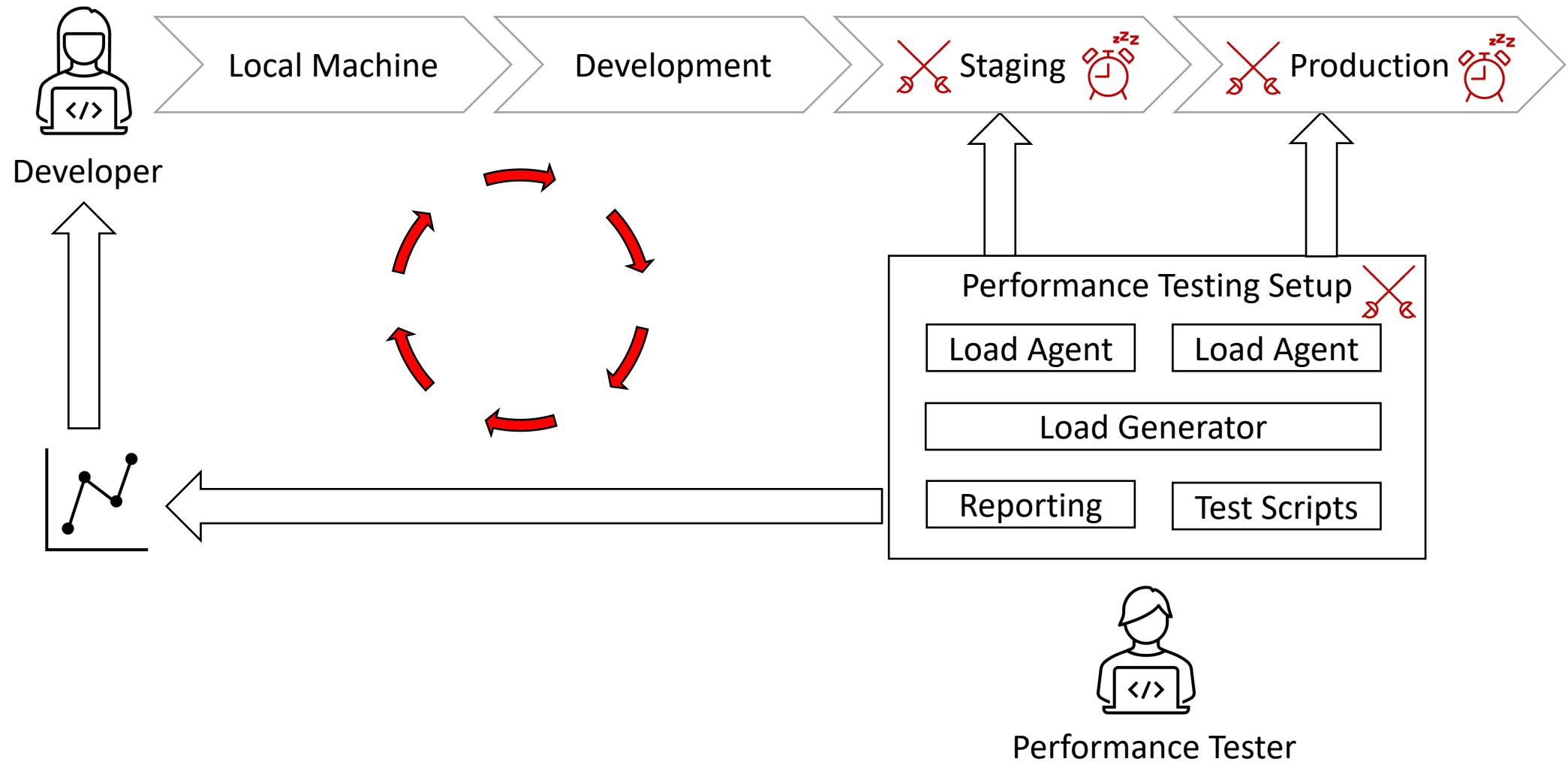
* **Legend:** Color represents Mean Time To Resolution, Darker the shade of Red, longer it takes to fix



