Carnegie Mellon

DevOps-Continuous Integration

Deployment Pipeline

1

Carnegie Mellon University

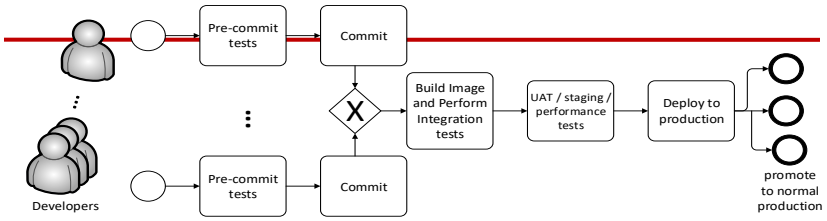
Overview

- **What is a deployment pipeline?**
- Testing
- Pipeline qualities

2

Carnegie Mellon University

Deployment pipeline



- Developer creates and tests code on local machine.
- Checks code into a version control system
- Continuous integration server (CI) builds the system and runs a series of integration tests.
- After passing the tests, the system is promoted to a staging environment where it undergoes more tests including performance, security, and user acceptance tests.
- After passing those tests, the system is promoted to provisional production where it undergoes even more tests.
- The system is finally promoted to normal production but the tests do not necessarily stop.

Carnegie Mellon University

Four environments

- We present the deployment pipeline as having four environments:
  - Development
  - Build (integration)
  - Staging
  - Production
- Different organizations may have more (or fewer) environments

Carnegie Mellon University

Development Environment

- Developer codes module
- Runs unit tests
- Checks results into version control system
- All developers should have the same development environment
  - Uniformity reduces errors that occur during integration and production
  - Environment created by IaC for a provisioning tool

Carnegie Mellon University

Integration environment

- Also called build environment
- Based on continuous integration server
- Creates executable image
- Runs system tests to check functionality

Carnegie Mellon University

Sample integration environment

- A sample integration environment can be found at

https://github.com/cmudevops/class-materials/blob/master/deployment%20workflow.pdf

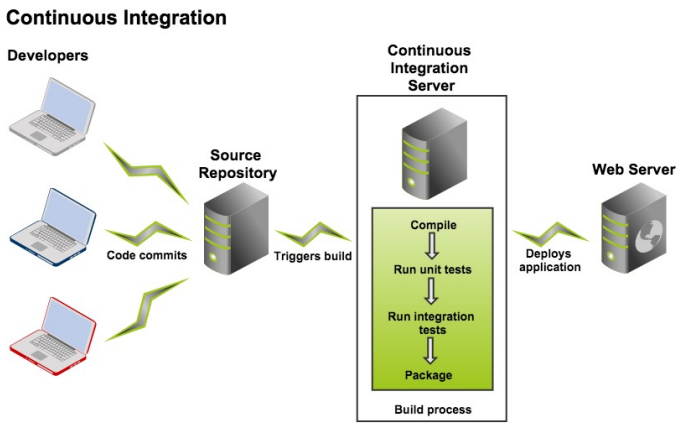
- Note all of the details

7

7

Carnegie Mellon University

Overview from Atlassian



8

8

Carnegie Mellon University

Creating an executable image

- Continuous Integration (CI) server is notified of new commits (or polls for them)
- When a new commit is detected, the CI server retrieves it
- The CI server retrieves all relevant code and their dependencies
- Executable image is created by compiling and linking all of the code and their dependencies.

Carnegie Mellon University

Build has multiple users

- Build is performed by a continuous integration server (CI server)
- CI is single threaded from perspective of systems (one build of a given system at a time)
- Build error introduced by one developer affects all developers

Carnegie Mellon University

Staging environment

- The staging environment is used for non-functional tests.
  - Performance
  - Security
  - Licensing
  - User
- We will discuss these in more detail in the next section.

Carnegie Mellon University

Overview

- What is a deployment pipeline?
- **Testing**
- Pipeline qualities

Carnegie Mellon University

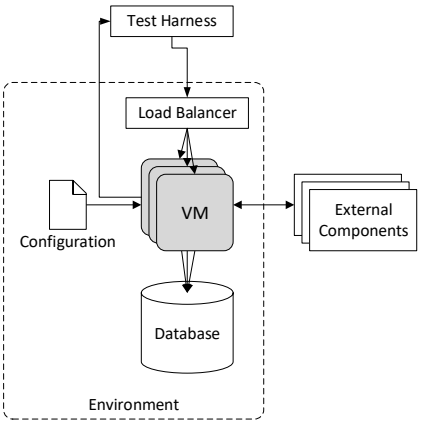
Testing

- Every stage (except production) has an associated test harness.
- Tests should be automated.
- Types of tests are
  - Sunny day tests
  - Negative tests
  - Regression tests
  - Static analysis
- Testing during production will be discussed when we get to chaos engineering

Carnegie Mellon University

Test Harness

A test harness generates inputs and compares outputs to verify the correctness of the code.



Carnegie Mellon University

Database test data

- Every stage has a database
- Minimal during development
- More substantial during build
- Realistic during staging
- After each test, database must be reconstituted to ensure repeatability
- Personally Identifiable Information (PII) must be obscured.

Carnegie Mellon University

Test tools

- Junit : Testing framework for the Java/J2EE
- Nunit: Testing framework with GUI for .Net platform
- MSTest: Testing framework with CL outside Visual Studio
- PYunit: Testing framework with Python
- TestNG: Very similar to Junit but covers more (functional, end-to-end, integration etc.)

