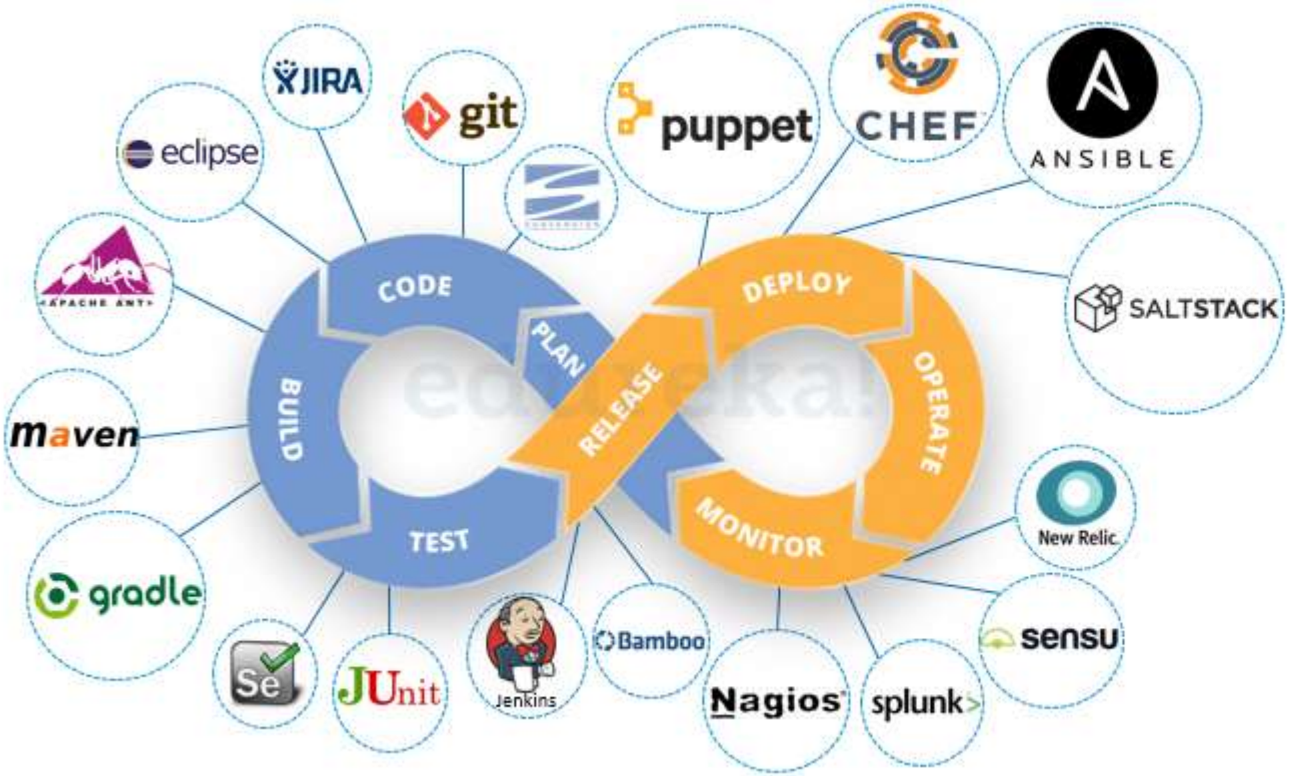
DevOps


CODE
PLAN
BUILD
TEST
RELEASE
DEPLOY
OPERATE
MONITOR
JIRA
git
eclipse
APACHE ANT
maven
gradle
Se
JUnit
Jenkins
Bamboo
Nagios
splunk>
sensu
New Relic
SALTSTACK
ANSIBLE
CHEF
puppet
edureka!

DevOps is not a tool or technology; it is an approach or culture that makes things better.

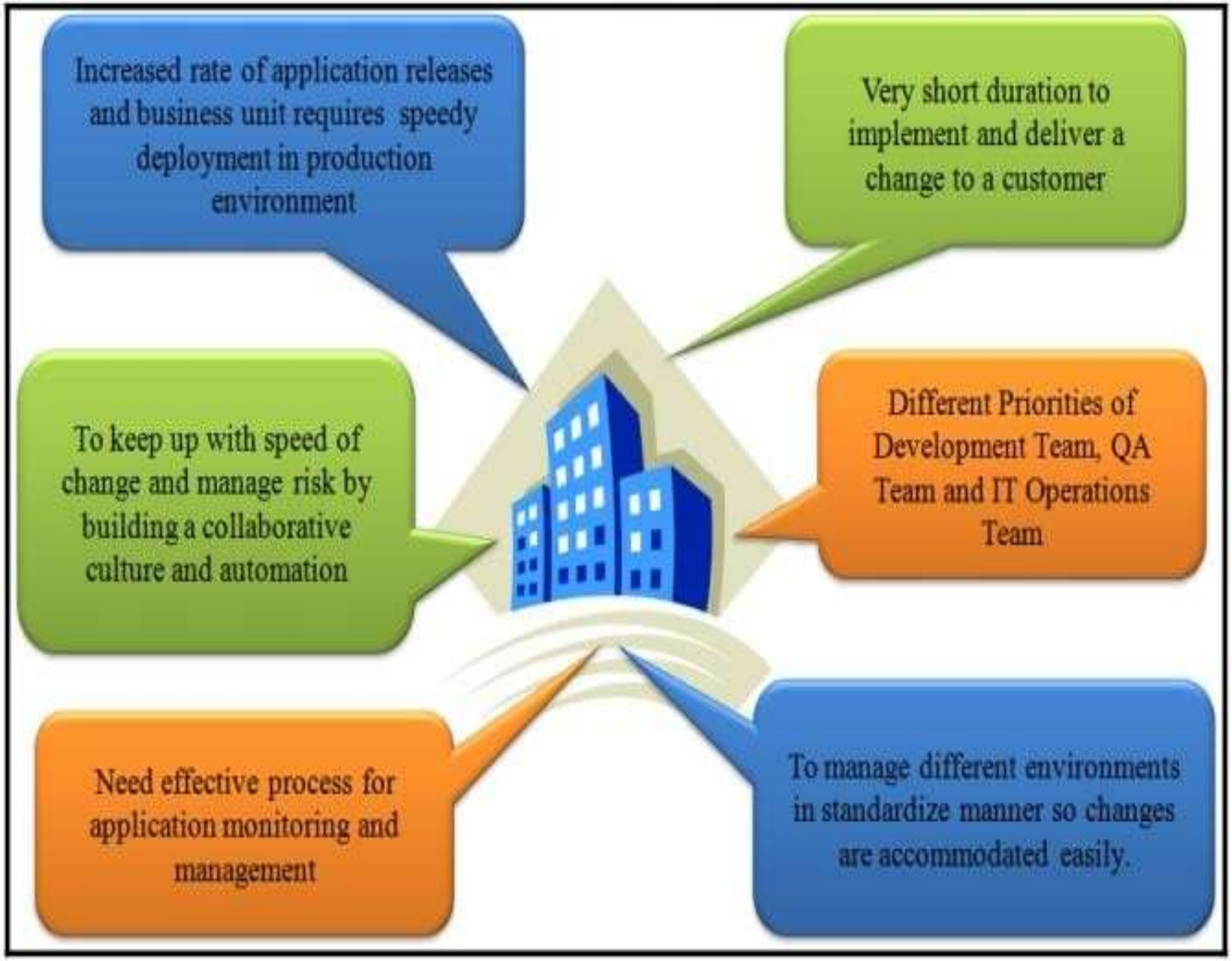
DevOps is a culture and set of process that brings development team and operation team together to complete the software development.

Both team works together in entire SFDC to ensure the quality of product, ability of product to work in different environments.

Lots of tool are available in market to automate this process

DevOps with the changing times

- Time changes everything. In the modern era, customers expect and demand extremely quick response, and we need to deliver new features continuously to stay in business.