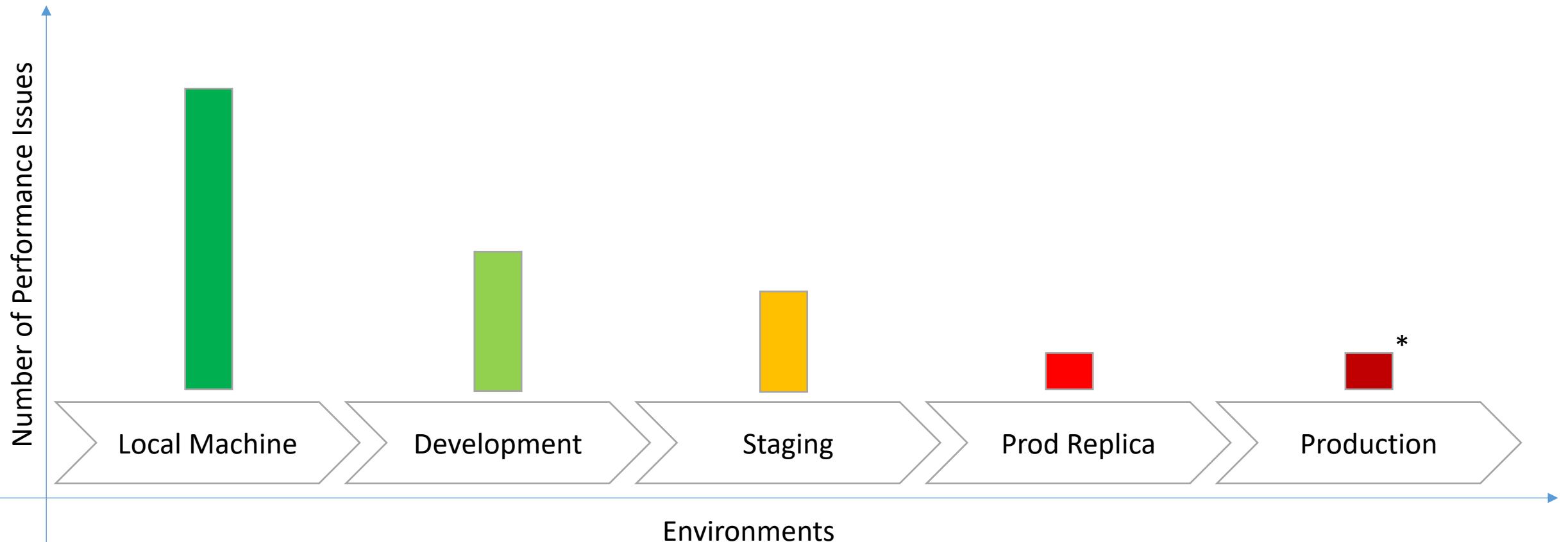
Perf Testing – The Usual Setup



Perf Testing – Usual Setup Issues



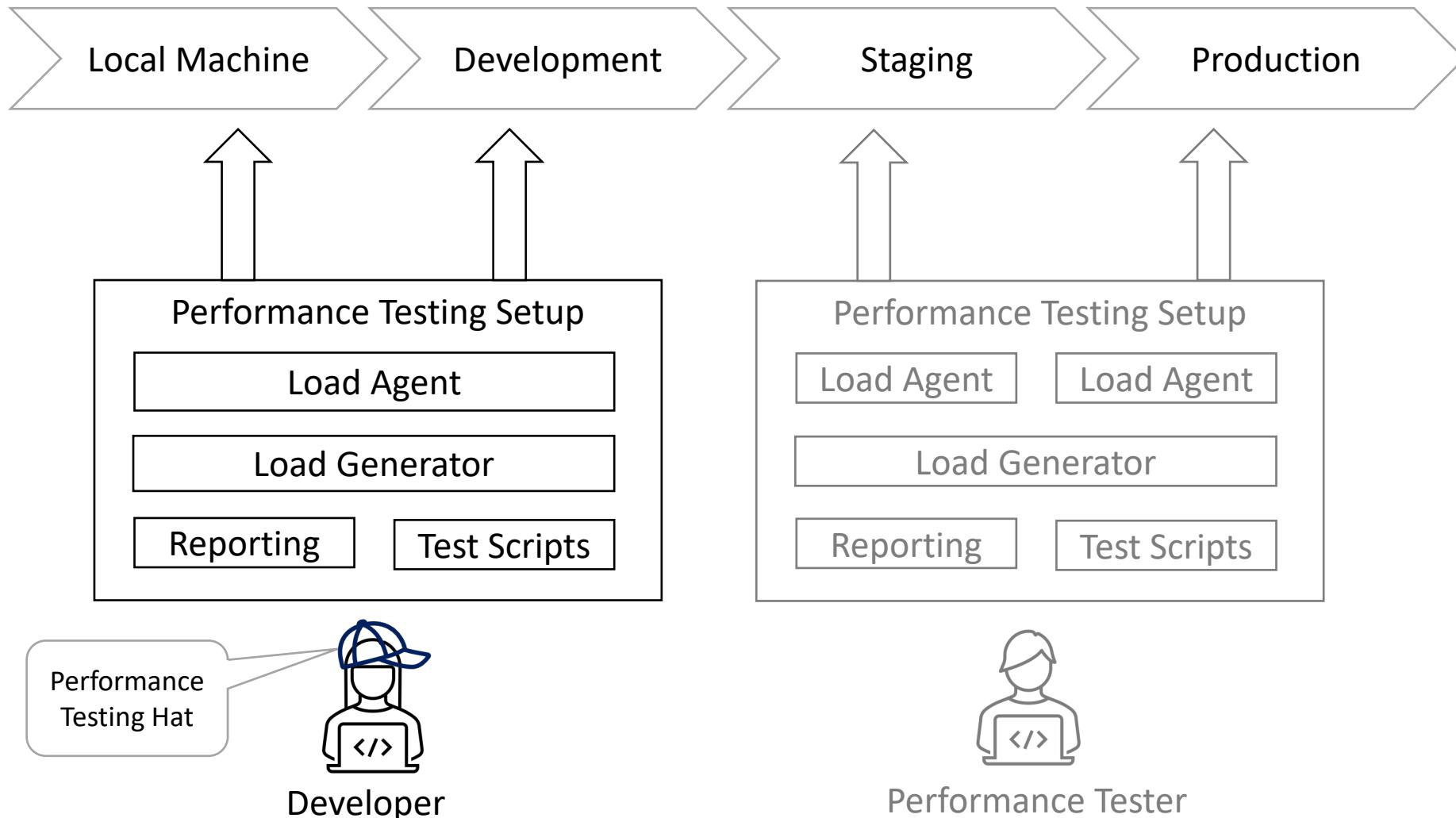
What it should look like instead



* **Legend:** Color represents Mean Time To Resolution. Green represents quickest resolution. Darker the shade of Red, longer it takes to fix.



Running Perf Test Early in your Development Cycle



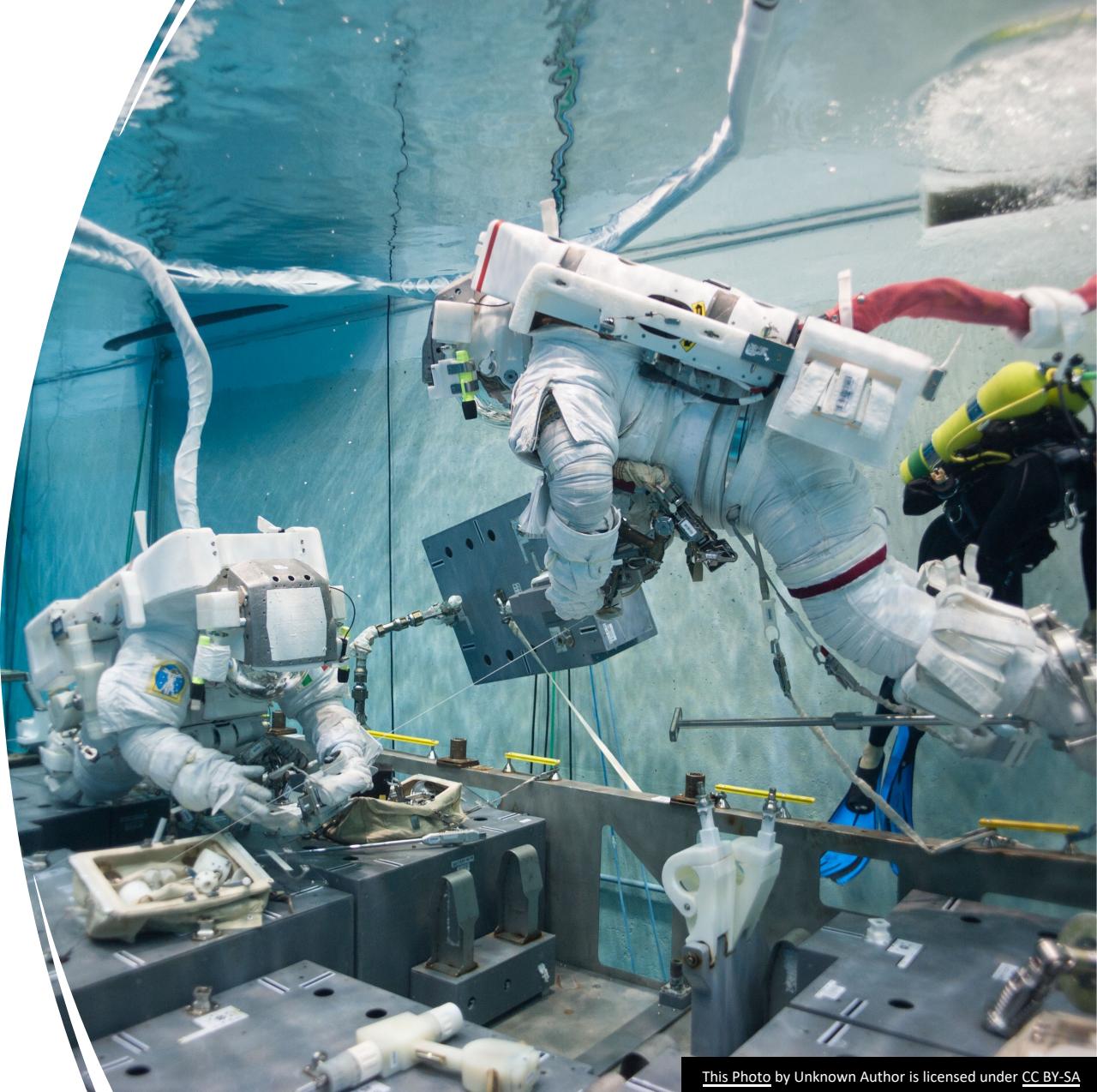


“Shift Left” Challenges



The Scale Challenge

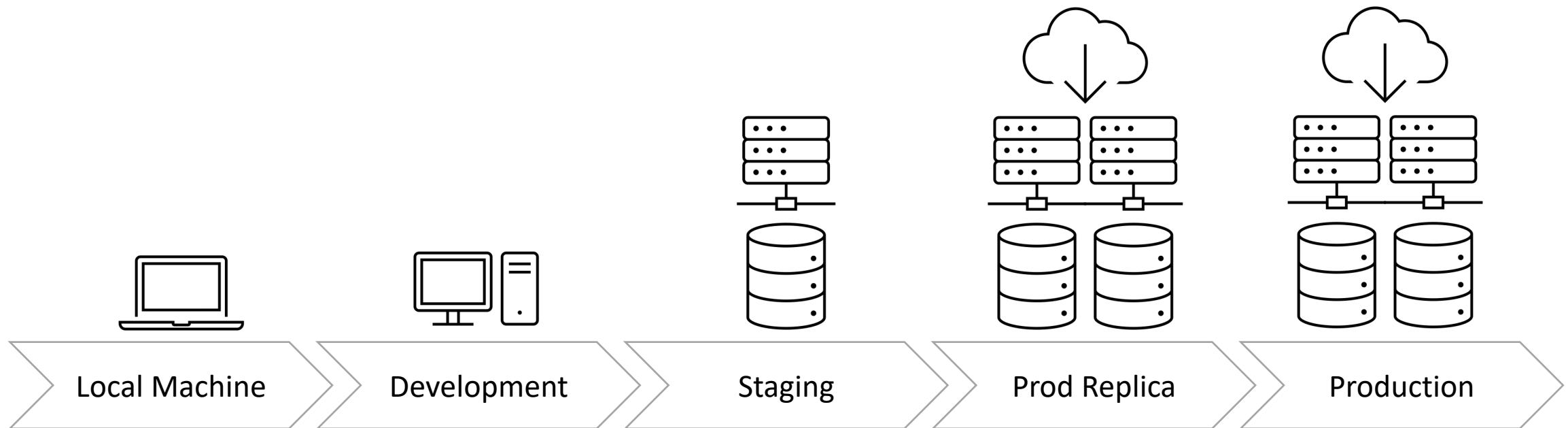
Creating truly representative performance test setups on local machines



