DEVOPS FUNDAMENTALS

AGENDA

Day-1

- **Introduction to DevOps**
  - What's DevOps
  - DevOps - Agile relation
  - DevOps for me / my team ?
  - DevOps challenges
  - DevOps Benefits
  - Best practices
- **DevOps Business Scenarios**
  - Continuous Integration and Delivery
  - Continuous Deployment
- **Implementing Continuous Integration - Jenkins**
  - CI in Development environment
  - GIT (Version Control) + Maven (Automated App Build)
- **Continuous Delivery - Jenkins**
  - Automated App. Build + Testing
  - JIRA (Defect Tracking) and Artifactory (Binary Repository)

AGENDA

…CONT

Day-2

- **Continuous Deployment**
  - Ansible as deployment tool
- **Configuration Management using Ansible**
  - Ansible
- **Containerization Using Docker**
  - Docker
  - Docker-compose

Day-3

- **Kubernetes**
  - Kubernetes cluster
  - Cluster components
  - API Objects
  - Deploy application to Kubernetes cluster
- **Application monitoring**
  - Prometheus
  - Grafana
- **Infra Provisioning**
  - Terraform

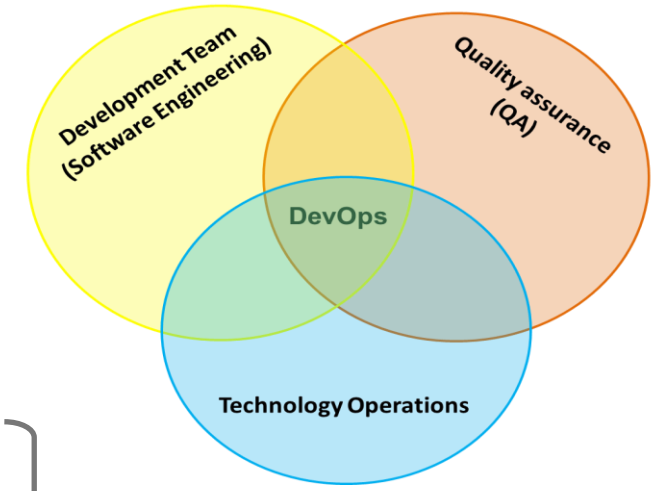
WHAT'S DEVOPS?

- DevOps is a software development and delivery process that emphasizes communication and collaboration between *Product Development Team, QA Team, Operations Team* and *Business owners* to increase organization's ability to deliver application and services at high velocity.
- *This speed enables organizations to better serve their customers and compete more effectively in the market.*

- **DevOps** building blocks:

  - Code
  - Build
  - Test

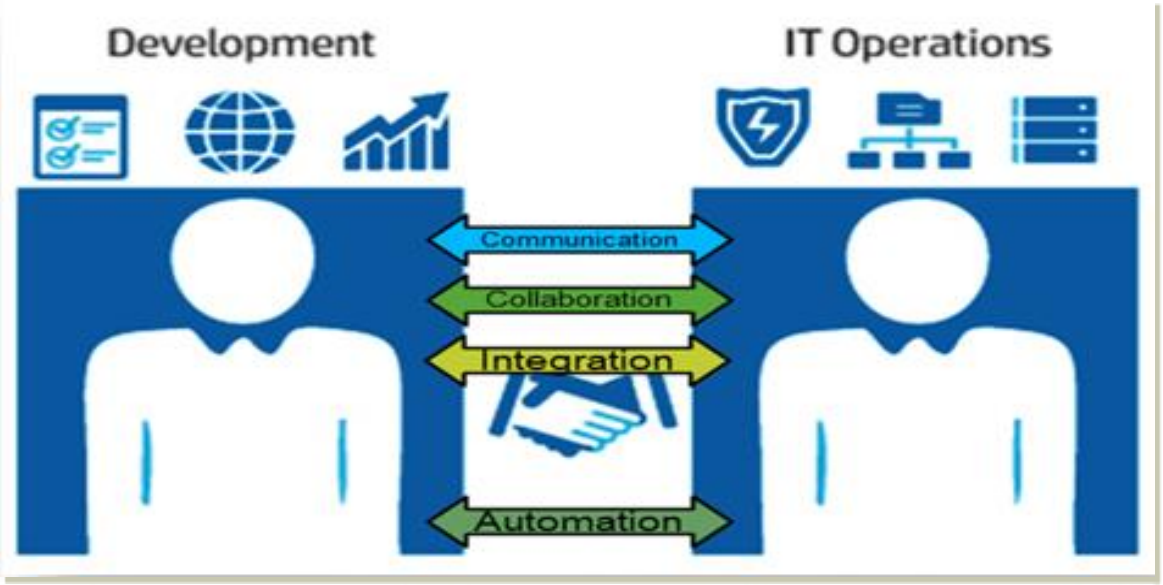
  } Development and QA Team



  - Packaging
  - Release management
  - Configuration management
  - Application and Infrastructure monitoring

  } Operations Team

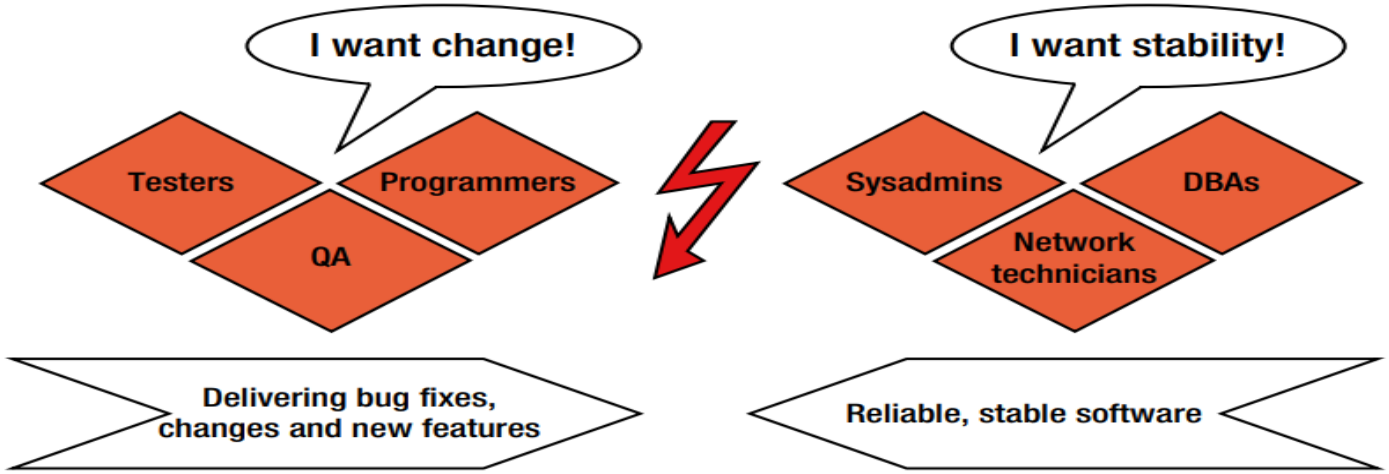
DEVOPS CULTURE



DevOps is more than just a tool or a process change; it inherently requires an organizational culture shift. This cultural change is especially difficult, because of the conflicting nature of departmental roles:

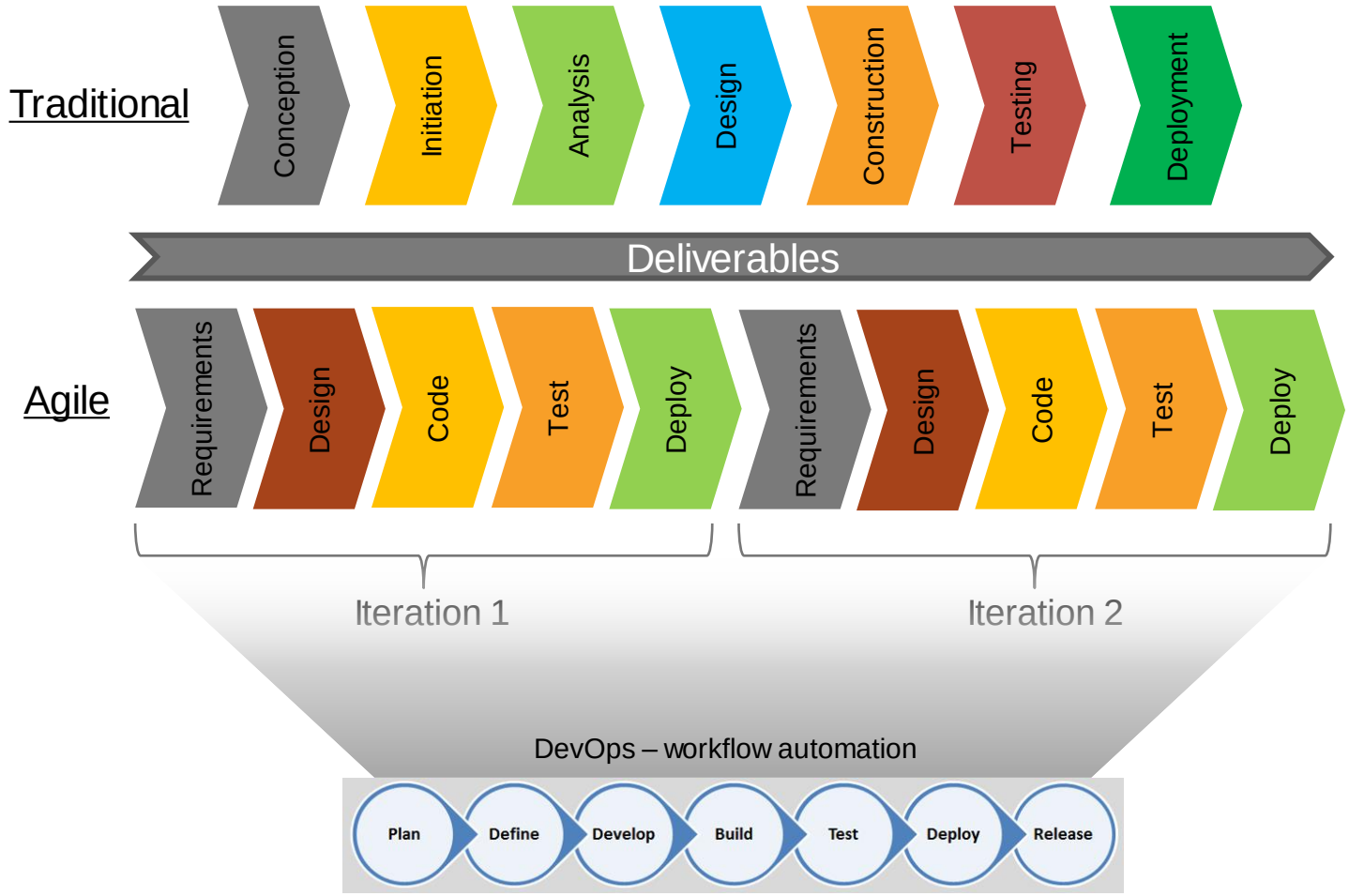
Operations — seeks organizational stability
Developers — seek change
Testers — seek risk reduction

DEVELOPMENT V/S OPERATIONS



- **Developers want change**
- **Deliver bug-fixes, changes and new features.**
  - The main task of the development team is to fulfil the customer's requirements, test the solution, and provide software updates in quick succession. New features that have been implemented and tested by the developers add potential value for the customer.
  - On one hand, the development team wants change. On the other hand, the operations team is mainly interested in reliable and stable software environment. Every change forwarded by the development team can endanger the existing reliability and stability of production environment.

DEVELOPMENT METHODOLOGIES - COMPARISON



AGILE PRACTICE AND DEVOPS RELATION

AGILE METHODOLOGY

Agile Model

Agile method proposes incremental and iterative approach to software design.

The agile process is broken into individual models that designers work on.

The customer has early and frequent opportunities to look at the product and make decision and changes to the project.

Agile model is considered unstructured compared to the Traditional (waterfall) model.

Small projects can be implemented very quickly. For large projects, it is difficult to estimate the development time.