Increased rate of application releases and business unit requires speedy deployment in production environment

Very short duration to implement and deliver a change to a customer

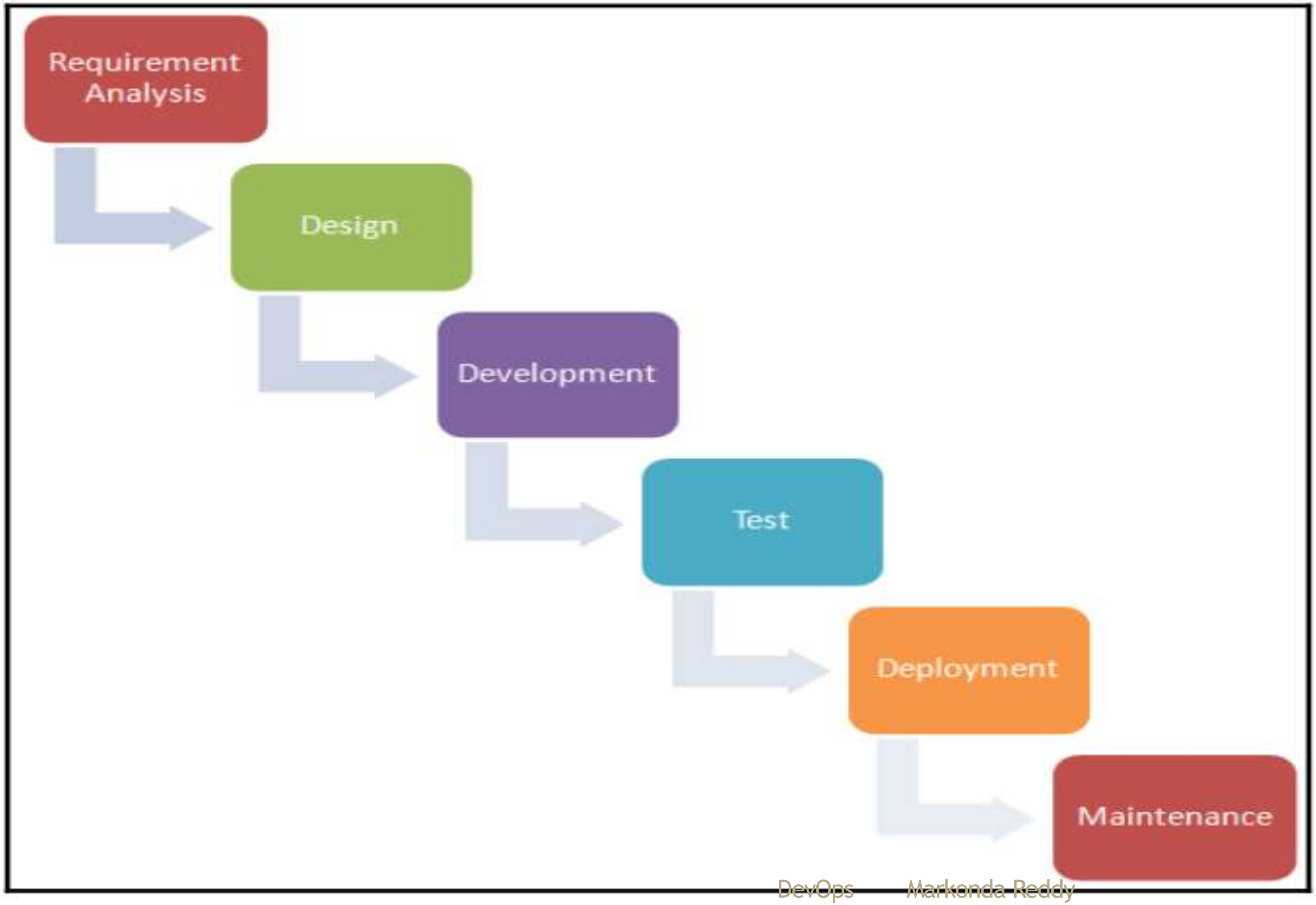
To keep up with speed of change and manage risk by building a collaborative culture and automation

Different Priorities of Development Team, QA Team and IT Operations Team

Need effective process for application monitoring and management

To manage different environments in standardize manner so changes are accommodated easily.

The waterfall model



Advantages and disadvantages

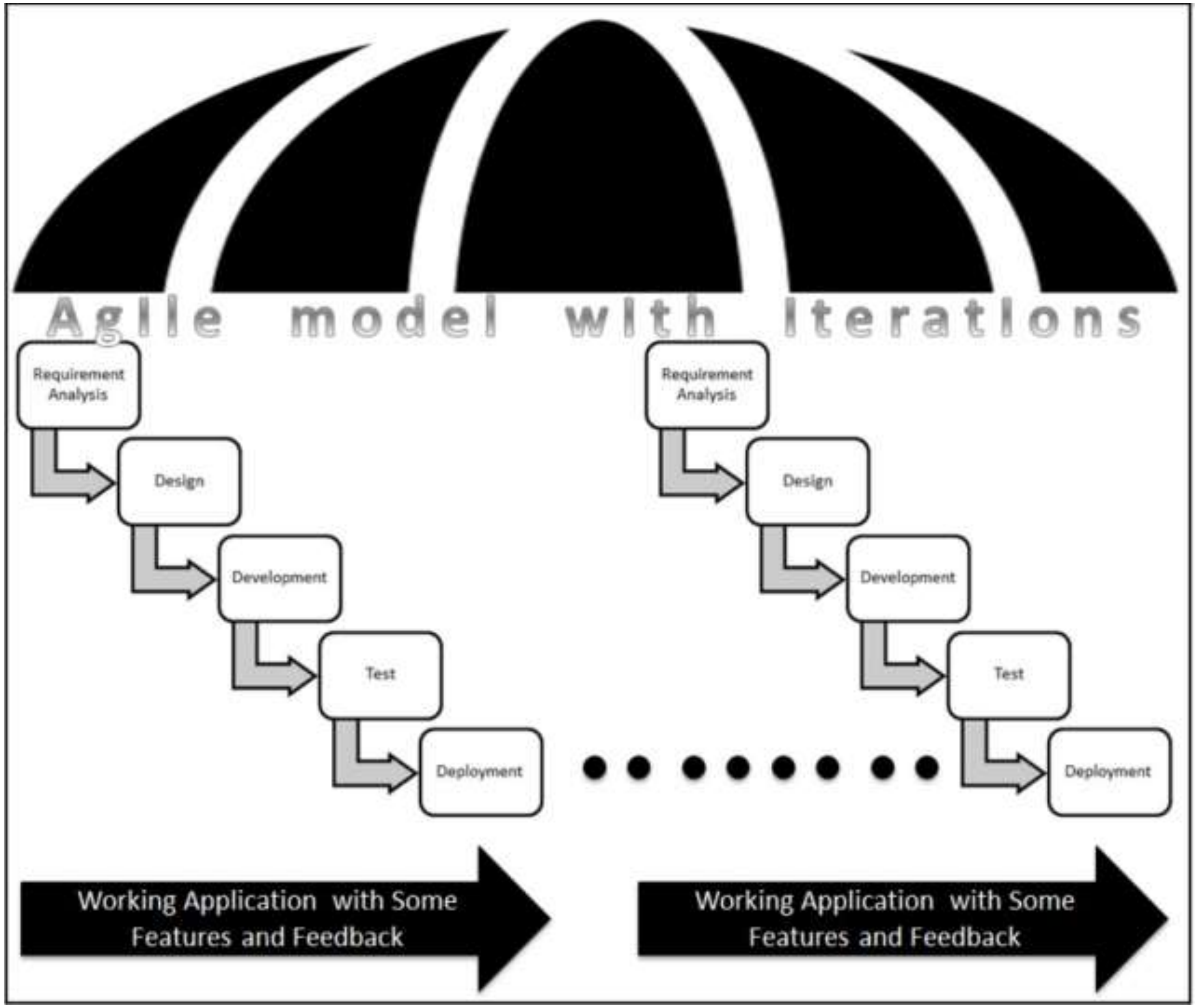
Advantages

1. Easy to understand
2. Easy to manage— the input and output of each phase is defined
3. Sequential process— order is maintained Better control

Disadvantages

1. No revision
2. No outcome or application package until all phases are completed
3. Not possible to integrate feedback until all phases are completed
4. Not suitable for changing requirements
5. Not suitable for long-term and complex projects

The agile model


Agile model with iterations
Requirement Analysis
Design
Development
Test
Deployment
Requirement Analysis
Design
Development
Test
Deployment
Working Application with Some Features and Feedback
Working Application with Some Features and Feedback

- One of the most attractive benefits of agile development is continuous delivery in short time frames or, in agile terms, sprints. Now, it is not a one-time deployment, but multiple deployments.