This Photo by Unknown Author is licensed under CC BY-SA



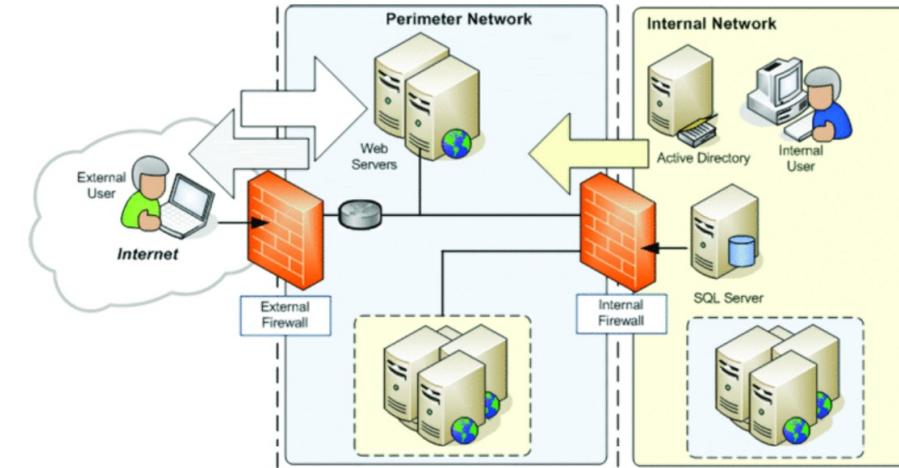
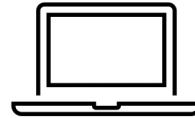
Local Machines are not representative of Production Architecture



Local Machines are not representative of Network Topology

Lower environments do not have production like sophisticated network topology which involves

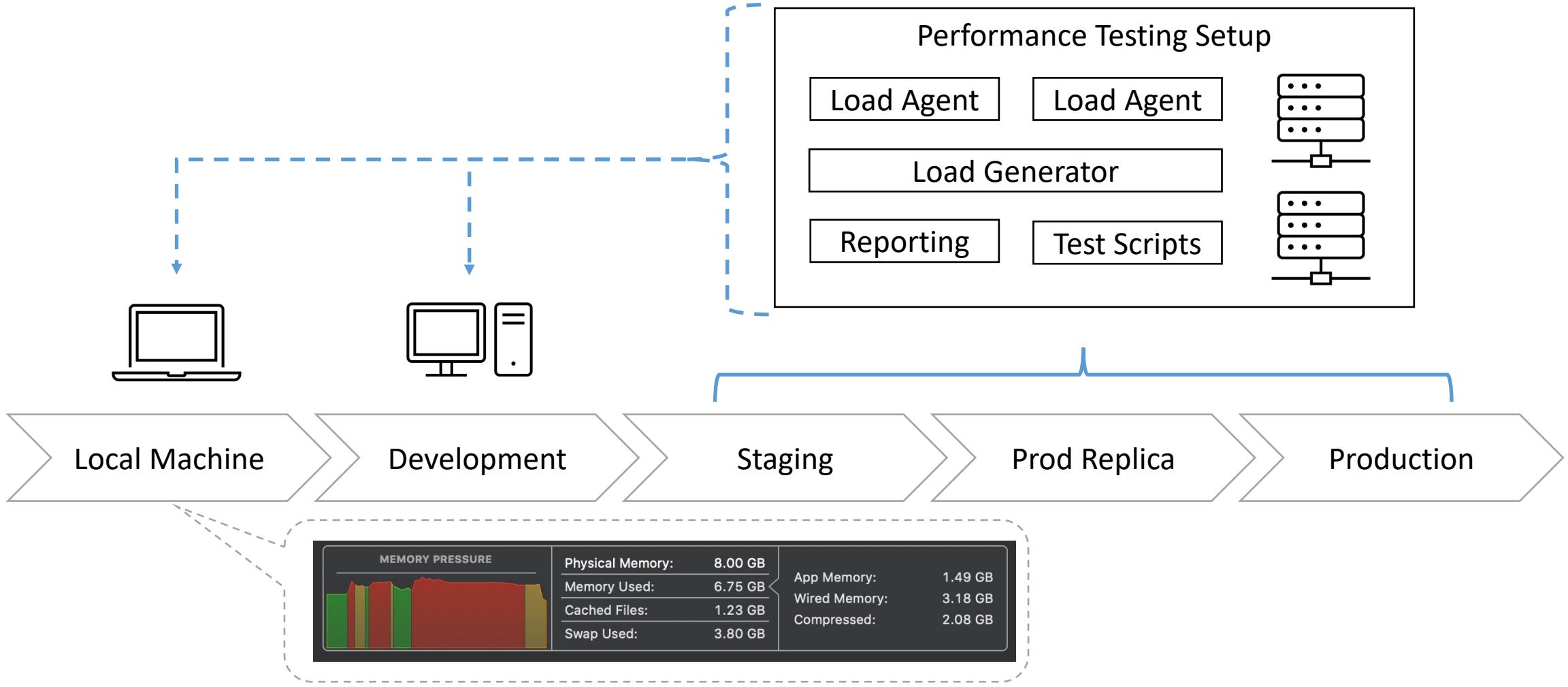
- Firewalls
- Proxies
- Latency levels etc.



[This Photo](#) by Unknown Author is licensed under [CC BY-SA](#)

