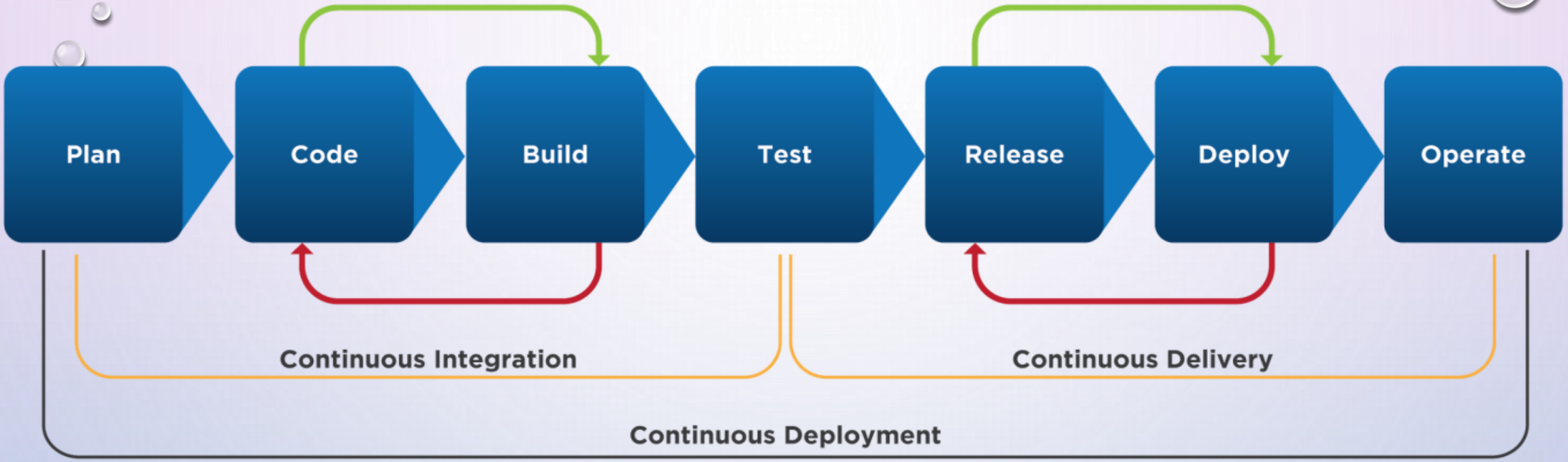
IMPLEMENTING CI/CD FOR THE *UDAPEOPLE*

WHAT IS CI/CD



Planning

- Requirement finalization
- Updates & new changes
- Architecture & design
- Task assignment
- Timeline finalization

Code

- Development
- Configuration finalization
- Check-in source code
- Static-code analysis
- Automated review & peer review

Build

- Compile code
- Unit testing
- Code-metrics
- Build container images or package
- Preparation or update in deployment templated
- Create or update monitor dashboards

Test

- Integration test with other component
- Load & stress test
- UI testing
- Penetration testing
- Requirement testing

Release

- Preparing release notes
- Version tagging
- Code freeze
- Feature freeze

Deploy

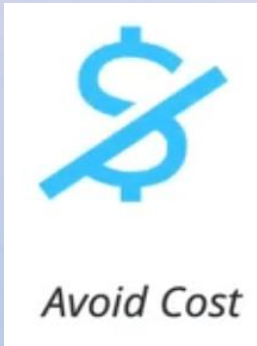
- Updating the infrastructure i.e staging, production
- Verification on deployment i.e smoke tests

Operate

- Monitor designed dashboard
- Alarm triggers
- Automatic critical events handler
- Monitor error logs

BENEFITS OF CI/CD
"COST"

- Less developer time on issues from new developer code
  For Example: *Catch Compile Errors After Merge.*
- Less infrastructure costs from unused resources
  For Example: *Automate Infrastructure Cleanup*.