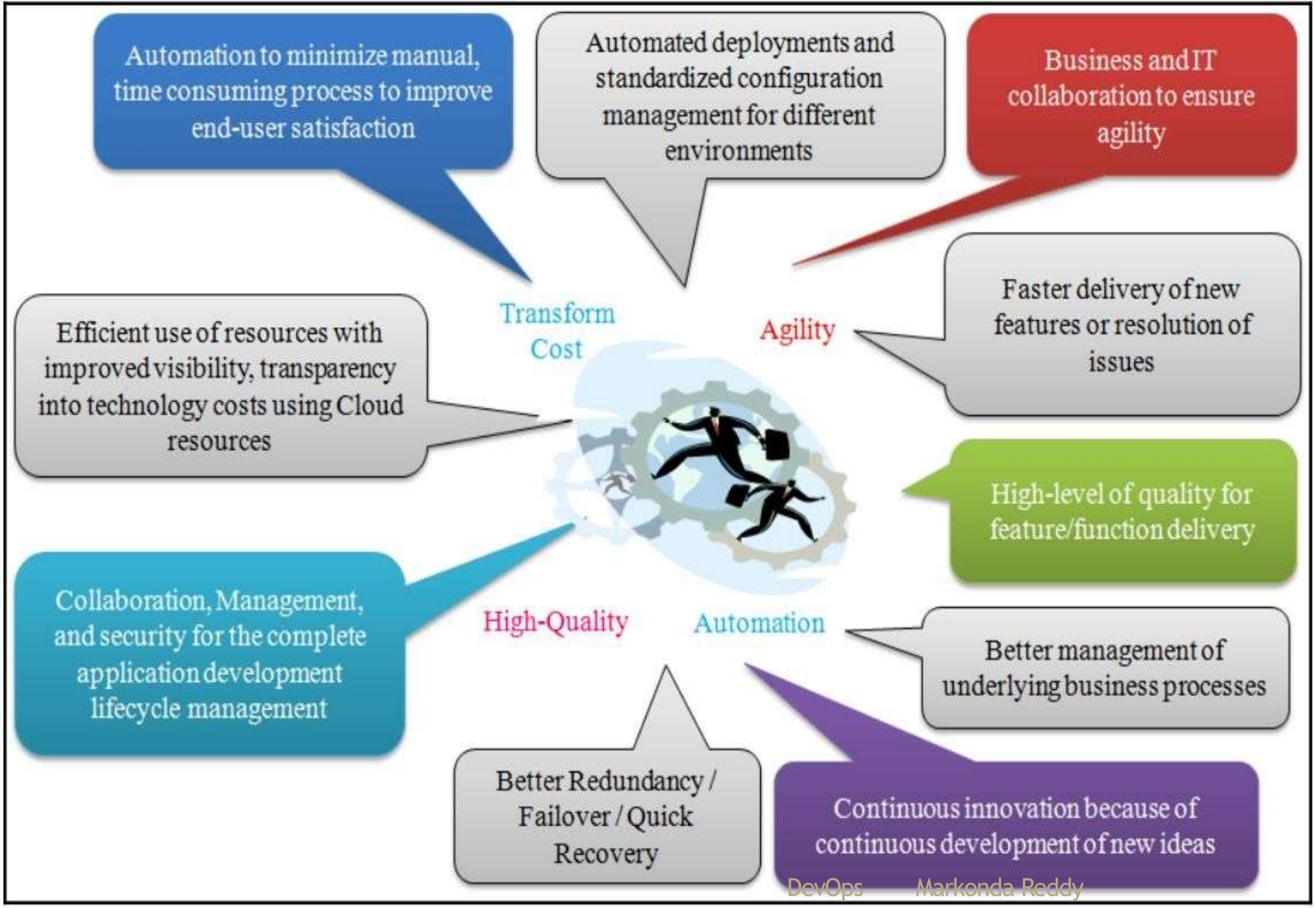
Collaboration

- DevOps attempts to fill these gaps by developing a partnership between the development and operations teams. The DevOps movement emphasizes communication, collaboration, and integration between software developers and IT operations.
- DevOps promotes collaboration, and collaboration is facilitated by automation and orchestration in order to improve processes. In other words, DevOps essentially extends the continuous development goals of the agile movement to continuous integration and release.
- DevOps is a combination of agile practices and processes leveraging the benefits of cloud solutions.

Why DevOps?

- DevOps is effective because of new methodologies, automation tools, agile resources of cloud service providers, and other disruptive innovations, practices, and technologies. However, it is not only about tools and technology-DevOps is more about culture than tools or technology alone.

The benefits of DevOps



Automation to minimize manual, time consuming process to improve end-user satisfaction

Automated deployments and standardized configuration management for different environments

Business and IT collaboration to ensure agility

Transform Cost

Agility

Efficient use of resources with improved visibility, transparency into technology costs using Cloud resources

Faster delivery of new features or resolution of issues

High-level of quality for feature/function delivery

Collaboration, Management, and security for the complete application development lifecycle management