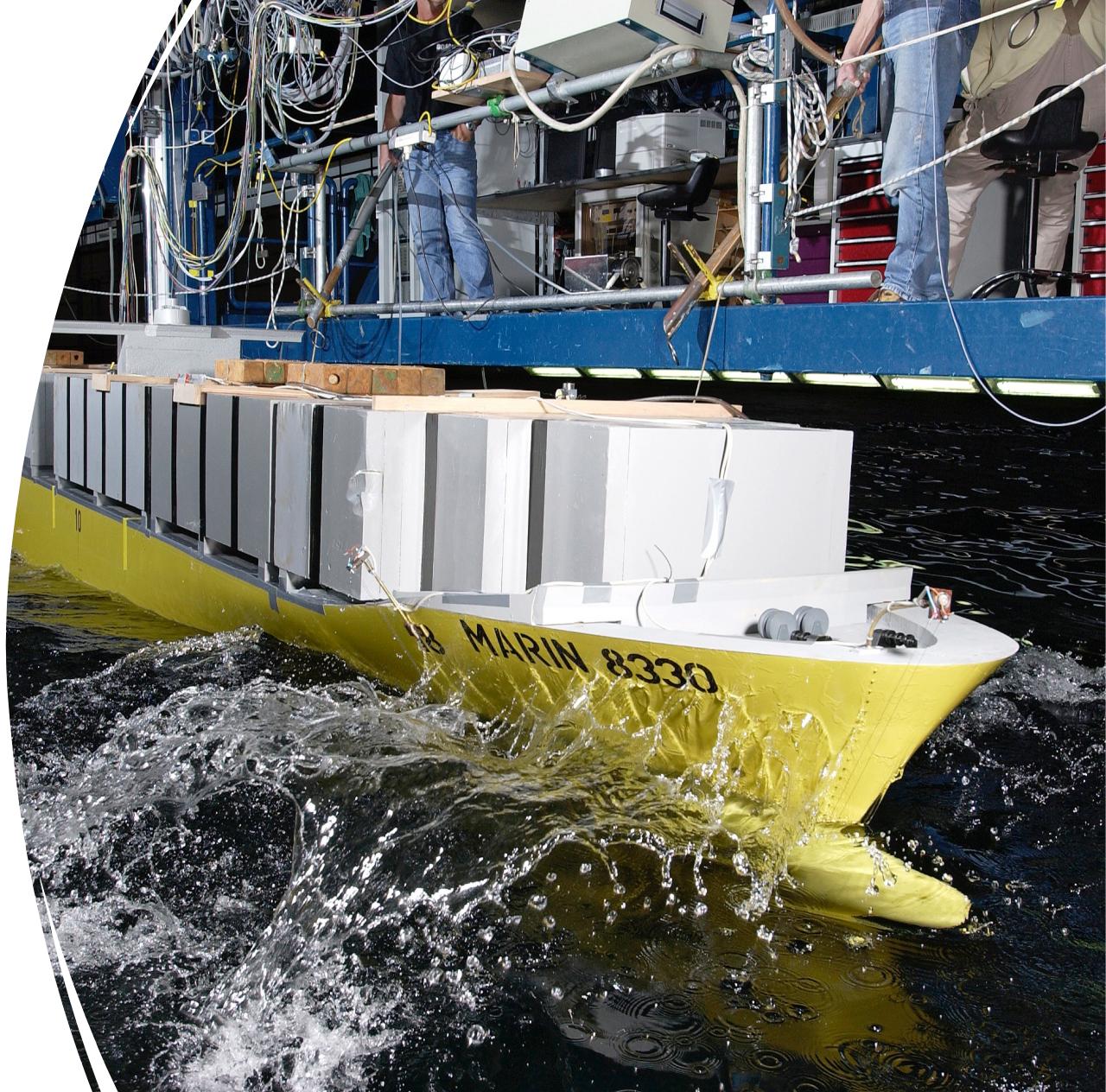


Developer Machines are not capable of Generating Adequate Load

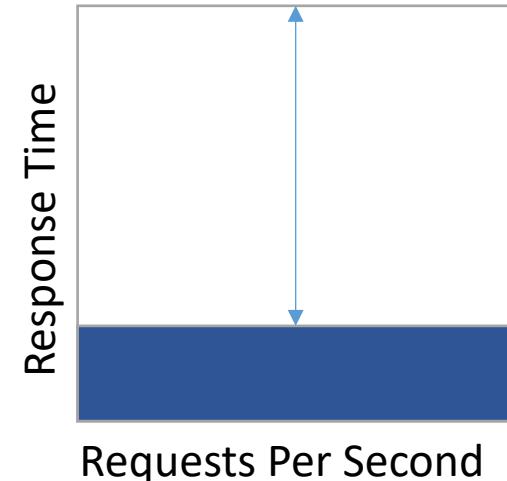
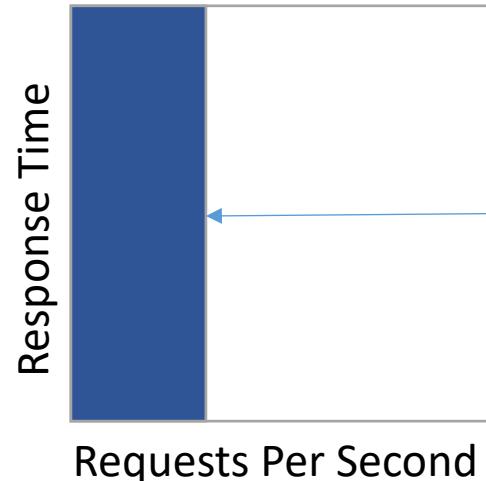
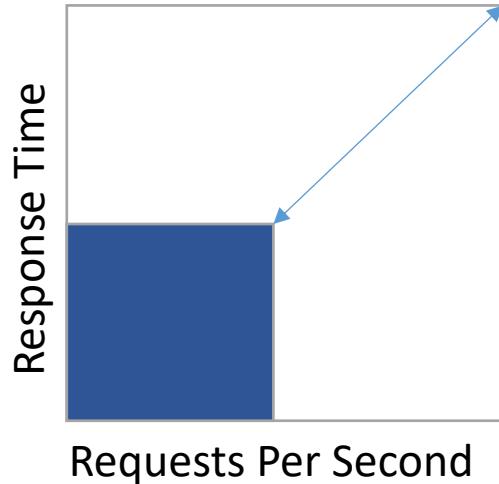


Shift Left = Scale Down

It is not possible to conduct sea trials on a full-scale ship with every design change.



Basis to Scale Down



How do we scale-down load **accurately** to validate performance KPIs?

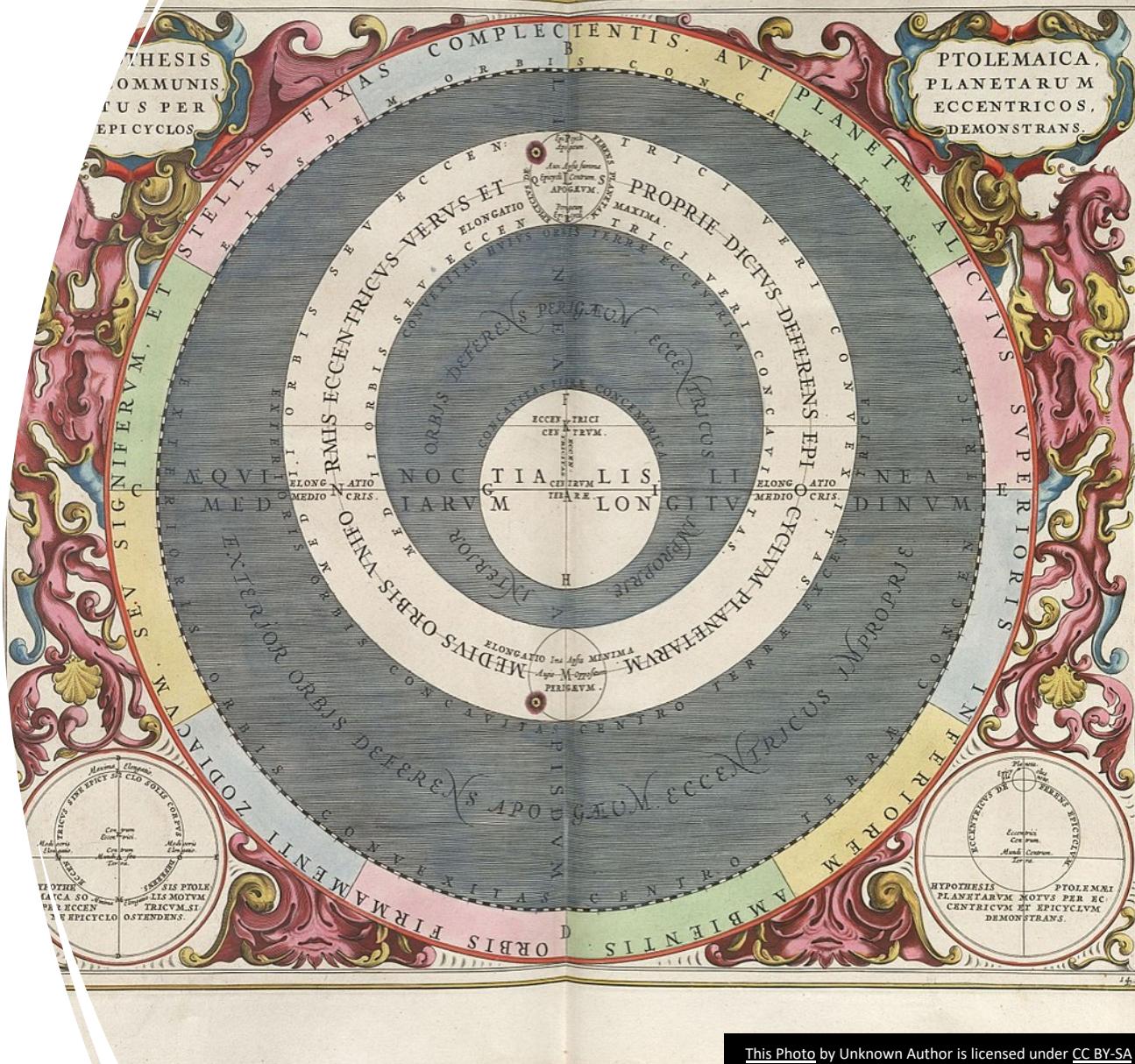
We cannot.

