



# Anatomy of A Continuous Delivery Pipeline

Kamalika Majumder

Consultant & Director

Cloudkata® By Staxa LLP

# What they need

Shorter Release Cycles

Continuous Integration

Quality Assurance

Security

# What they want

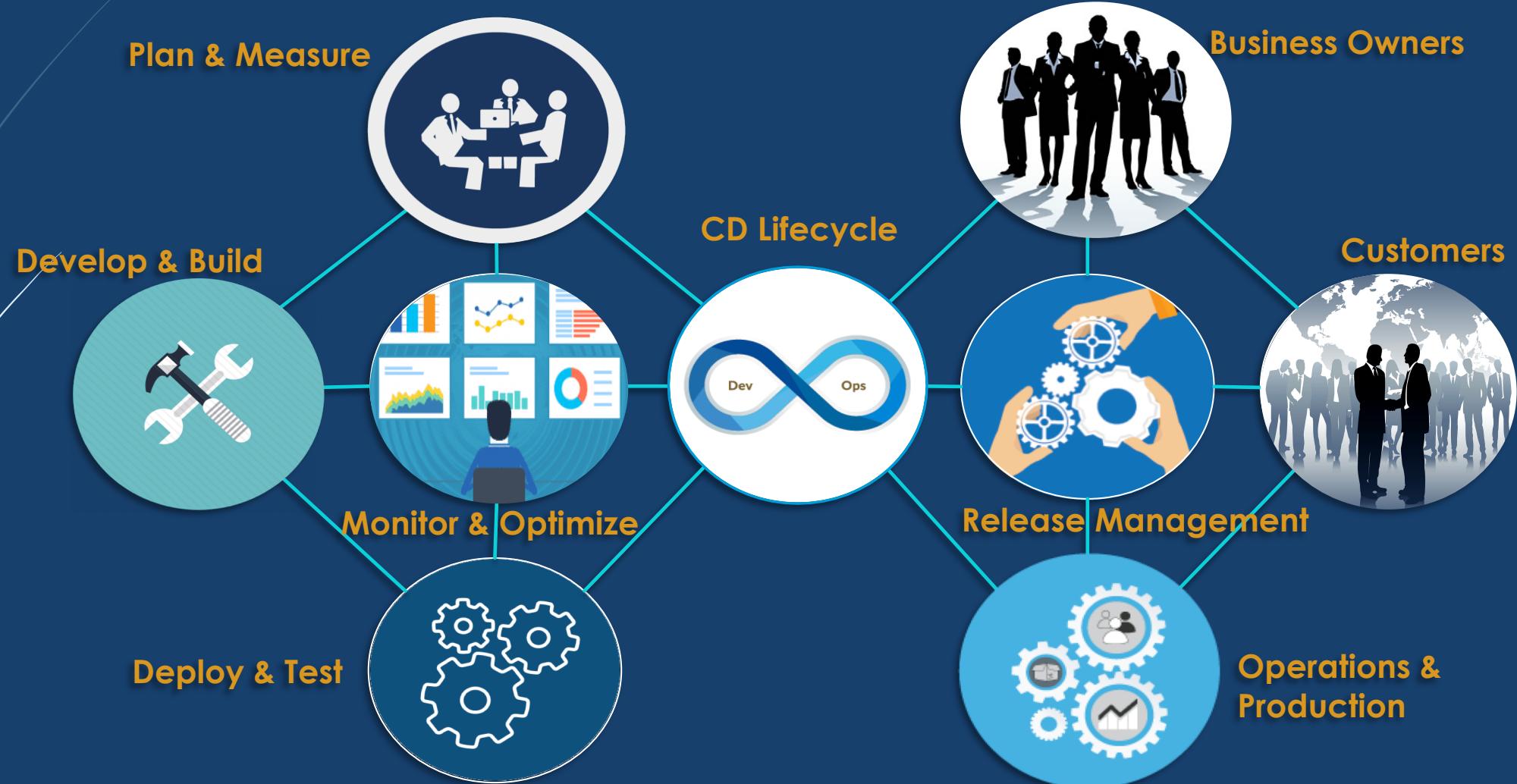
Fast Sellers

Best Sellers

# What I hear from most of my clients?

- ▶ We have all the DevOps tools but still we are not delivering fast enough ?
- ▶ Too much lead time to go from UAT to production.
- ▶ We have been doing CI/CD for years , why cant we release when we want to ?
- ▶ Despite automated build and deploy why does it takes so long to release ?
- ▶ Why do we still see production issues ?
- ▶ How can we reduce time to market ?