High-Quality

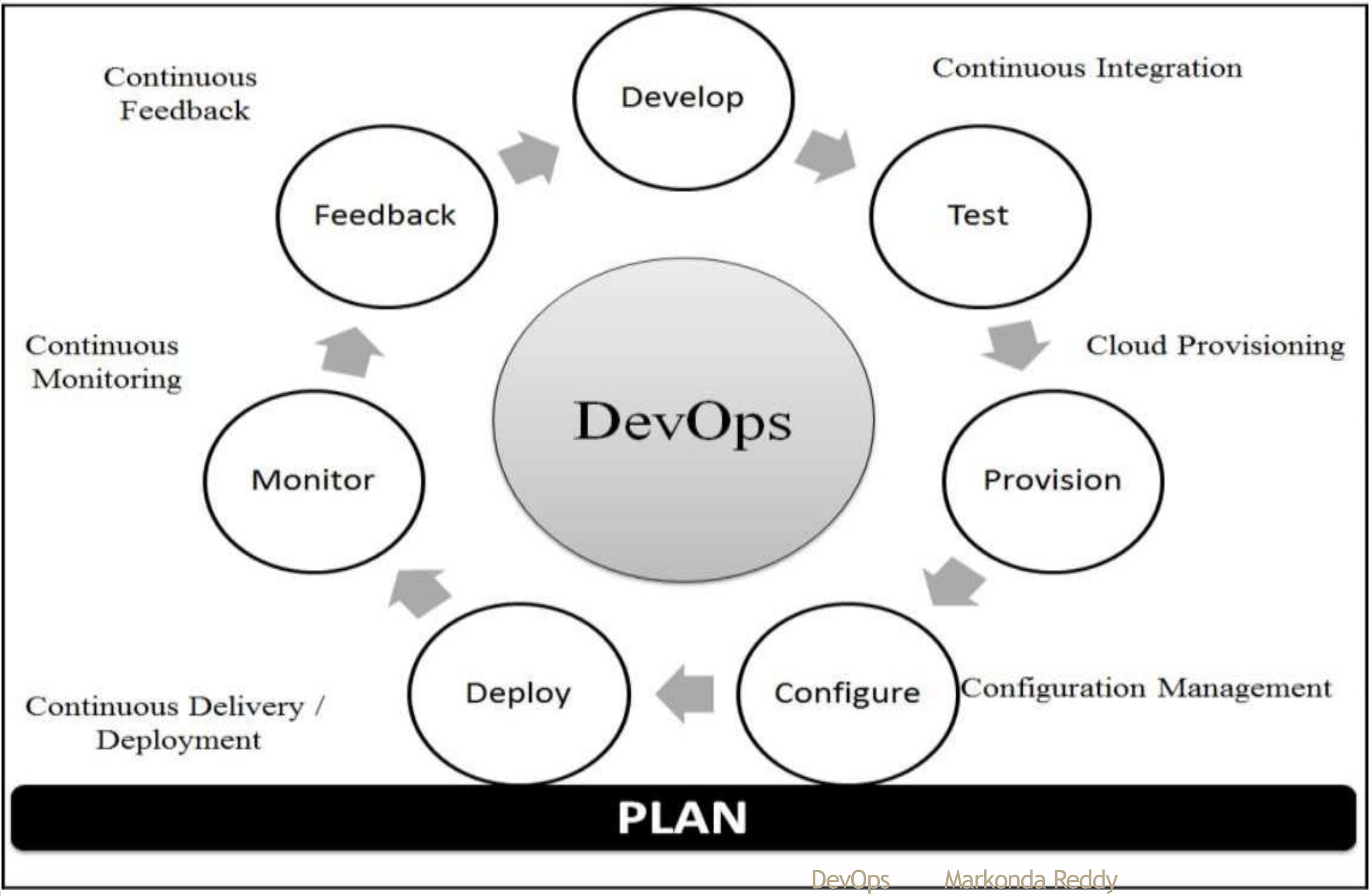
Automation

Better management of underlying business processes

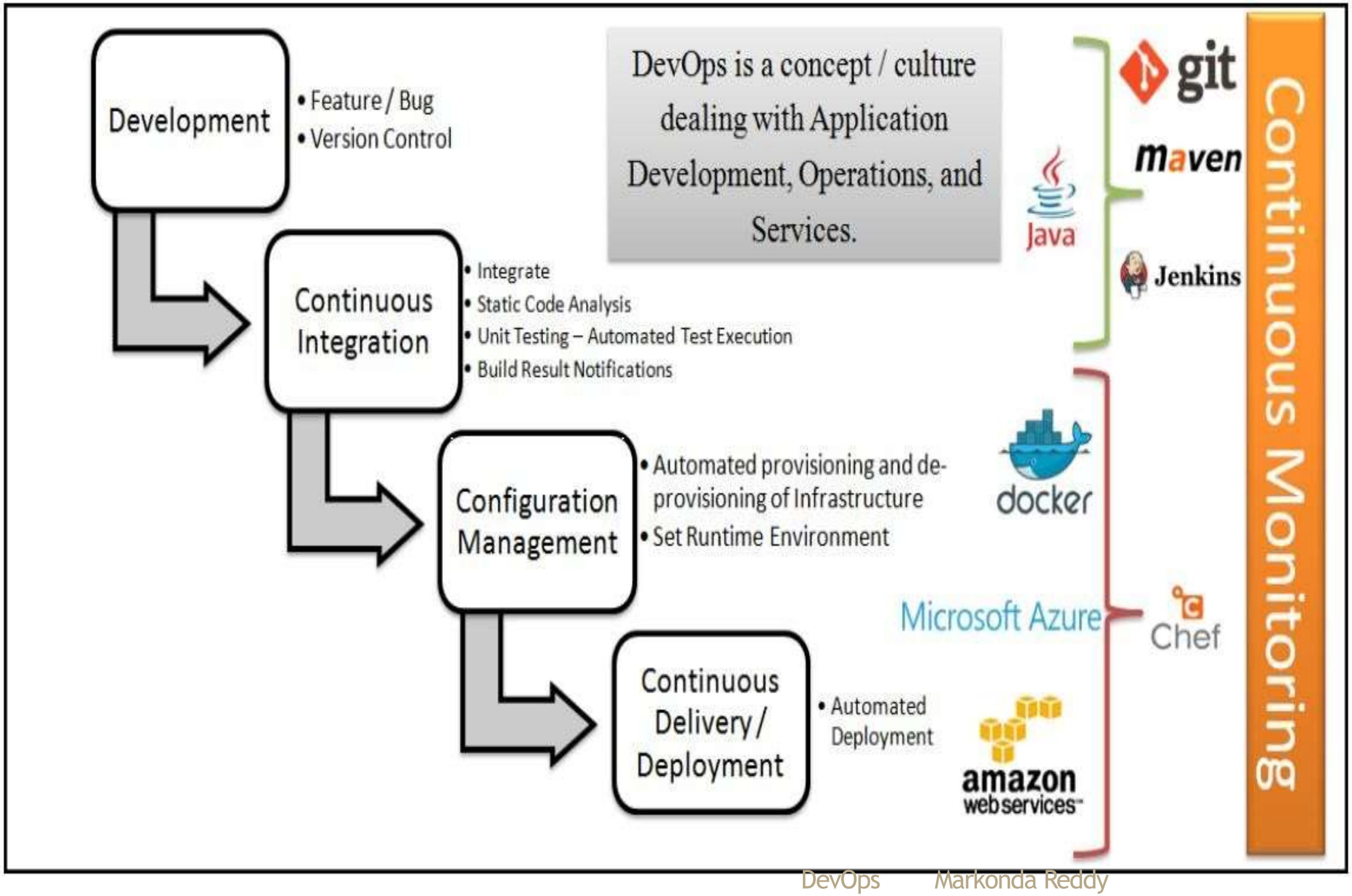
Better Redundancy / Failover / Quick Recovery

Continuous innovation because of continuous development of new ideas

The DevOps lifecycle – it's all about "continuous"



Application delivery pipeline with the toolset



Development

- Feature / Bug
- Version Control

Continuous Integration

- Integrate
- Static Code Analysis
- Unit Testing – Automated Test Execution
- Build Result Notifications

Configuration Management

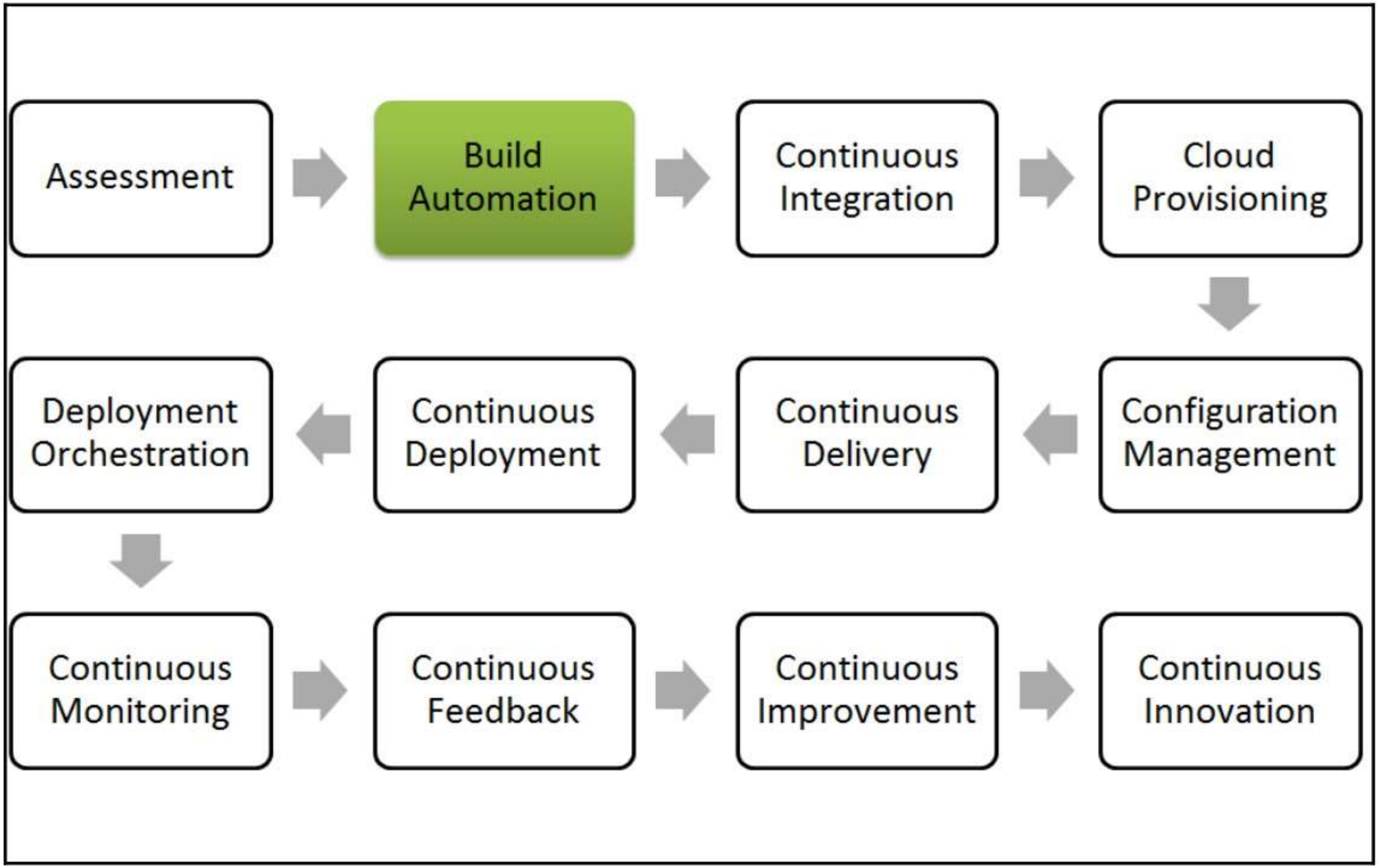
- Automated provisioning and de-provisioning of Infrastructure
- Set Runtime Environment

Continuous Delivery / Deployment

- Automated Deployment

DevOps is a concept / culture dealing with Application Development, Operations, and Services.

