
epam
ENGX
ENGINEERING
EXCELLENCE
Continuous Integration
& Continuous Delivery
September 26, 2018

# ABOUT SPEAKER

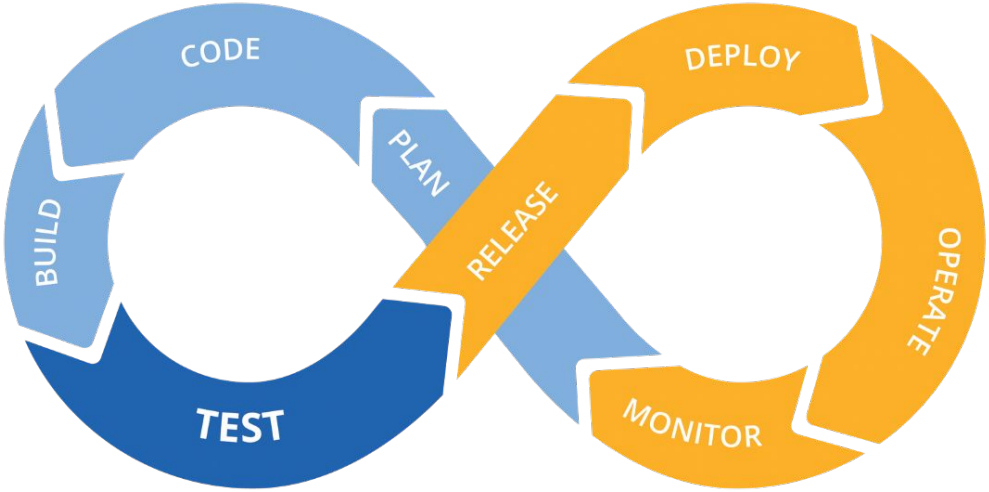
**Alexey Shcheglov**

**Project Manager**

15+ years in IT.
Project manager at EPAM Engineering Excellence Center.
We are responsible for propagation best engineering practices and tools across all EPAM locations.

# Agenda

- CI principles
- CD principles
- CI\CD tools



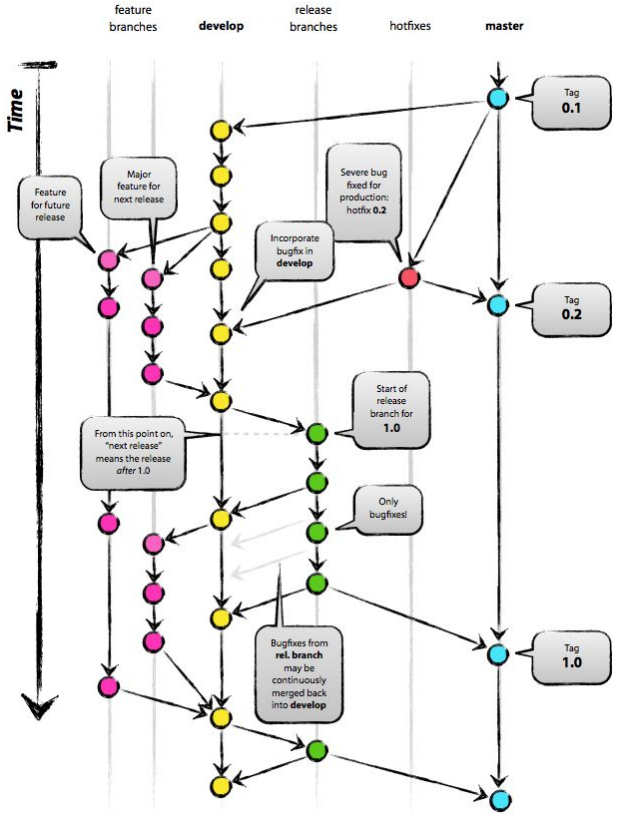
# CONTINUOUS INTEGRATION PRACTICES

**Continuous Integration (CI)** is a development practice that requires developers to integrate code into a shared repository several times a day.