One stop station for application lifecycle management  
and continuous delivery of quality products



# Back to Basics

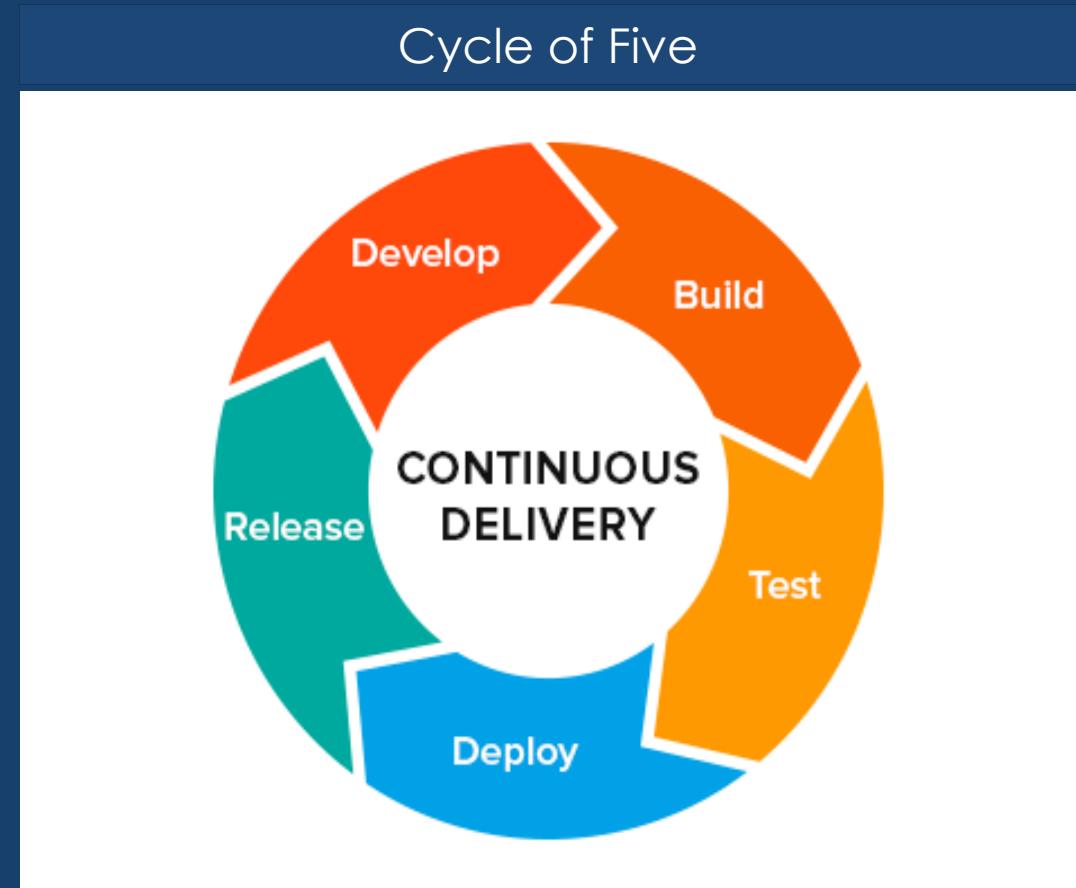
“Continuous Delivery is a **software development discipline** where you build software in such a way that the software can be released to production at any time.”