However, we can **scale down the trend** and
invalidate hypotheses



Hypothesis Invalidation

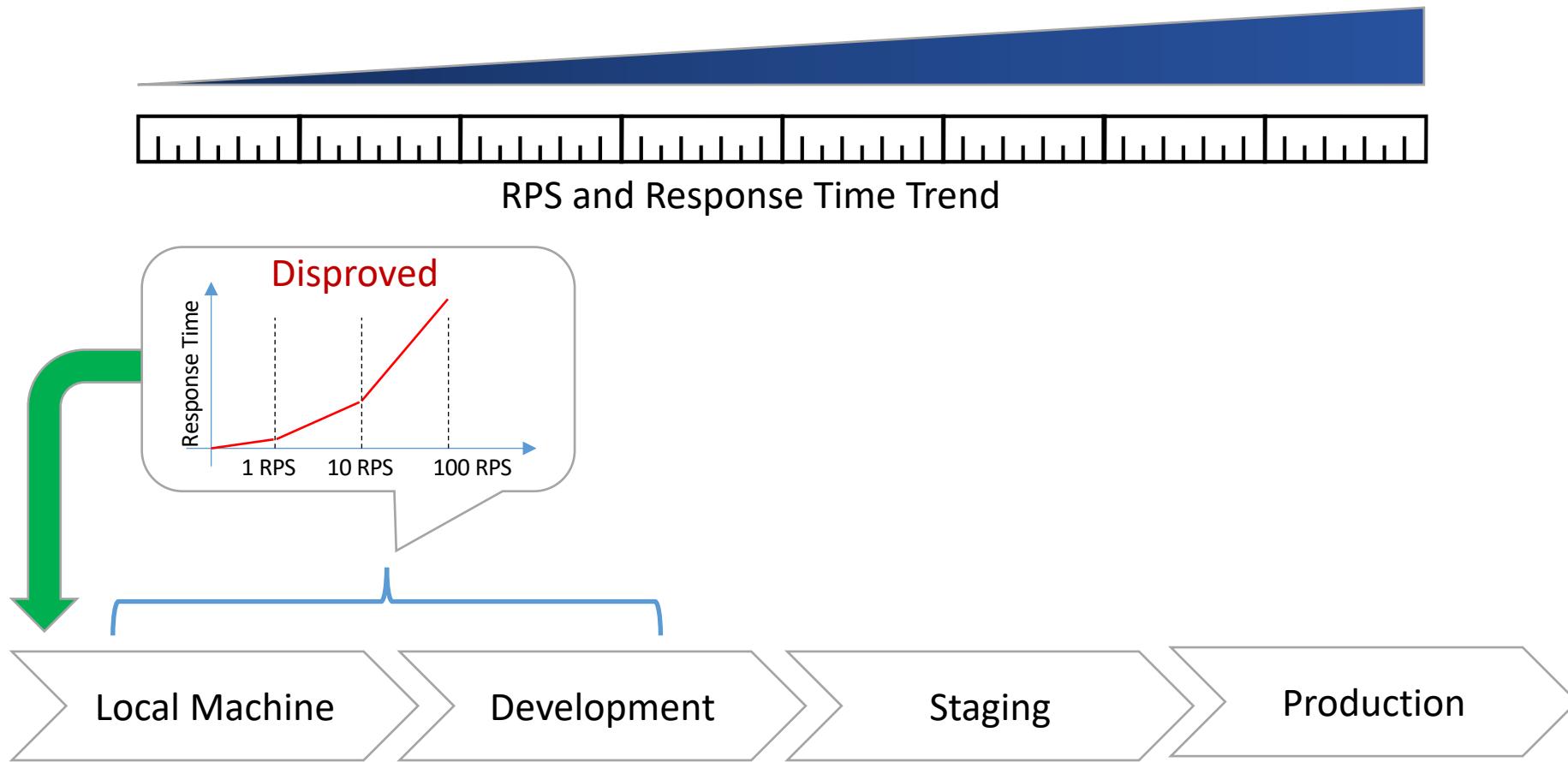
*a hypothesis is used to formulate provisional ideas... the hypothesis is proven to be "true" or "false" through a **verifiability**- or **falsifiability**- oriented experiment – Wikipedia*



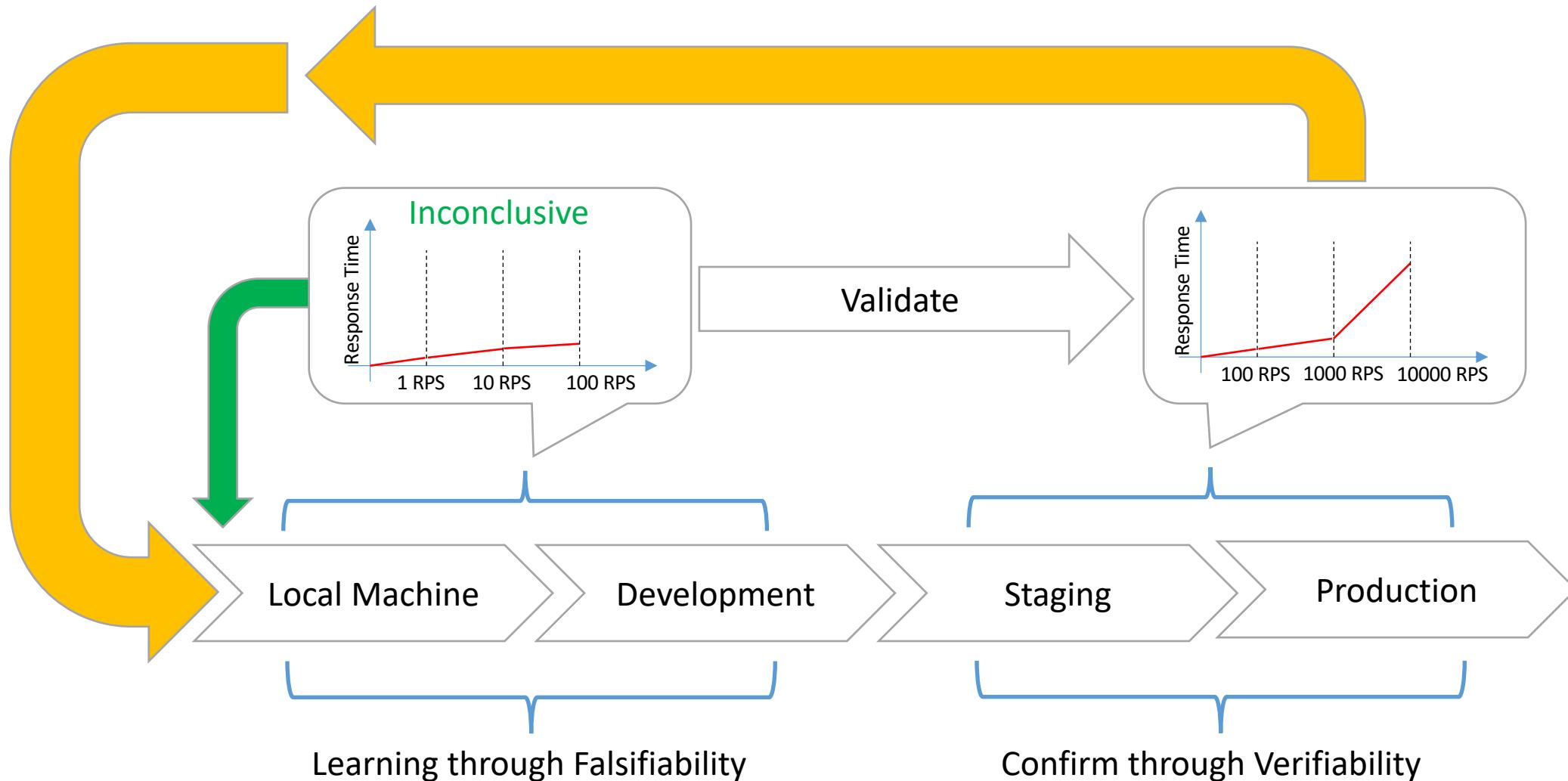
This Photo by Unknown Author is licensed under CC BY-SA



