Quality Assurance In Microservice Architectures

Krishnan Chandran   Irina Barykina

Department of Informatics,
Intelligent Adaptive Systems, UHH

2016

-
-

Continuous Delivery

Martin Fowler's definition:

Continuous delivery is a software development discipline where you build software in such a way that it can be released to production at any time. Principle benefits: reduced deployment risk, transparency, immediate user feedback.

- ▶ Key prerequisites for CD: extensive automation of all possible parts of the delivery process (DeploymentPipeline) and close collaboration between everyone involved in delivery (DevOpsCulture)
- ▶ Drawbacks of monolithic architecture from CD perspective: long build time, delayed feedback (because of large number of people checking in changes simultaneously), unclear responsibility for broken builds or bugs, time consuming test runs (automatic and manual), small change can affect the whole application. However, it is not impossible to maintain CD with monolithic architecture

Tools

- ► Unit Testing
  - ► Sociable Unit testing vs. Solitary unit testing
  - ► Tool 1: xUnit family of testing frameworks derived from SmallTalk testing framework, designed by Kent Beck.
  - ► Tool 2: Frameworks for mocking and stubbing, for example Mockito
- ► Contract Testing
  - ► Integration contract testing: performed by using of test doubles of service that is to be consumed
    - ► Mocking
    - ► Stubbing
  - ► Consumer driven contract testing: performed by provider against consumer defined contracts
    - ► Pact: Analogies: Pacto, Janus
    - ► Data passing between services like SOAP UI

End-to-end testing

Subtitles are optional

- ► End-to-end testing tools

<The UML diagram of example architecture>

- ▶ Scenario 1: Testing microservices within application.
    - ▶ Unit testing:
    - ▶ Contract testing:
    - ▶ End-to-end testing:
- ▶ Scenario 2: Testing microservices that use third-party service
    - ▶ Unit testing:
    - ▶ Contract testing:
    - ▶ End-to-end testing:
- ▶ Scenario 3: Testing of microservice that will be or is already exposed to public domain
    - ▶ Unit testing:
    - ▶ Contract testing:
    - ▶ End-to-end testing:

- ▶ Books
- ▶ Articles and blogs