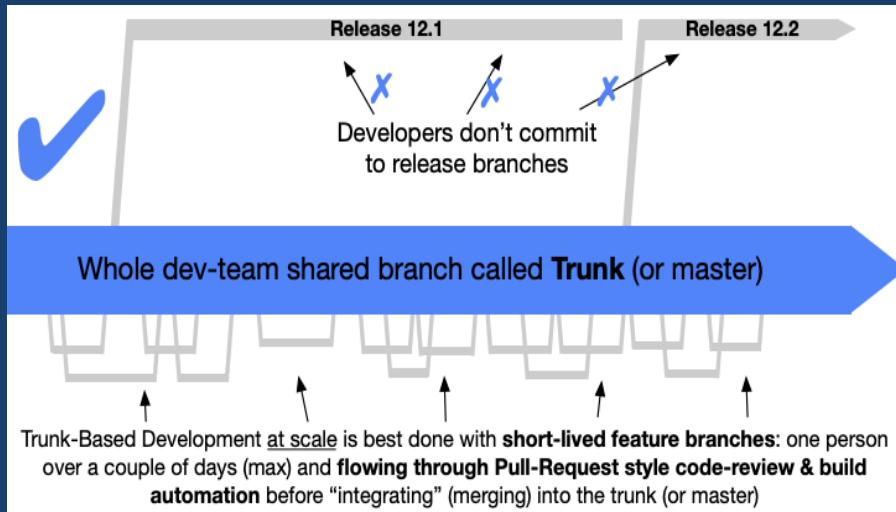
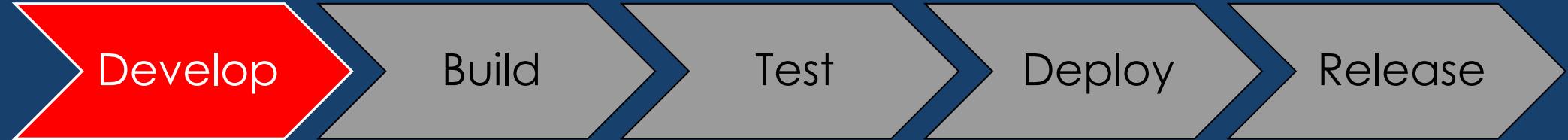
- Martin Fowler

Ability to get changes of all types

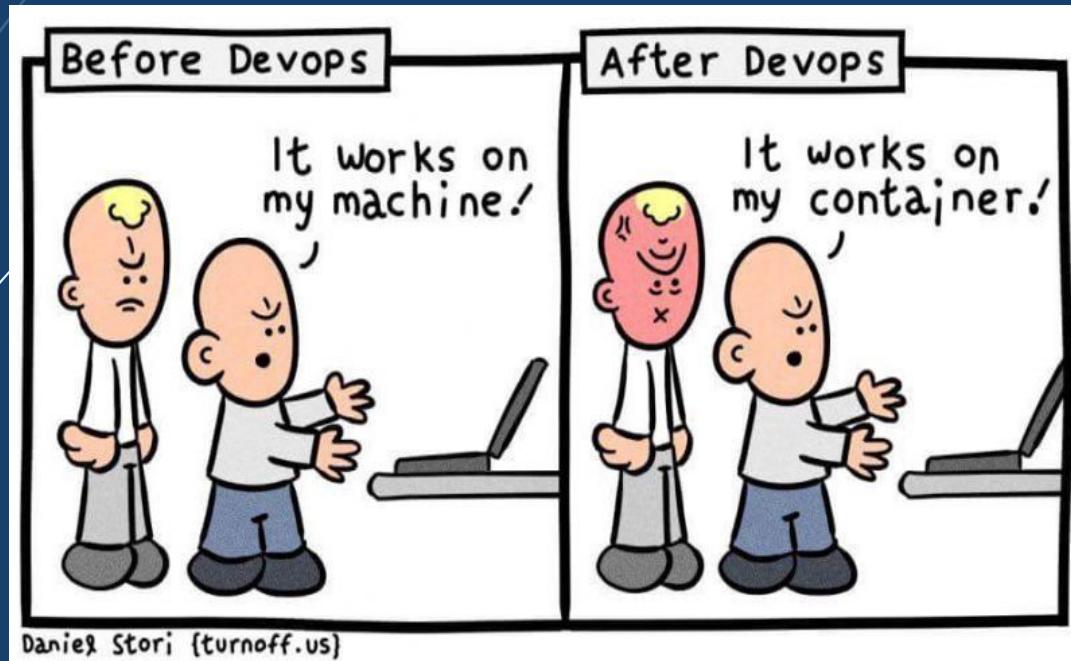
- Into the hands of users, safely and quickly in a sustainable way &
- Ensuring our code is always in a deployable state