Build automation



Assessment

Build Automation

Continuous Integration

Cloud Provisioning

Configuration Management

Continuous Delivery

Continuous Deployment

Deployment Orchestration

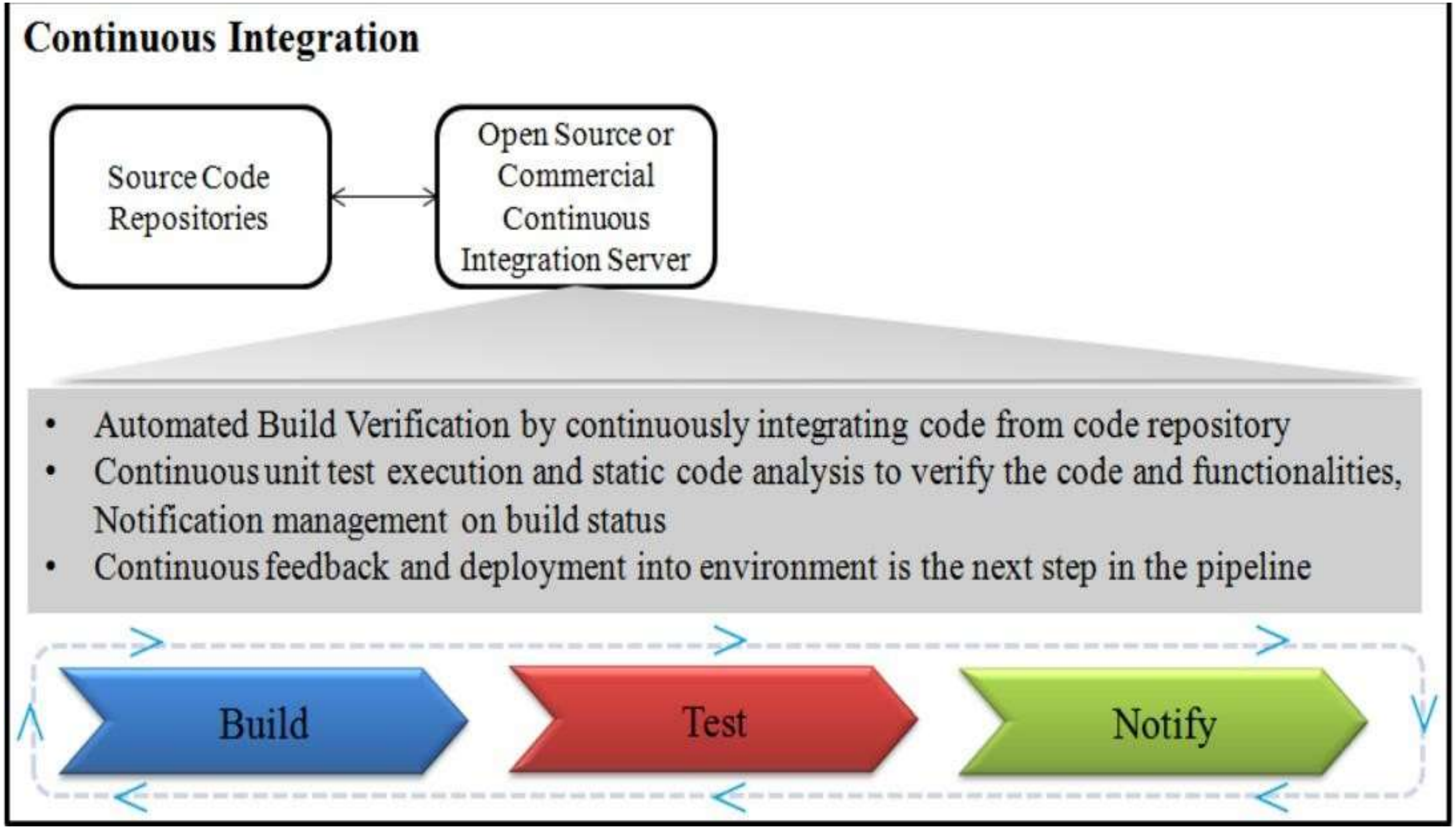
Continuous Monitoring

Continuous Feedback

Continuous Improvement

Continuous Innovation

Continuous integration



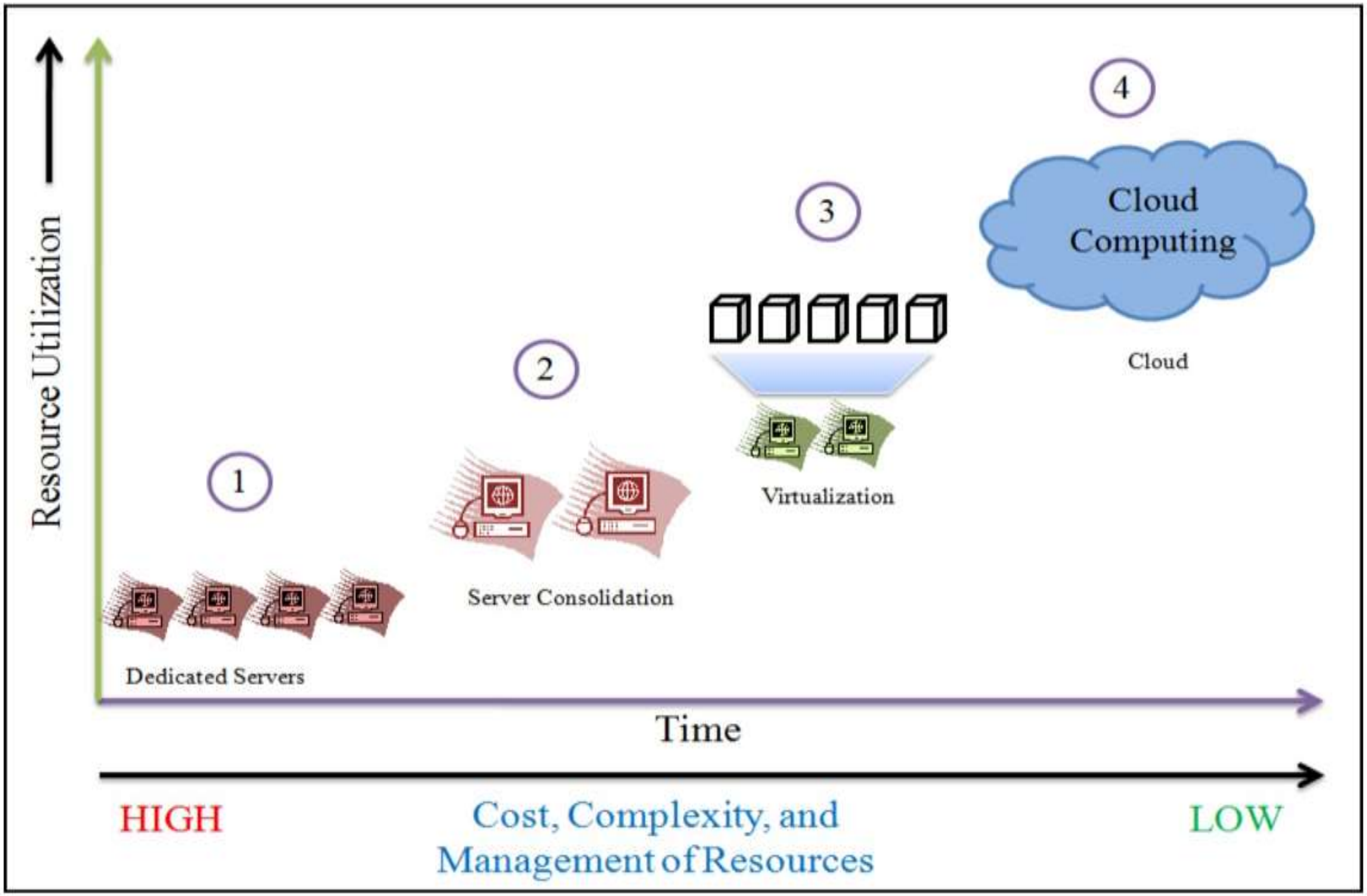
Continuous Integration

- Automated Build Verification by continuously integrating code from code repository
- Continuous unit test execution and static code analysis to verify the code and functionalities, Notification management on build status
- Continuous feedback and deployment into environment is the next step in the pipeline