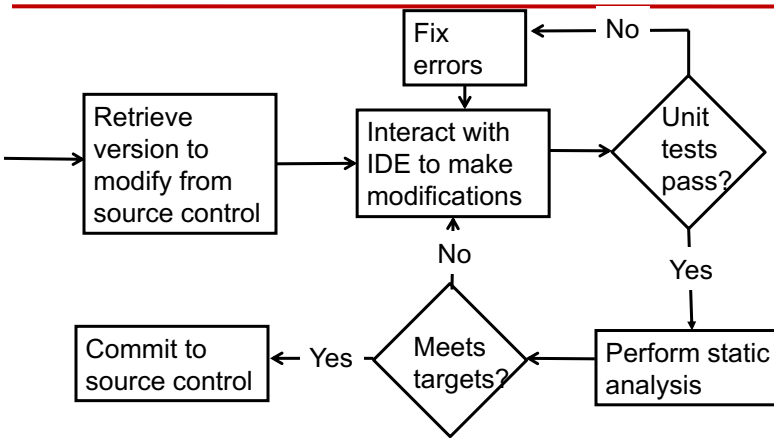
Carnegie Mellon University

Tests during development

- Unit tests on a single module.
- Dependencies may be included or stubbed.
- May use test harness or a static analyzer.
- Static analyzer looks at source code and locates potentially troublesome code structures.
  - May have false positives or false negatives

Carnegie Mellon University

Workflow during development



Carnegie Mellon University

Tests during integration

- Unit tests for each of the modules included in the build
- System wide functional tests
- Types of errors discovered include
  - Incorrect use of interfaces
  - Violation of contracts of use
  - Inconsistent libraries
  - Incorrect or inconsistent configuration parameters
  - …

Carnegie Mellon University

Staging tests

- Performance
- Security
- Licensing
- User

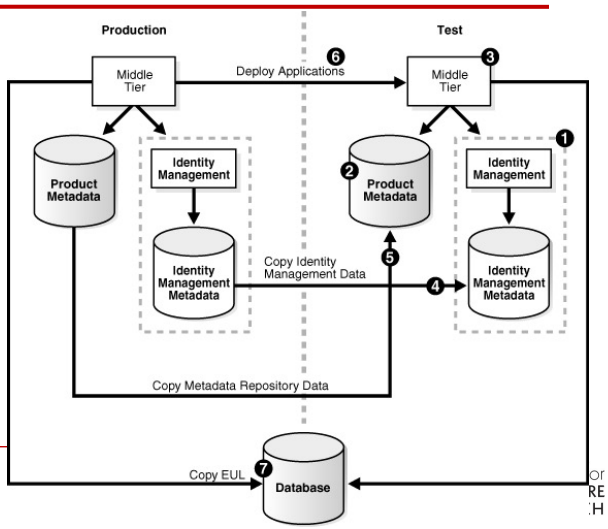
Carnegie Mellon University

Creating the staging data base

This is Oracle's recommendation.

EUL is end user layer.

EUL can be used to protect PII although this is not an ideal solution for testing.



Carnegie Mellon University

Performance testing

- Test system under load
- Tools to generate synthetic loads (e.g. Artillery)
- May also use production requests
  - Saved from actual system on prior input
  - Teed from actual input

Carnegie Mellon University

Security testing

- Common Vulnerabilities and Exposures data base (CVE)
  - Buffer overflow
  - SQL injection
- Penetration testing tools
- Static analyzers
- Fuzz Testing
- Model checking

Carnegie Mellon University

Security test cases

- Sample security test cases
  - A password should be in encrypted format
  - Application or System should not allow invalid users
  - Check cookies and session time for application
  - For financial sites, the browser back button should not work.
- Partially automated, partially manual

Carnegie Mellon University

Licensing testing

- Every piece of software used, including open source and COTS comes with a license. E.g.
  - License must be included in derivative works
  - Only for non-commercial systems
  - Included code may not be modified
  - …
- Conformance to these licenses must be verified.

institute for SOFTWARE RESEARCH

Carnegie Mellon University

User testing

- Automated user testing tools
- May have real users with full functionality (canary testing)
- May have real users with limited functionality. E.g. financial systems functions with no modifications allowed

institute for SOFTWARE RESEARCH

Carnegie Mellon University

Overview

- What is a deployment pipeline?
- Testing
- **Pipeline qualities**

institute for SOFTWARE RESEARCH

Carnegie Mellon University

Desirable qualities of deployment pipeline

- Traceability
- Cycle time

institute for SOFTWARE RESEARCH

Carnegie Mellon University

Traceability

When any code gets into production, it must be reconstructable

- All components that go into the executable image can be identified
- All tests that were run can be identified
- All scripts used during the pipeline process can be identified.

When a problem occurs in the system when it goes into production, all of the elements that contributed to the system can be traced.

Carnegie Mellon University

Achieving traceability

- Tracing database
- Tools must enter information into database
  - Test case ids
  - Version and release numbers of included code and dependencies

Carnegie Mellon University

Cycle time

- Cycle time is the time between the commit and the placing into provisional production.
- Systems should move through the pipeline quickly
- The time depends on
  - How large are the components that are constructed in the build stage
  - How long does it take to run tests on the system

Carnegie Mellon University

Summary

- Deployment pipeline consists of sequence of environments managed by tools
- Each environment has types of tests that are run in that environment
- Management of database for tests is important
- Traceability and cycle time are important qualities of a deployment pipeline.