- Commit every day
- Automate the build
- Test the build
- Keep the build fast
- Keep CI green
- Fix broken build immediately

# COMMIT EVERY DAY

- Easy code review
- Less merge conflicts
- Enables refactoring
- Risks mitigation
- Difficult task planning
- Incomplete feature separation
- Feature branches are not CI



# AUTOMATE & TEST THE BUILD

- Computers perform repetitive tasks, people solve problems
- If somethings difficult, do it more often.
- Single command to build full system
- Self-testing code & End-To-End testing
- Team seniority & Development discipline
- Additional costs

# KEEP THE BUILD FAST

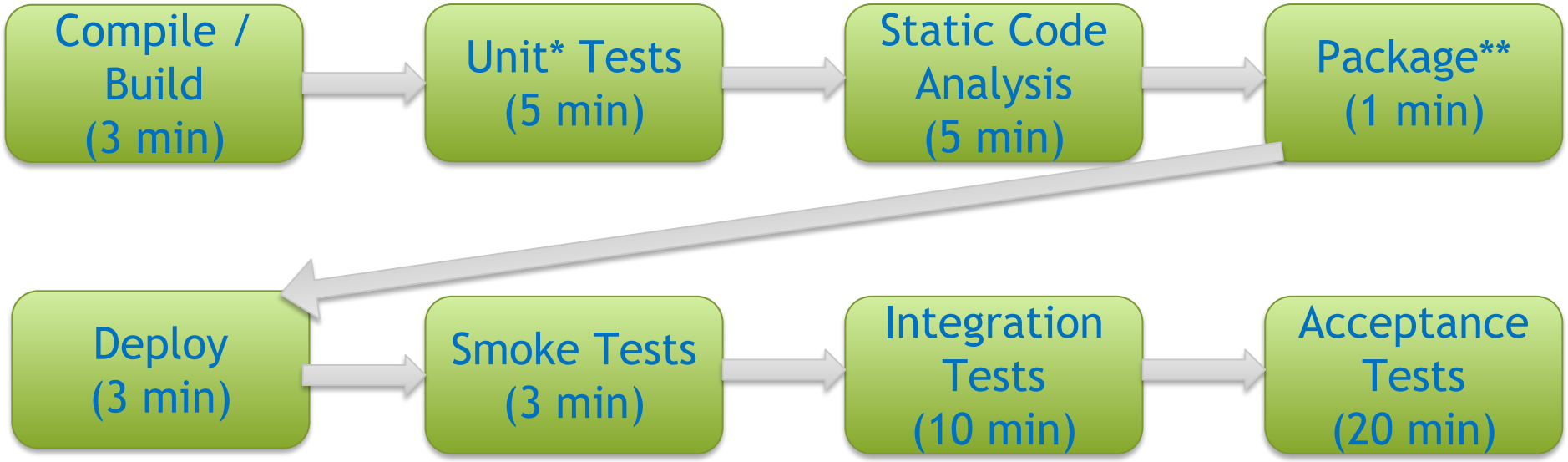
- Main CI point is fast feedback
- 10-15 minutes is OK
- Trade-off between time, tests and resources
- Several stages / pipelines
- Additional efforts
- Additional resources

# FIX BROKEN BUILD IMMEDIATELY

- Development on stable base
- Revert last commit
- Pre-commit verification
- Build Monitor & Build cop

- Build stability
- Development culture

# CI JOB / PIPELINE EXAMPLE



# CONTINUOUS DELIVERY PRACTICES

**Continuous Delivery (CD)** is an approach in which teams produce software ensuring that it can be reliably released at any time.

- Built-in quality
- Everybody has responsibility for the release process
- Build binaries only once
- Use the same mechanism to deploy to every environment

# BUILT-IN QUALITY

- Minimize manual testing
- Earlier defect discovery
- Defects covered by automated test
- Developers and AutoTesters are the same team
- Software Engineer in Testing (SET)
- Team experience

# DEVOPS CULTURE

- Development, operations and support have common goals
- Shared responsibility
- Automation