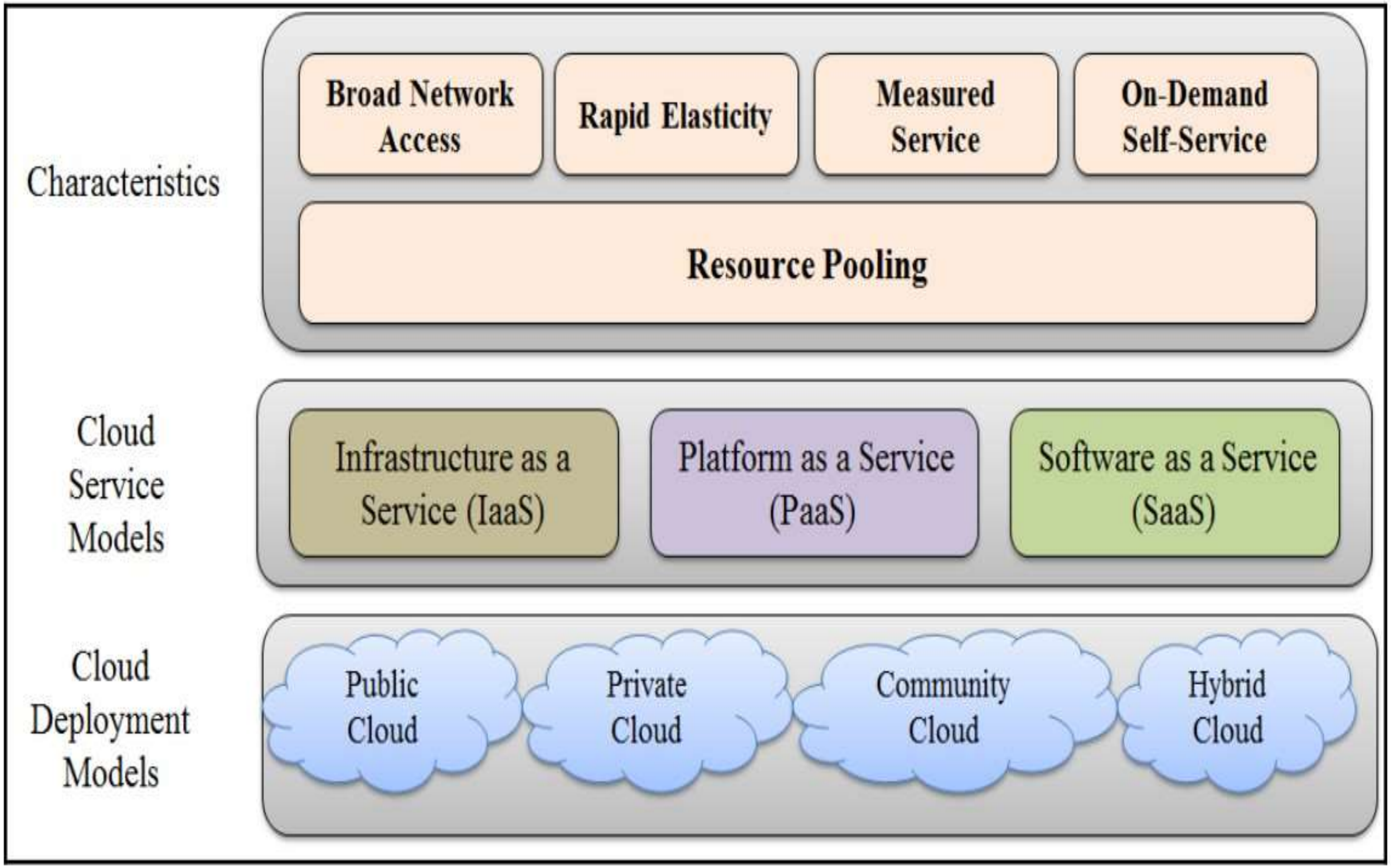
Benefits of CI

- Automated integration with pull or push mechanism
- Repeatable process without any manual intervention
- Automated test case execution
- Coding standard verification
- Execution of scripts based on requirement
- Quick feedback: build status notification to stakeholders via e-mail
- Teams focused on their work and not in the managing processes

Cloud computing



Deployment environments, considering three service models and four deployment models



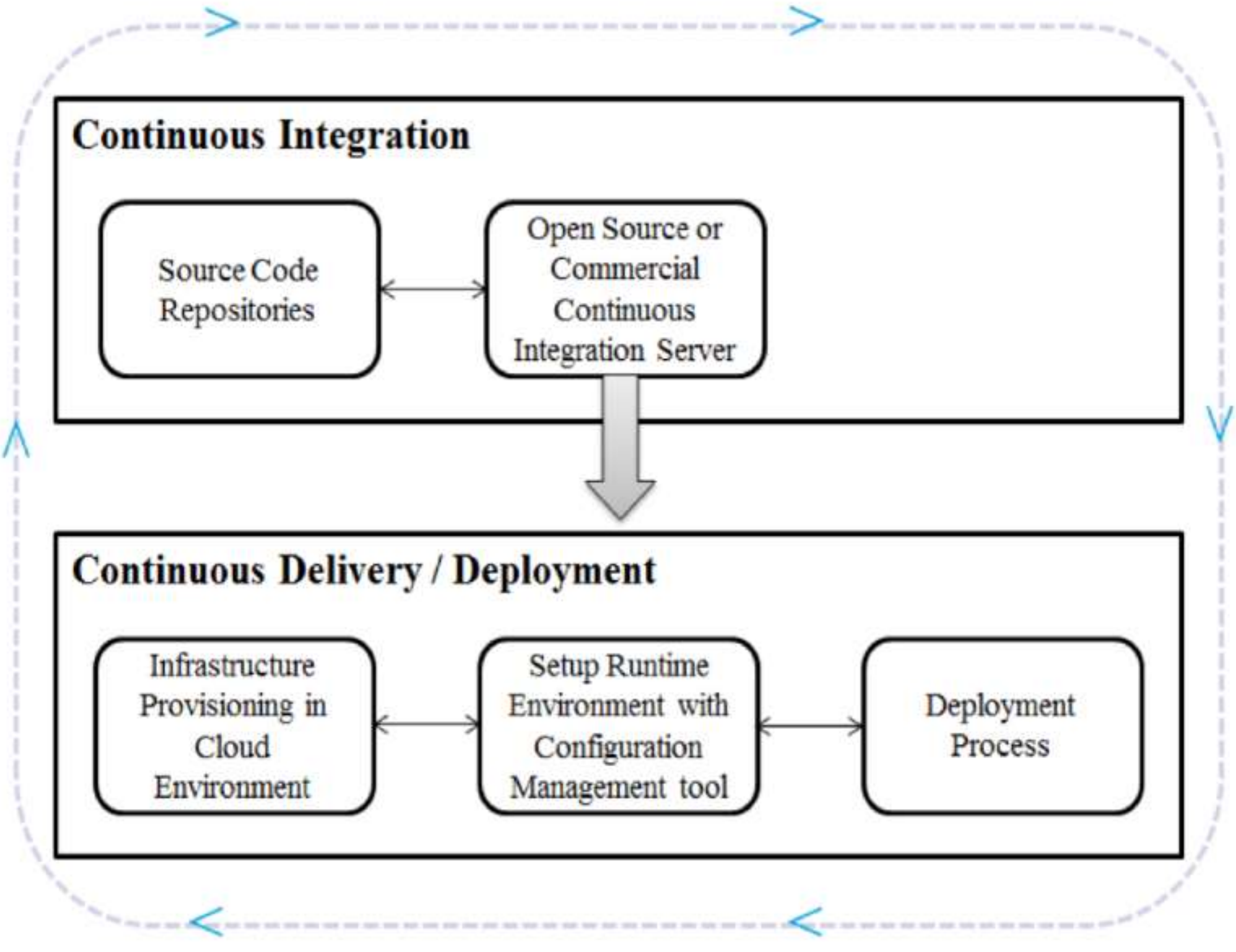
- **Public cloud:**This cloud Infrastructure is available to the general public
- **Private cloud:**This cloud Infrastructure is operated for and by a single organization
- **Community cloud:**This cloud infrastructure is shared by specific community that has shared concerns
- **Hybrid cloud:**This cloud infrastructure is a composition of two or more cloud models

Configuration management

- Configuration management(CM) manages changes in the system or, to be more specific, the server runtime environment. Let's consider an example where we need to manage multiple servers with same kind of configuration. For example, we need to install Tomcat on each server. What if we need to change the port on all servers or update some packages or provide rights to some users? Any kind of modification in this scenario is a manual and, if so, error-prone process. As the same configuration is being used for all the servers, automation can be useful here.