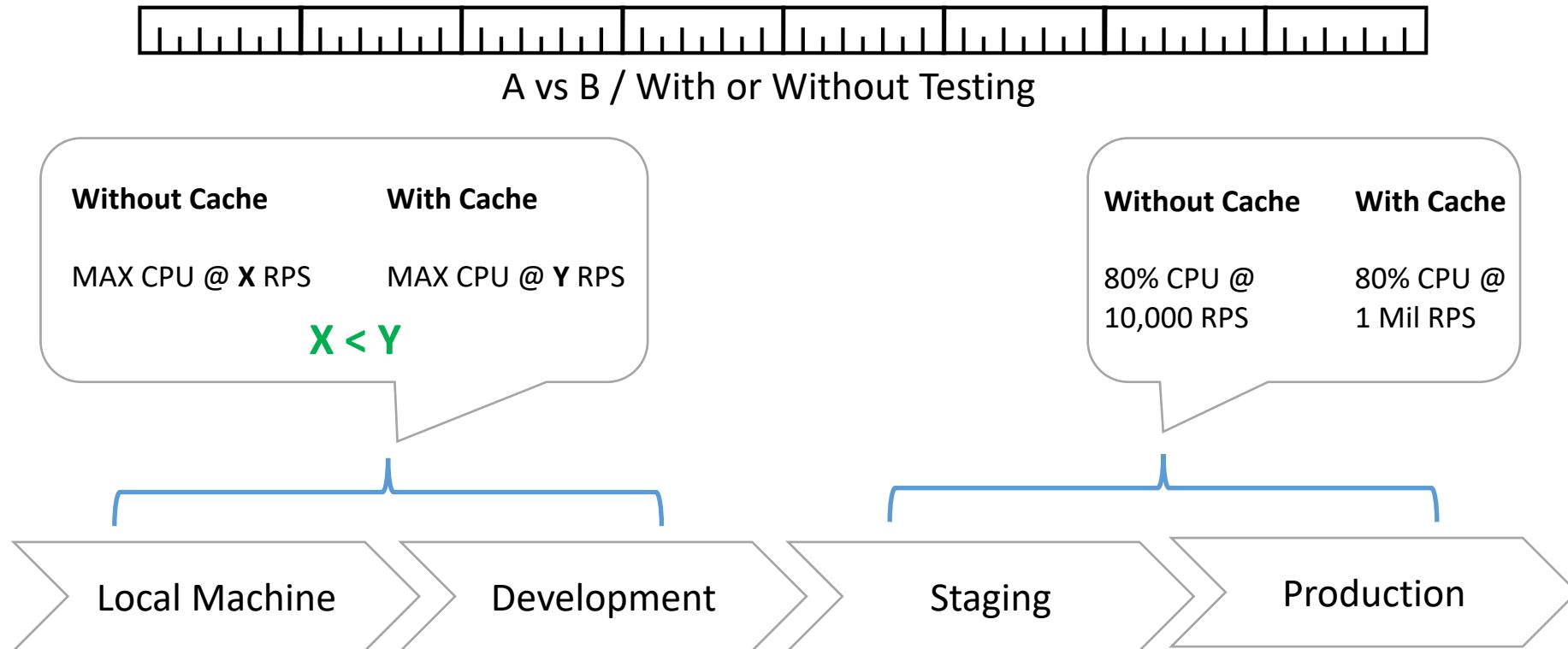
Invalidating the Scaled Down Problem



Invalidating the Scaled Down Problem



Invalidating the Scaled Down Problem



The Capacity Challenge

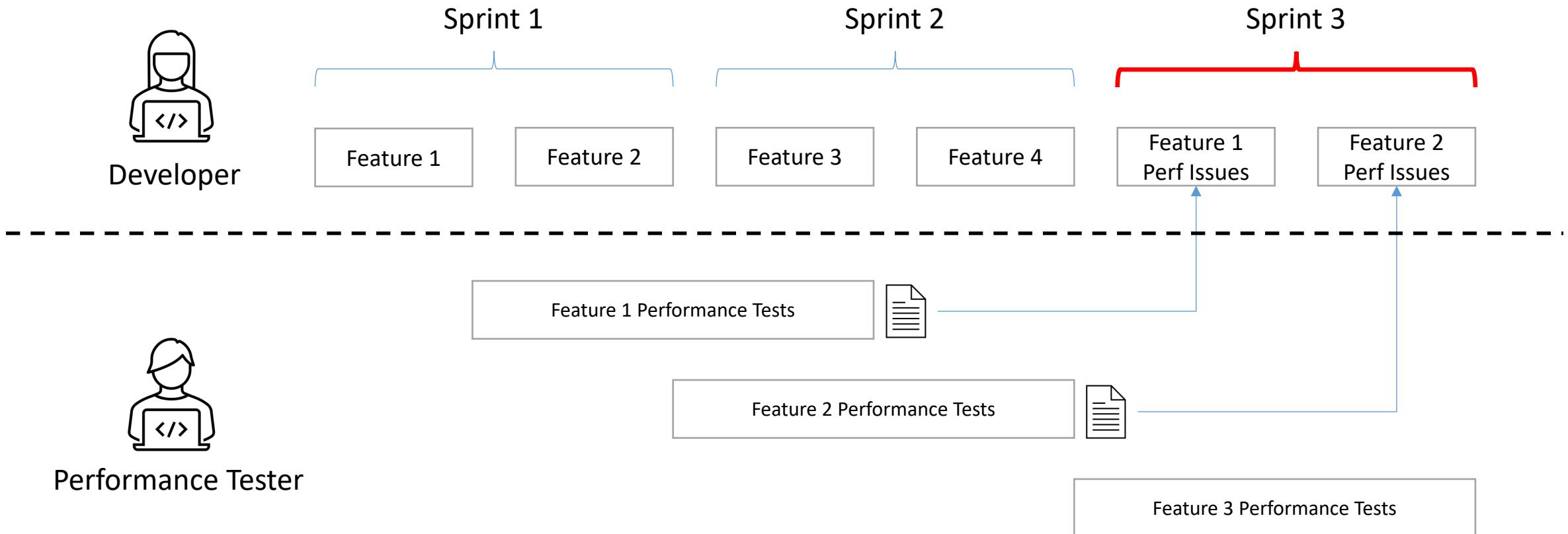
The problem with running Performance Tests within your Sprint Cycle



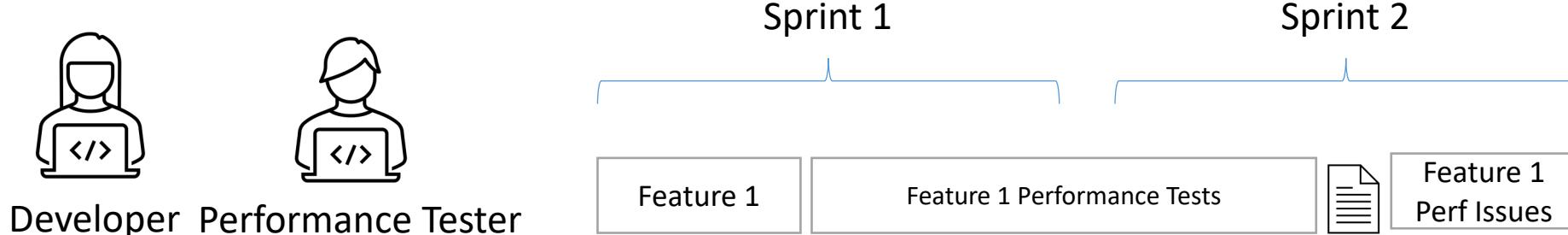
This Photo by Unknown Author is licensed under CC BY-SA-NC