- It is not about build/system engineers

# BUILD BINARIES ONLY ONCE

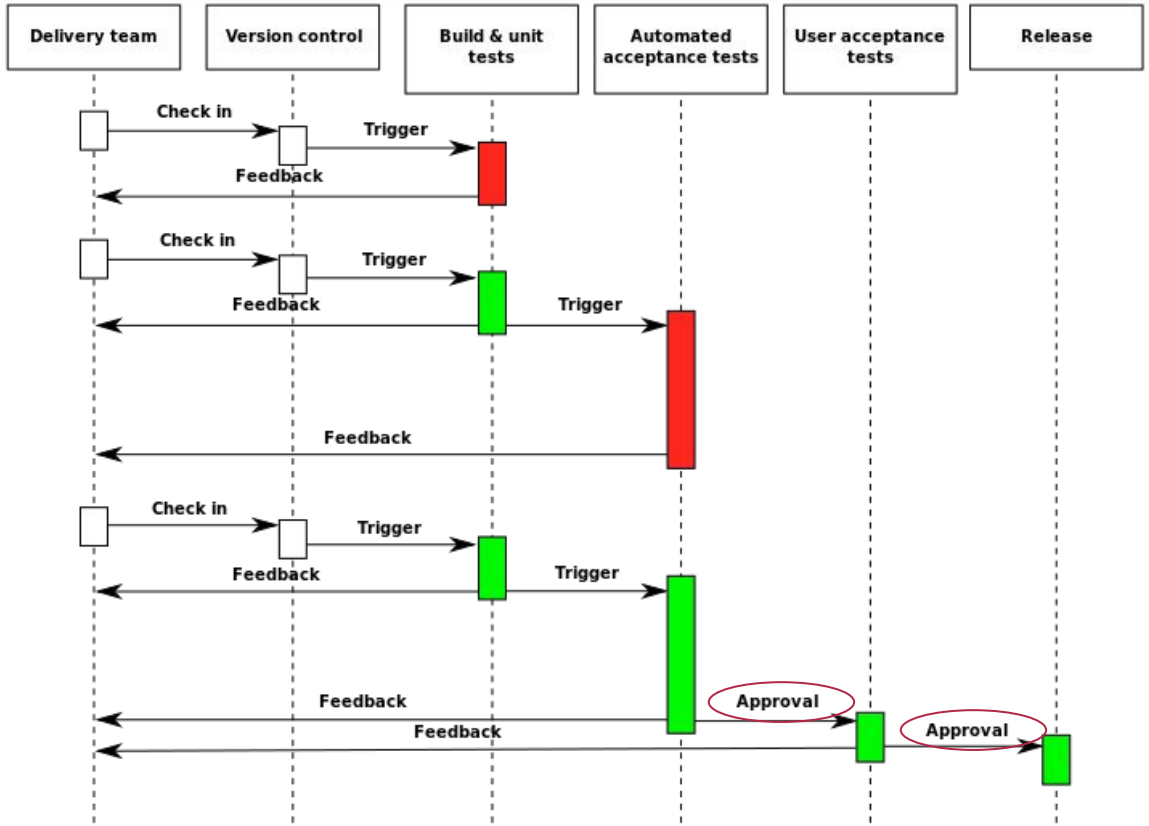
- Each build may produce different binary
- Speed-up deployment
- Binary Repositories (Nexus, Artifactory)

- Additional tool to support
- Difficulties in Dependency management

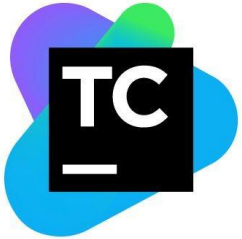
# THE SAME DEPLOYMENT APPROACH

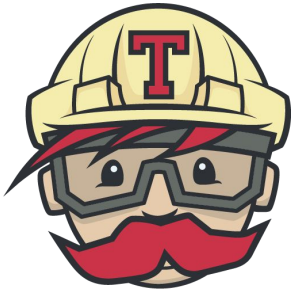
- As much as similar environments
- Deployment should be tested
- Deployment should be incorporated in CI/CD pipeline
- Are hot fixes and small fixes exceptions?
- Difficult to accomplish under pressure
- Needs very stable CI environment

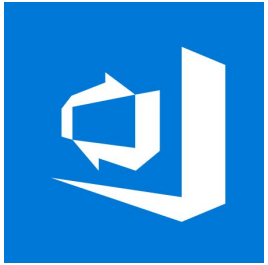
# CI\CD PIPELINE



# CI\CD TOOLS






Feel free to

# ASK YOUR QUESTIONS