ref: <https://continuousdelivery.com/>



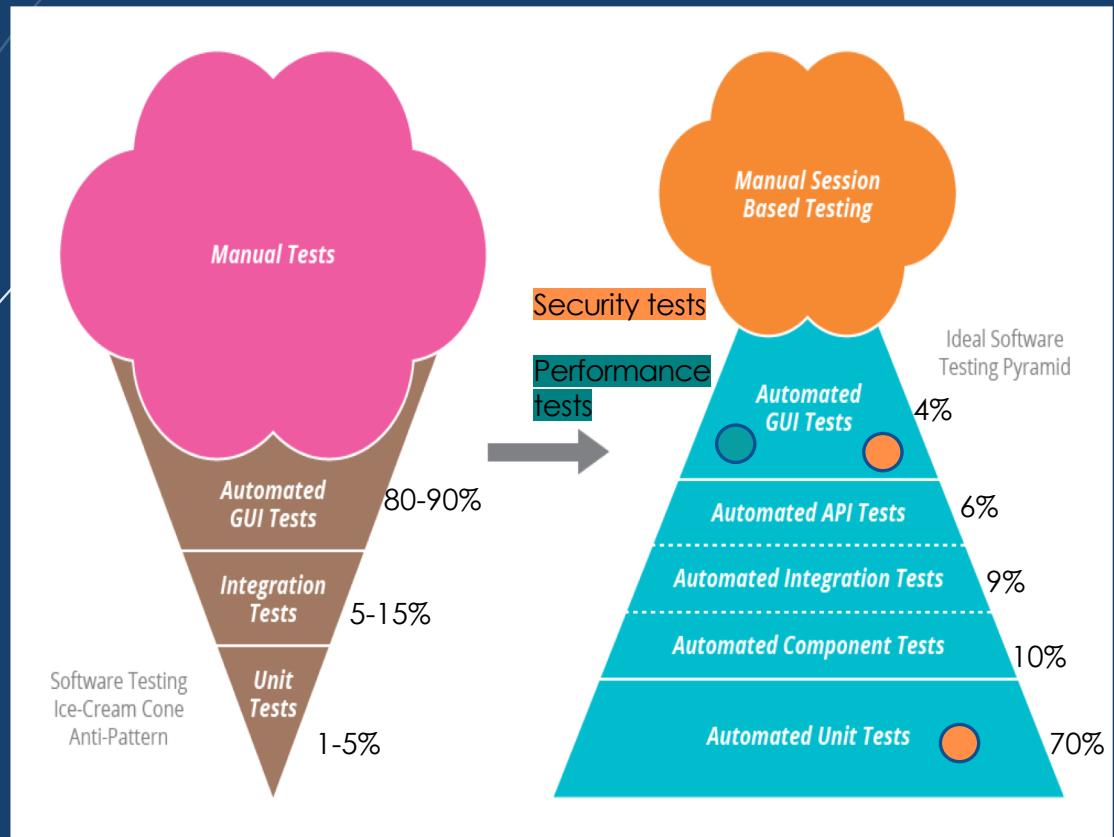
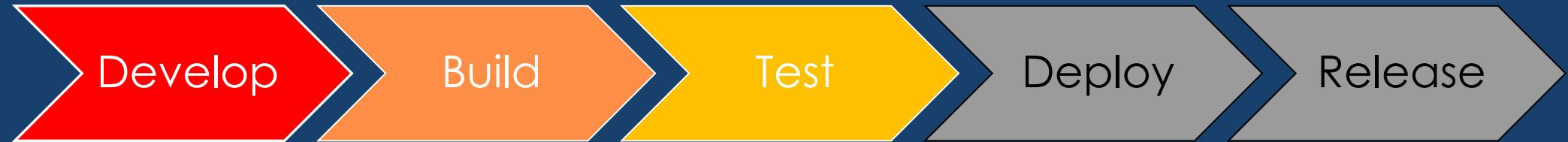