Perf Tests and Sprint Cadence



Perf Tests and Sprint Cadence

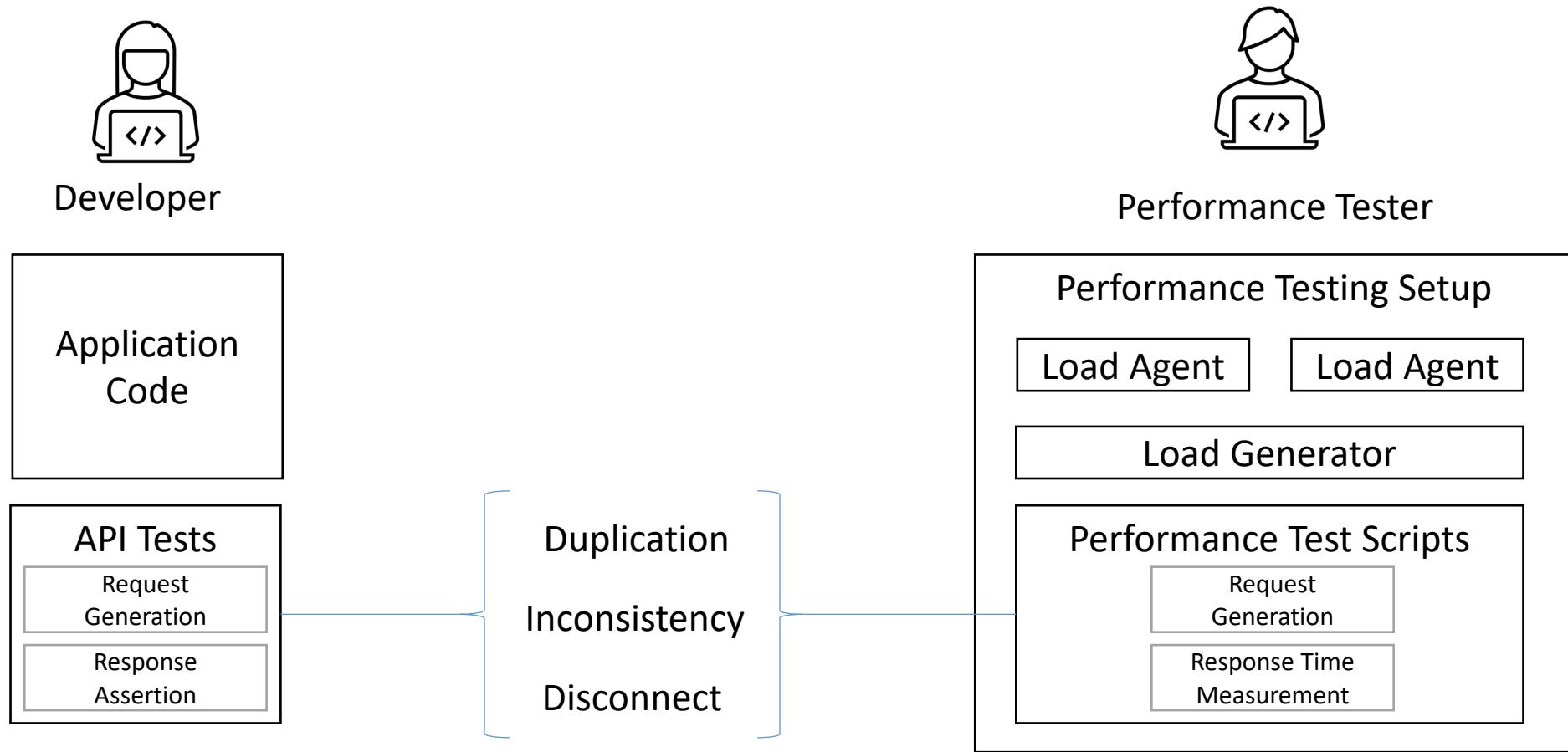


Reduce this Feedback Cycle

- 1. Reduce Effort**
- 2. Reduce Complexity**
- 3. Reduce Repetition**
- 4. Automate**
- 5. Automate**
- 6. Automate...**



Reduce Repetition: Performance Test Scripts



Shift Left = Repurpose

Leverage API Tests / Specs as Perf Test Scripts

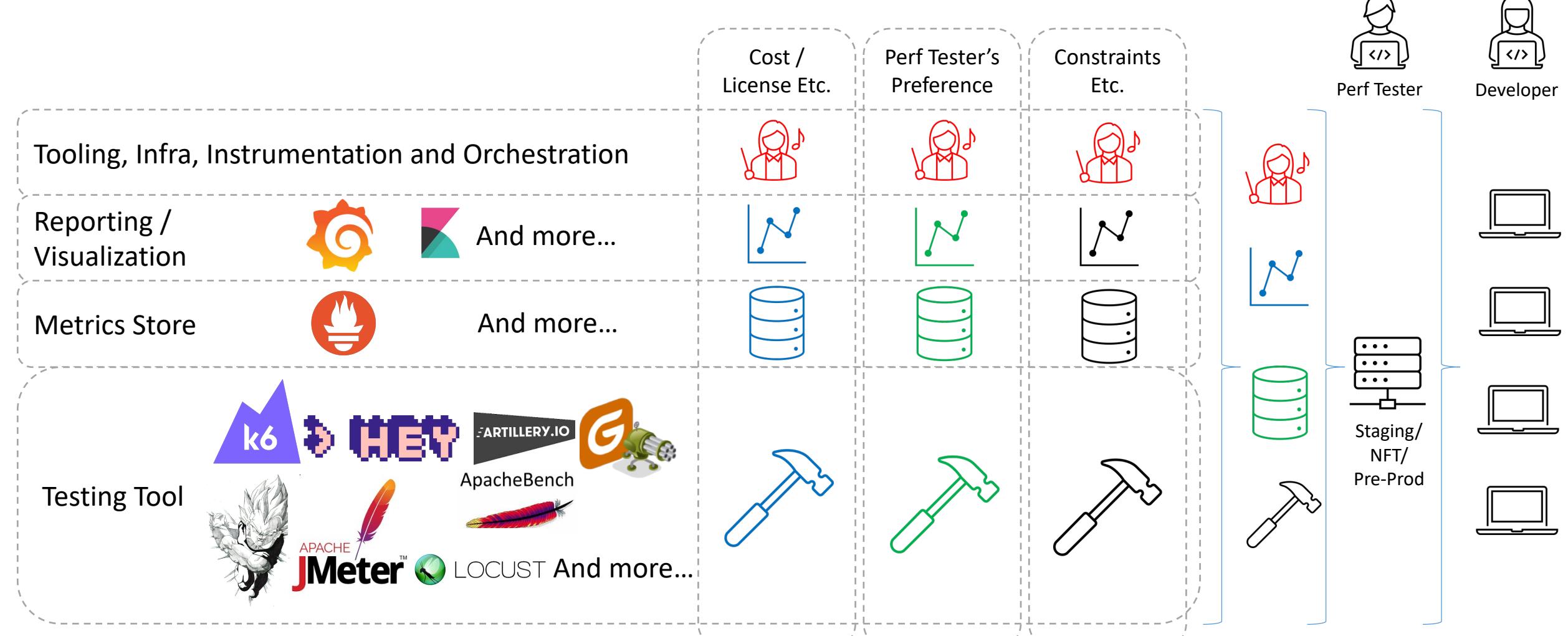
- **Reduced Maintenance** – No need to update two separate scripts
- **Consistency** – While Perf tests may not require result assertion, they will still be consistent with API specs
- **Co-ordinated Effort** – Performance Testers and Developers work more closely and can help each other



Image Attribution: [Vas Foto](#), CC BY-SA 4.0, via Wikimedia Commons



Reduce Complexity: Perf Testing Stack



Third party marks and brands are the property of their respective holders



Shift Left = Containerize

- Containerize the performance test setup so that there are no manual steps to bring it up in a local machine or in a higher env
- Develop your Perf Test Setup just like code on Dev Machine and promote it to higher environments