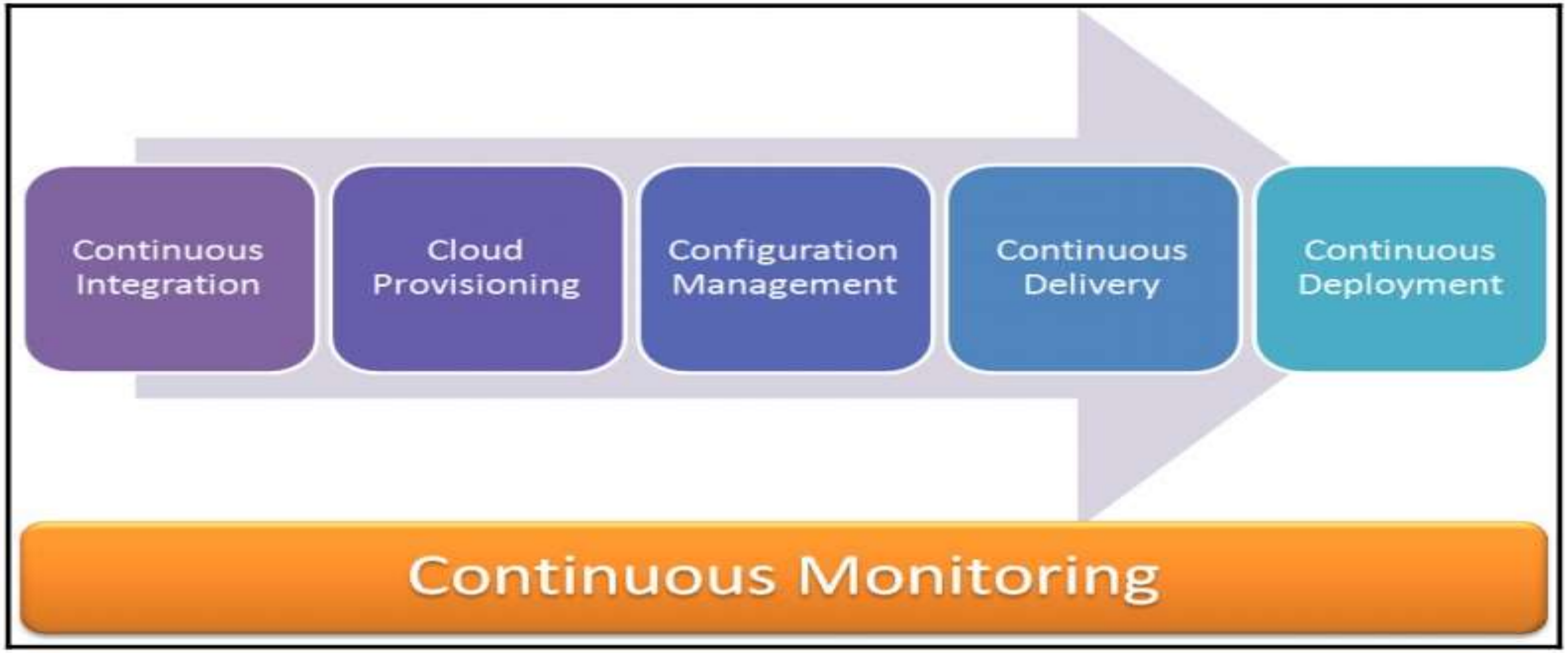
Continuous delivery/continuous deployment

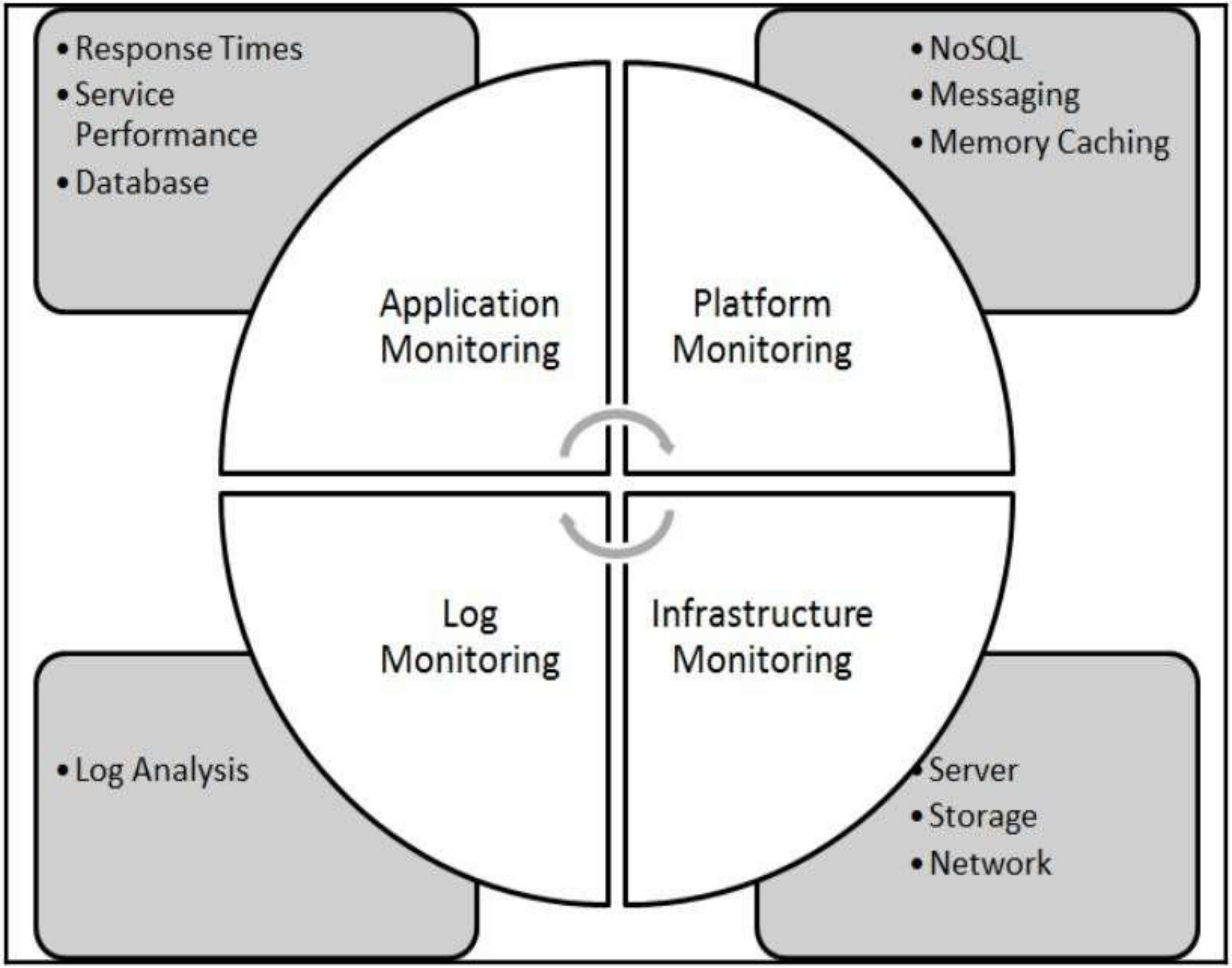
- **Continuous delivery** is a process of deploying an application in any environment in an automated fashion.
- **Continuous deployment** on the other hand, is all about deploying an application with the latest changes to the production environment.