- Everything is Version Controlled
- Trunk Based Development
- Every Developer must commit to trunk/master at least once on a daily basis
- MRs/PRs are peer review methods, not approval mechanisms
- Hide unfinished code with feature toggles
- Trunk is always in a releasable stage
- Never check in on a broken trunk.

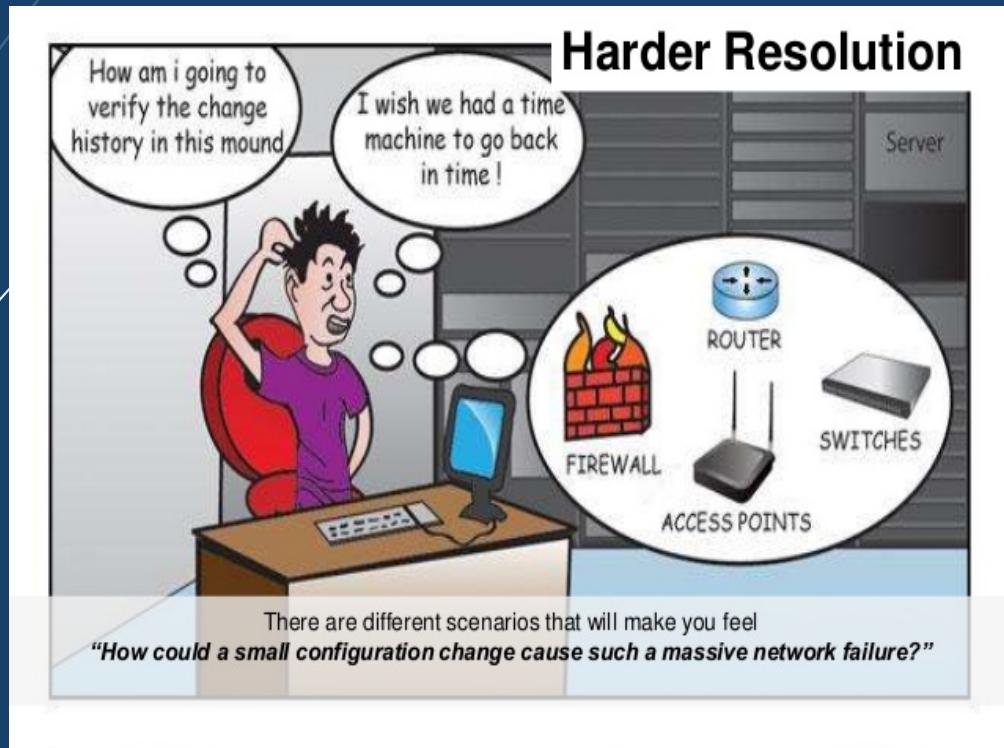
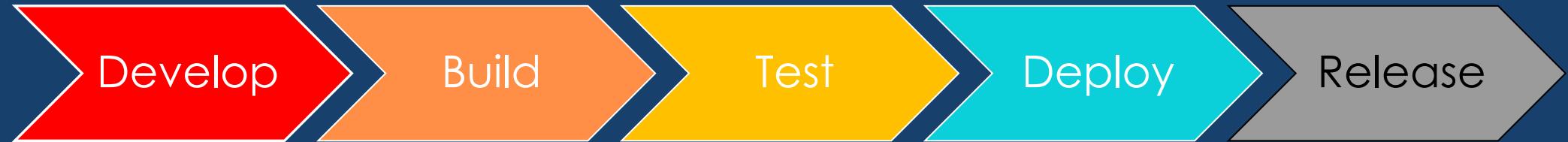