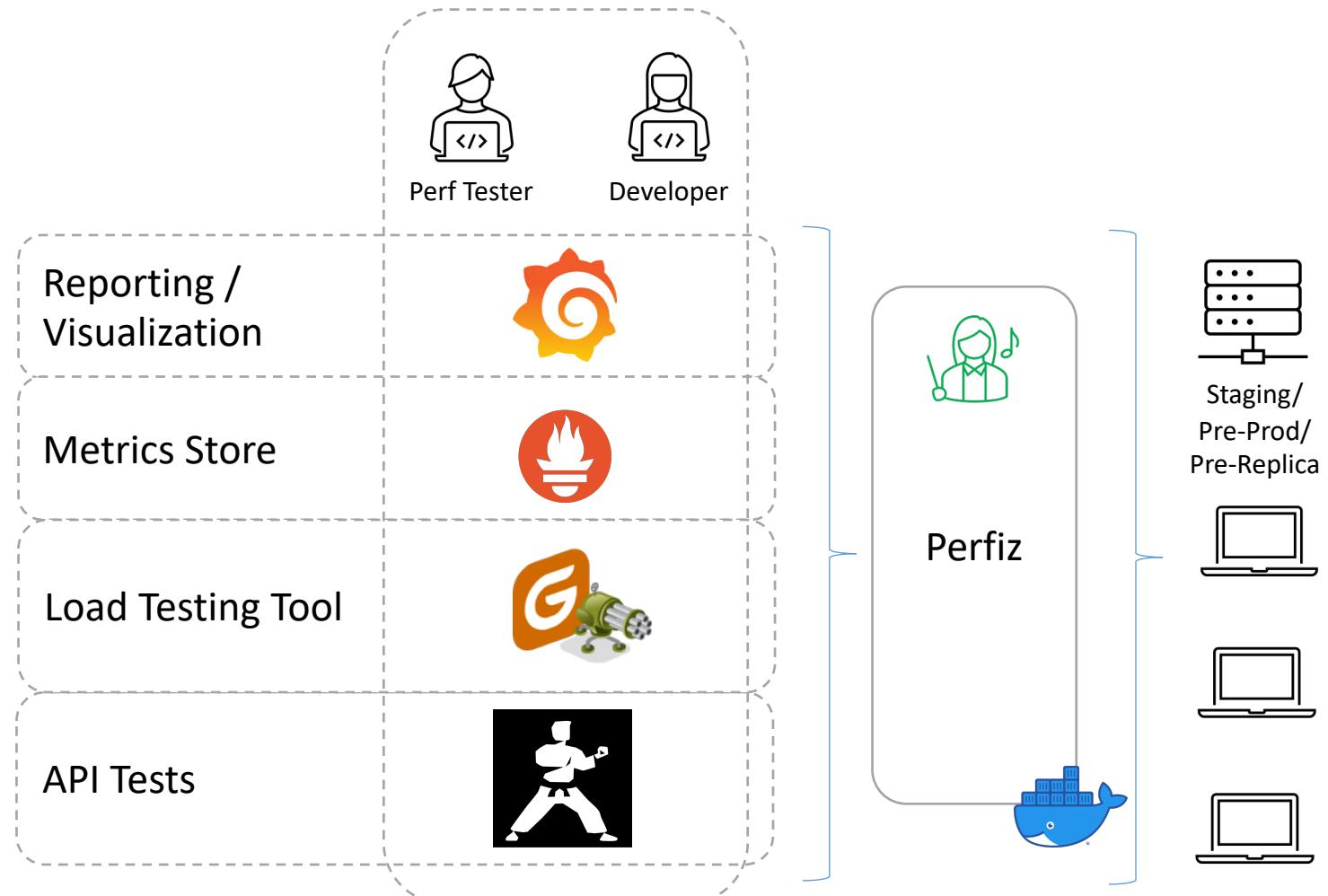
Live Demo

Perfiz - An Open-Source
Containerized Perf Test Setup

<https://perfiz.com/>



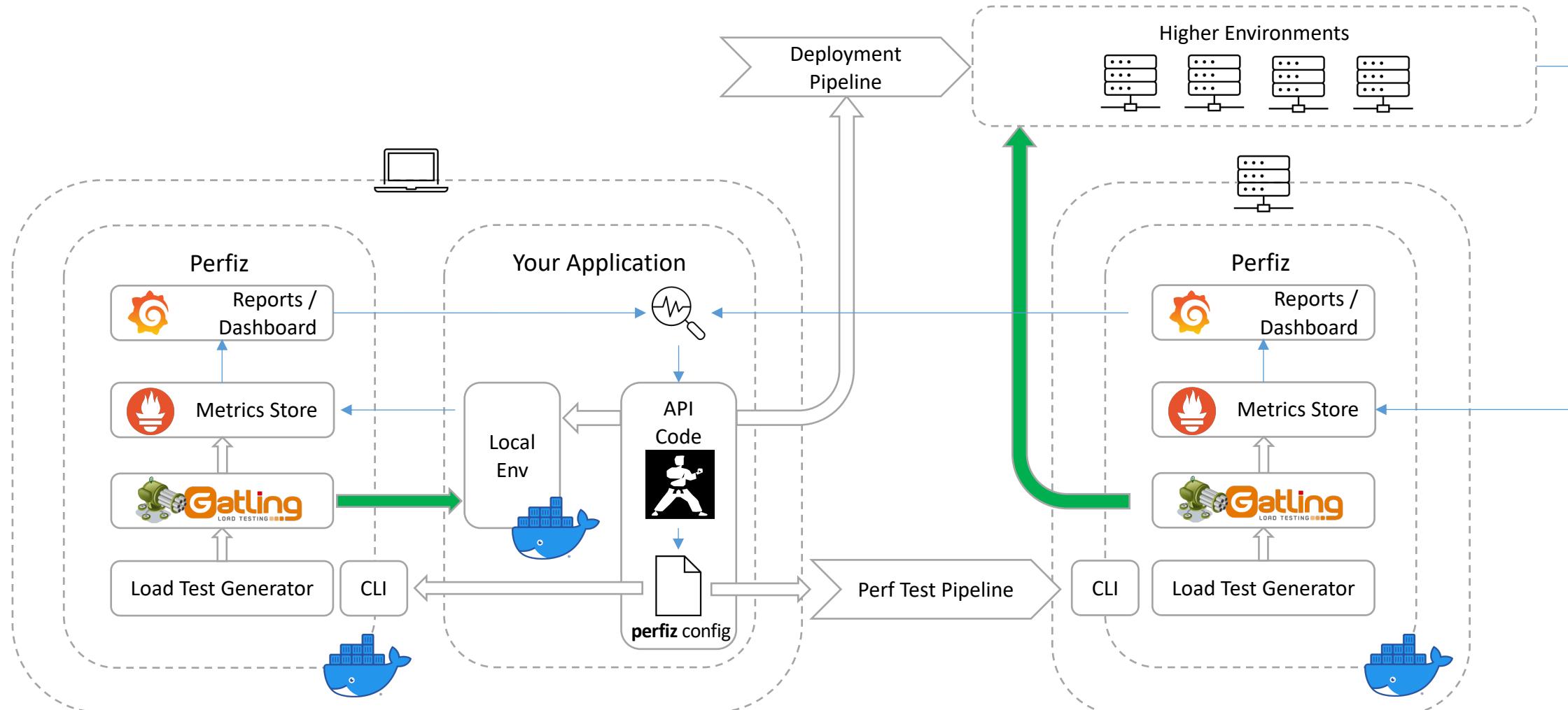
The Perfiz Stack



Third party marks and brands are the property of their respective holders



Perfiz – How it works?



Third party marks and brands are the property of their respective holders

@harikrishnan83 | #testingguy





Shift Left - Mindset

Performance Testing vs Performance Engineering



Shift Left = Performance Testing as a Learning Exercise

- “**Testing**” – This word may be leading us to believe Perf Testing is a Verification Activity.
- Instead, a large part of Perf Testing should be a series of **Spikes** which help us learn and think scientifically about our solutions.
- Avoid “**Guesswork**” in Architecting Applications by Validating your Hypothesis (OR invalidating them early)



Scientific Method

Avoiding Guesswork

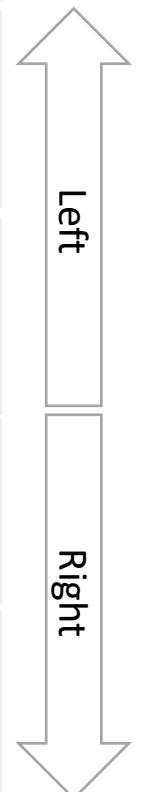


This Photo by Unknown Author is licensed under CC BY-SA



Continuous Evolution Template

Problem Statement	Baseline KPI	Target KPI	Hypothesis	Experiments		Validated Learning
				Falsifiability	Verifiability	
Improve Max Throughput	10 K RPS @ 80% CPU	100 K RPS @ 80% CPU	Adding an In-Memory Cache will reduce Repeat Computations	On Local Machine , there must be a change in CPU usage after adding Cache	NA	Miss Rate is 100 %
			TTL on Cache Keys is too small and thereby keys are expiring before they are accessed a second time	Same as Above	On Staging , should be able to achieve 10x improvement in RPS	Staging achieved only 4x Improvement Miss Rate is 40% Eviction Rate is High
			Eviction Rate is High because we are running out of space. Increase Cache Size to X.	On Local Machine , decreasing Cache Size should cause Eviction Rate (To establish cause)	Same as Above	Application is stable at 80% CPU and X GB Memory
			Based on Previous Learnings, below configuration should achieve Target KPI <ul style="list-style-type: none"> TTL: X Cache Size: Y 	On Prod-Replica , under baseline load <ul style="list-style-type: none"> CPU Should Drop Cache Hit Rate 100% 	On Prod-Replica , should be able to achieve Target KPI	





Questions?

Thanks!

HARI KRISHNAN

harikrishnan83@gmail.com
@harikrishnan83

August 2-7, 2021
testinguy.org | @testinguy | #testinguy