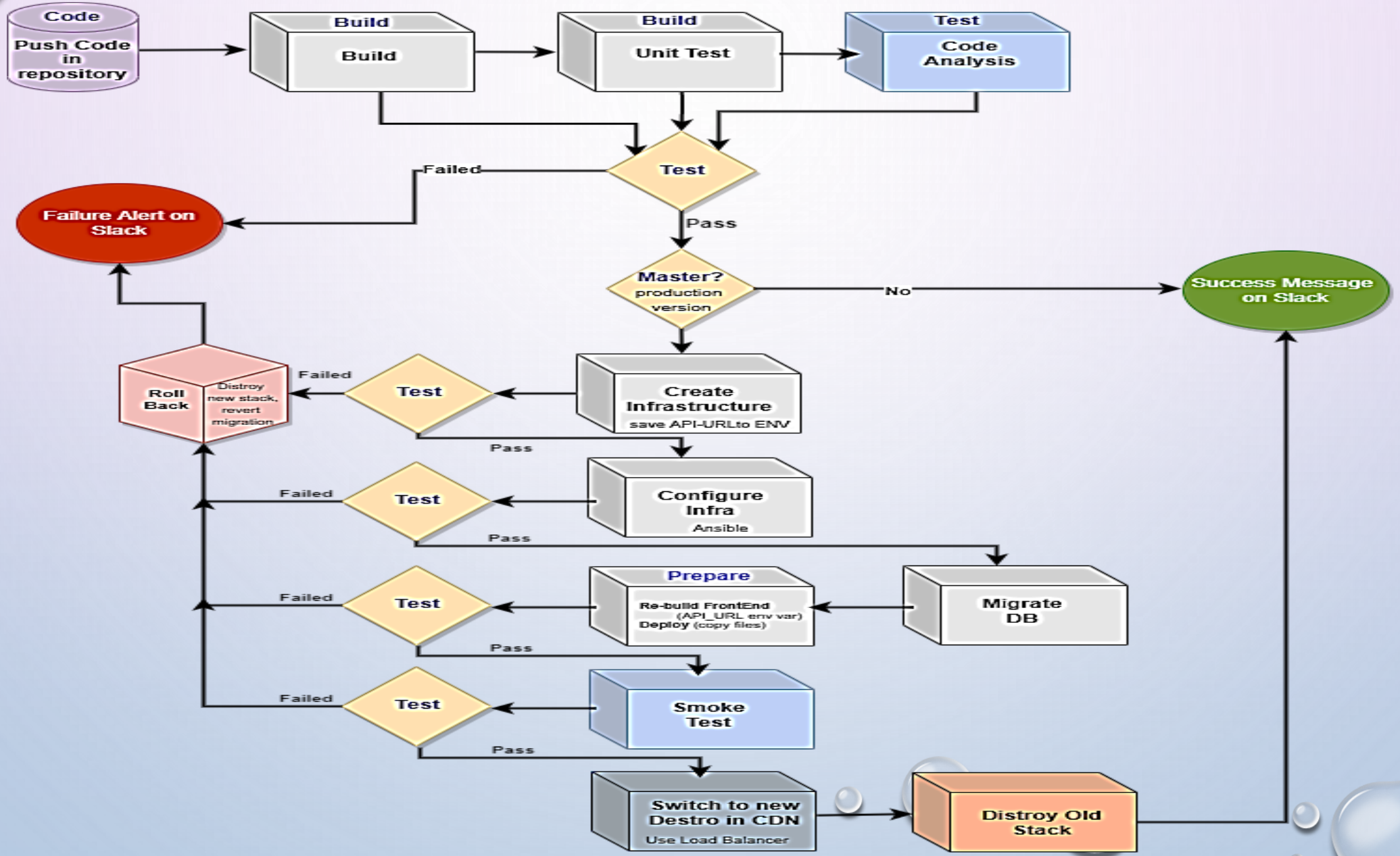
- Less bugs in production and less time in testing
  For Example: *Catch Unit Test Failures.*
- Prevent embarrassing or costly security holes
  For Example: *Detect Security Vulnerabilities.*
- Less human error, Faster deployments
  For Example: *Automate Infrastructure Creation*.

BENEFITS OF CI/CD
"REVENUE"

- New value-generating features released more quickly
  For Example: *Faster and More Frequent Production Deployments.*
- Less time to market
  For Example: *Deploy to Production Without Manual Checks.*

- Reduced downtime from a deploy-related crash or major bug
  For Example: *Automated Smoke Tests*.
- Quick undo to return production to working state
  For Example: *Automated Rollback Triggered by Job Failure*.

UDAPEOPLE PIPELINE



THANK YOU