Ref: <https://medium.com/@dwdraju/how-it-works-in-my-machine-turns-it-works-in-my-container-1b9a340ca43d>

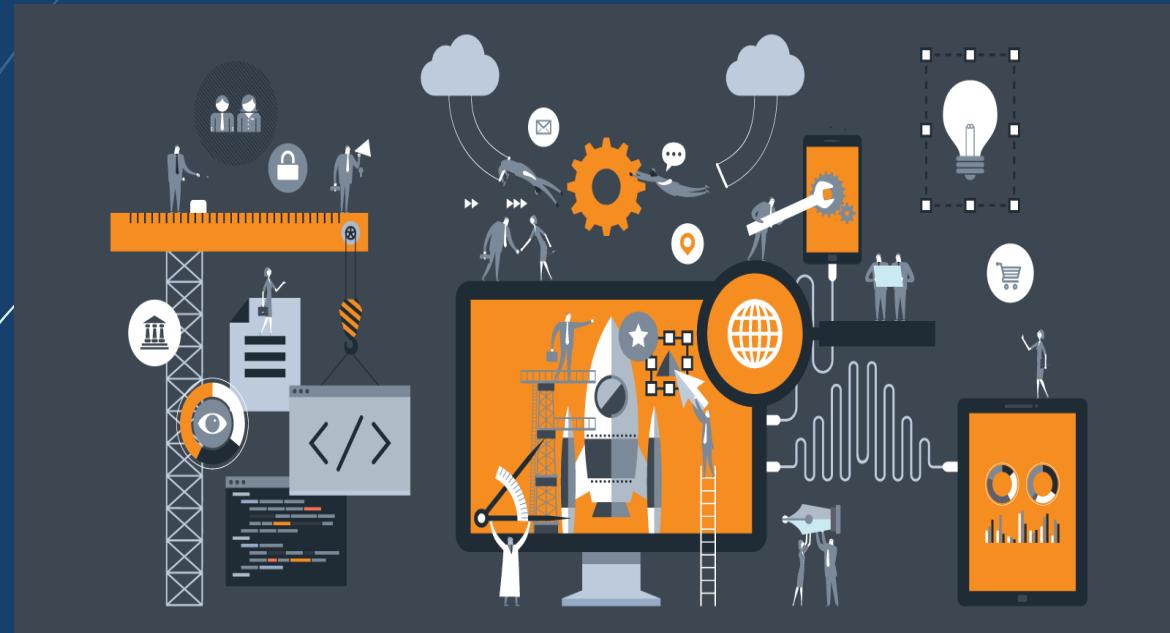
- ▶ Think beyond localhost
- ▶ Build once deploy anywhere
- ▶ Centralized package store
- ▶ Semantic Versioning for all packages
  - Apps, Configs, DB scripts etc.
- ▶ CI Pipeline for build, package, publish, version upgrades and build promotion.



- ▶ Test Driven Development
- ▶ When you Fail Fast and fail early you can fix early.
- ▶ Cover maximum on unit level
- ▶ Test Automation at every level
- ▶ Continuous Integration and Reporting for fast ,useful feedback
- ▶ Parallel test pipelines.