Continuous Integration
Source Code Repositories
Open Source or Commercial Continuous Integration Server
Continuous Delivery / Deployment
Infrastructure Provisioning in Cloud Environment
Setup Runtime Environment with Configuration Management tool
Deployment Process

Production ready application by using automated process throughout application lifecycle.

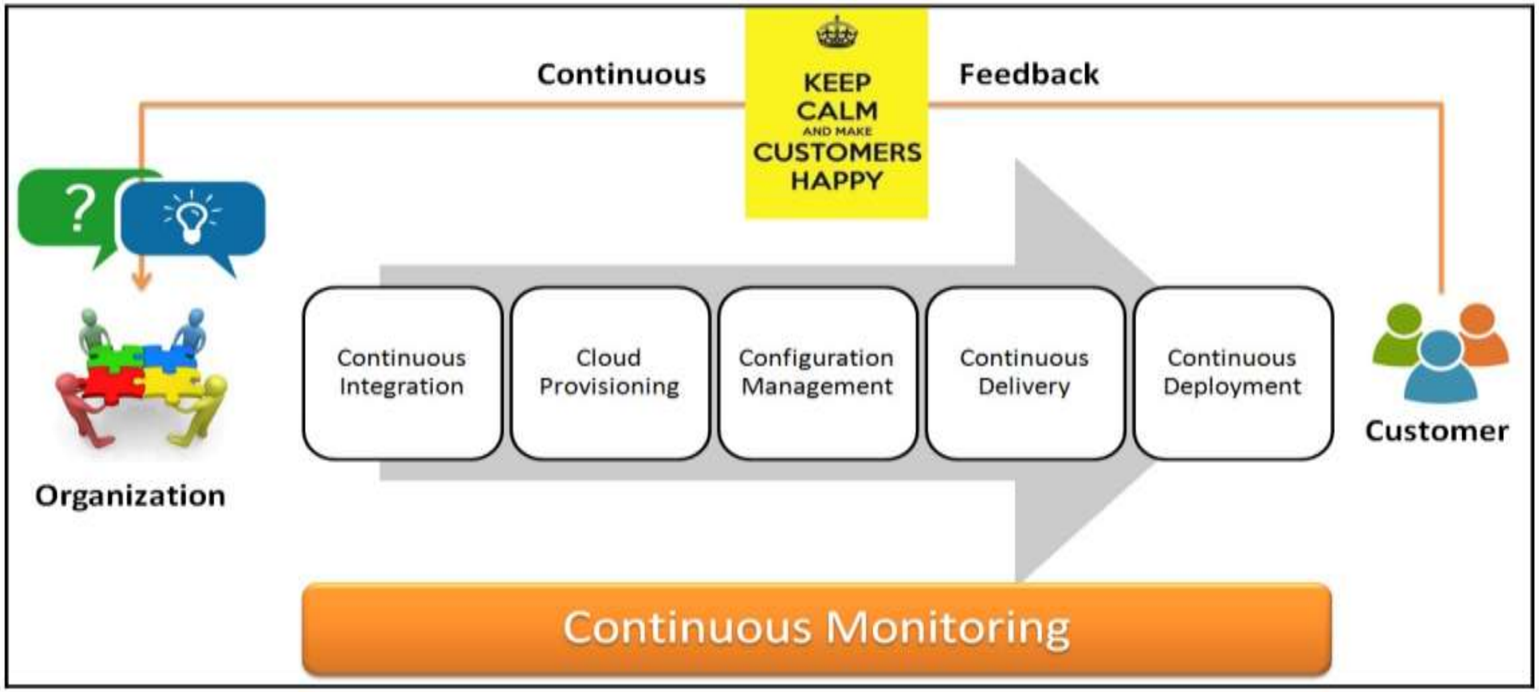
Continuous monitoring



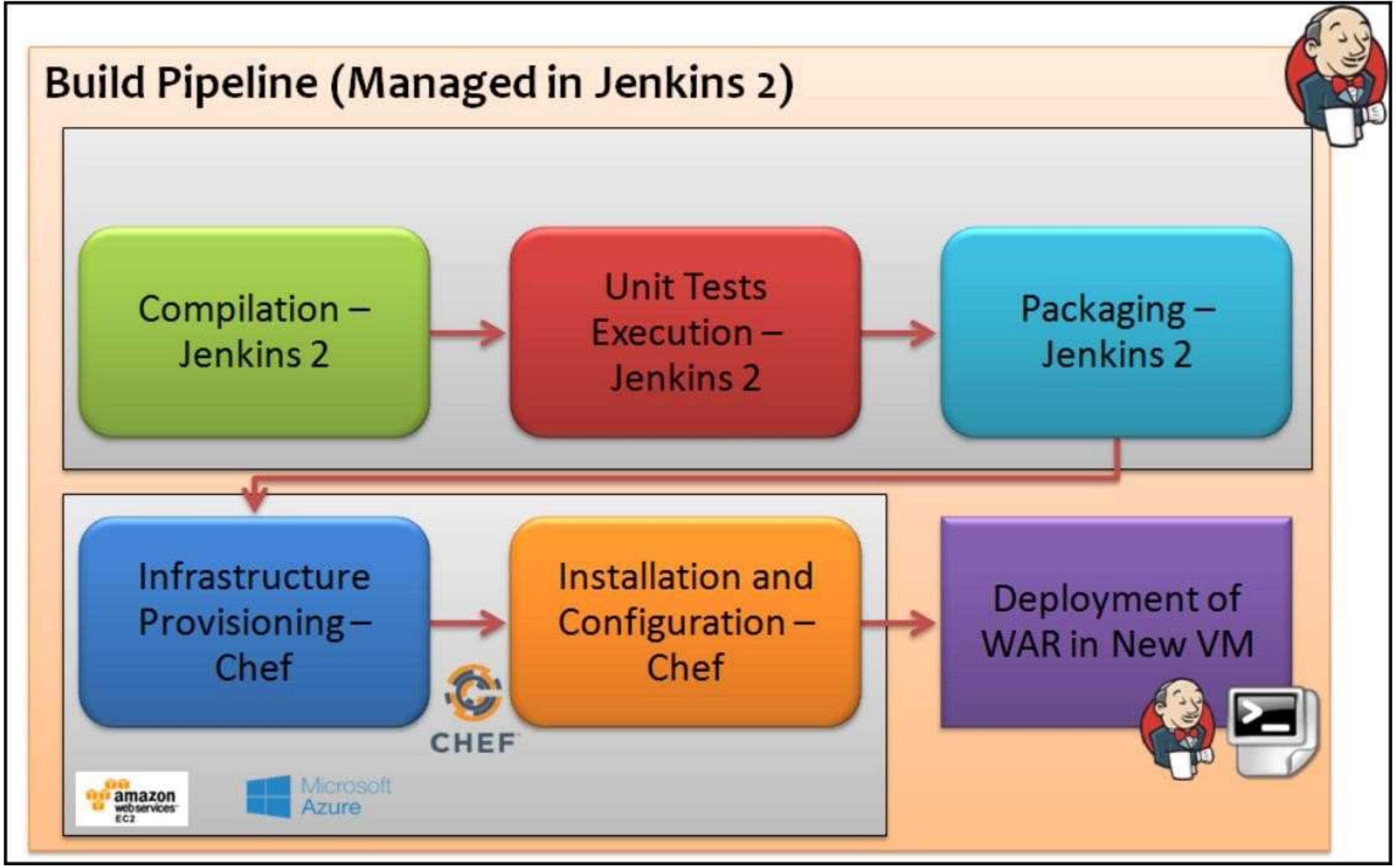

• Response Times
• Service Performance
• Database
• NoSQL
• Messaging
• Memory Caching
Application Monitoring
Platform Monitoring
Log Monitoring
Infrastructure Monitoring
• Log Analysis
• Server
• Storage
• Network

Continuous feedback

- Continuous feedback is the last important component in the DevOps culture and provides a means of improvement and innovation.



End-to-end orchestration: Jenkins plugins


Build Pipeline (Managed in Jenkins 2)
Compilation – Jenkins 2
Unit Tests Execution – Jenkins 2
Packaging – Jenkins 2
Infrastructure Provisioning – Chef
Installation and Configuration – Chef
Deployment of WAR in New VM
CHEF
amazon webservices EC2
Microsoft Azure

Thank you
for
listening!