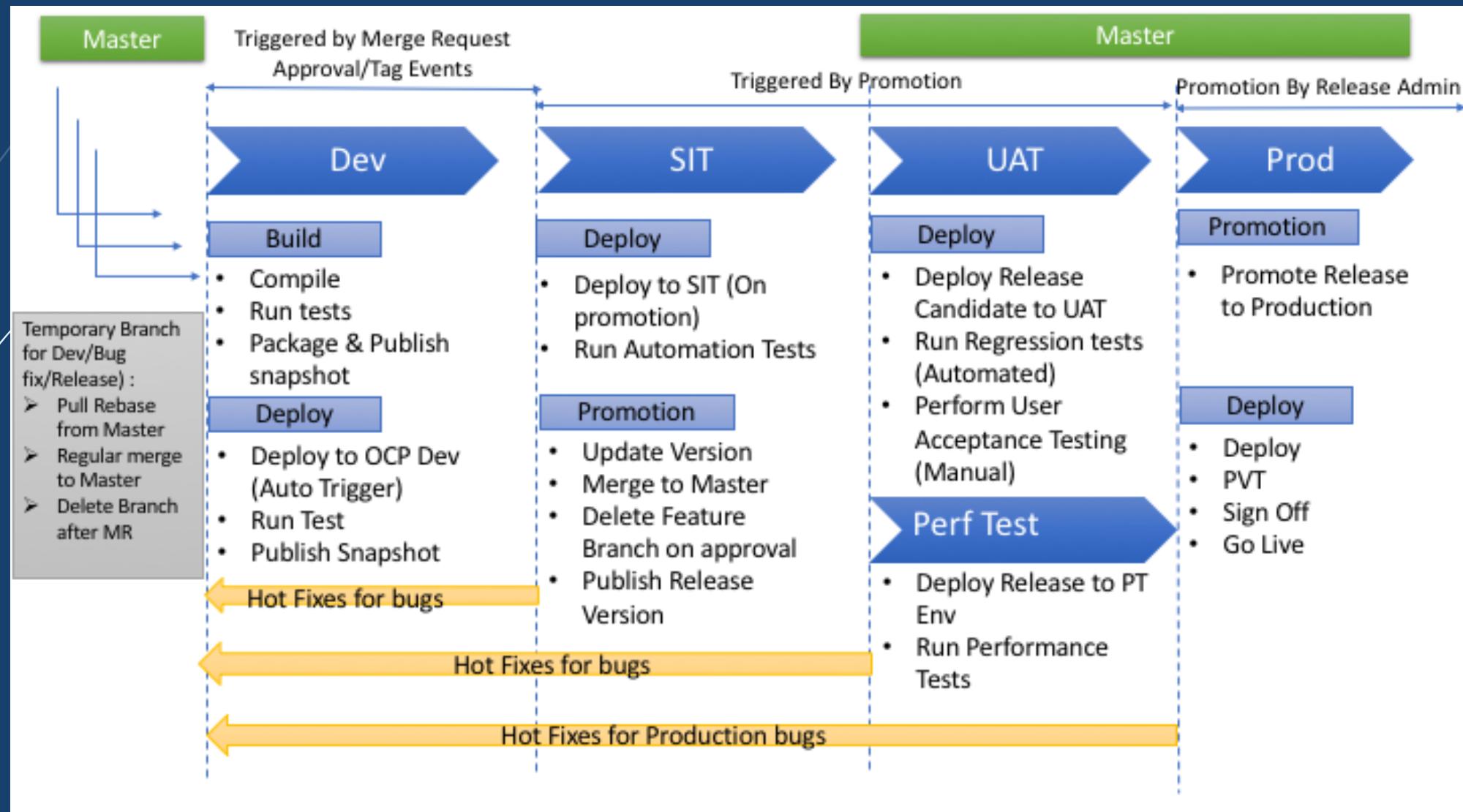


- ▶ Version Controlled Infrastructure As Code
- ▶ One Design Multiple Implementations
- ▶ Configuration Management - model server configs, relationships with other servers, packages in code.
- ▶ Testing Infrastructure Code
- ▶ Scalable Environment On Demand
- ▶ Immutable Infrastructure.
- ▶ Zero Downtime Deployments

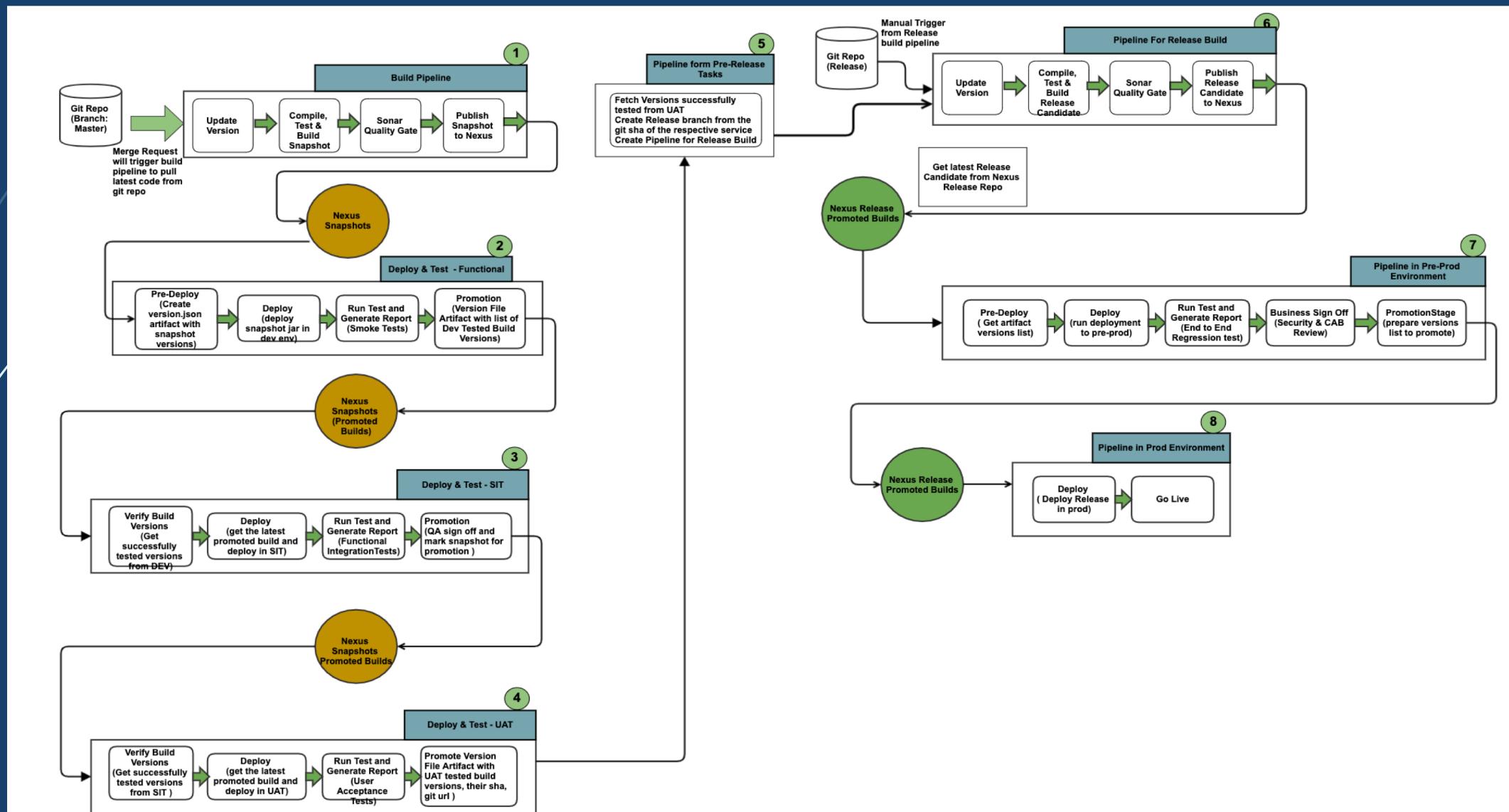


- ▶ Role Based Access Control for prod deployment pipeline
- ▶ End to End traceability - Code to Customers
- ▶ Production Readiness Checkpoint
- ▶ Robust Monitoring, Logging and Alerting
- ▶ Always have a plan B

# Model CD Workflow



# A Sample CD Pipeline



# Closing Note

Write to Me : [kamalika@cloudkata.com](mailto:kamalika@cloudkata.com)

For More visit [cloudkata.com](http://cloudkata.com) and do share your thoughts.

For more updates connect with us at



@Cloudkata®,  
Staxa LLP



@Cloudkata



@cloudkata.staxa.llp