Error can be fixed in the middle of the project.

Development process is iterative, and the project is executed in short (2-4) weeks iterations. Planning is very less.

Documentation attends less priority than software development

Every iteration has its own testing phase. It allows implementing regression testing every time new functions or logic are released.

In agile testing, when an iteration ends, shippable features of the product is delivered to the customer. New features are usable right after shipment. It is useful when you have good contact with customers.

Testers and developers work together.

At the end of every sprint, user acceptance is performed.

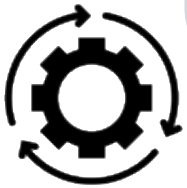
It requires close communication with developers and together analyse requirements and planning.

AGILE SCRUM FRAMEWORK AT A GLANCE

- working in iterations,
- build cross-functional teams,
- appoint a product owner and Scrum master,
- introducing regular meetings for iteration planning,
- daily status updates,
- sprint reviews

DEVOPS IN BUSINESS

DevOps Drivers



Market Competition

Ever-changing Business Needs

Quick to Market Requirement

- Tight delivery deadlines
- “The code works on my machine” – blame game
- Disconnect bet’n Development and Operations team.

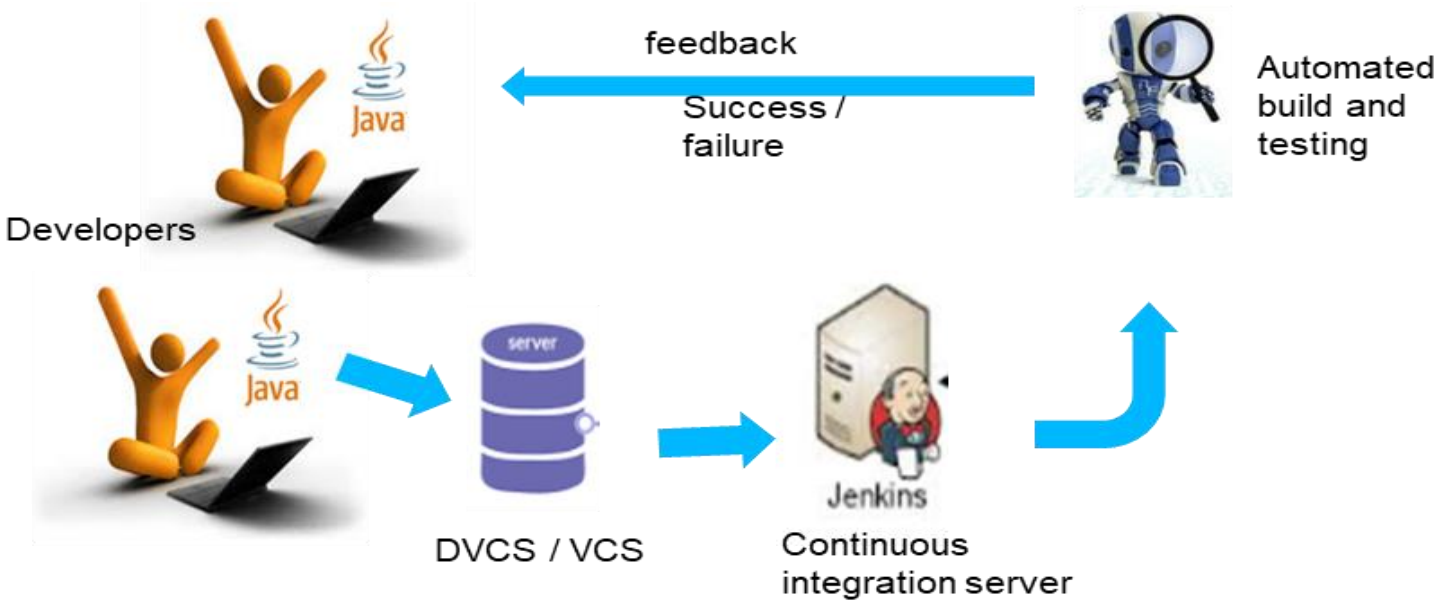
Conflict Scenarios

- Conflict during deployment
- Conflict after deployment
- Conflict about performance

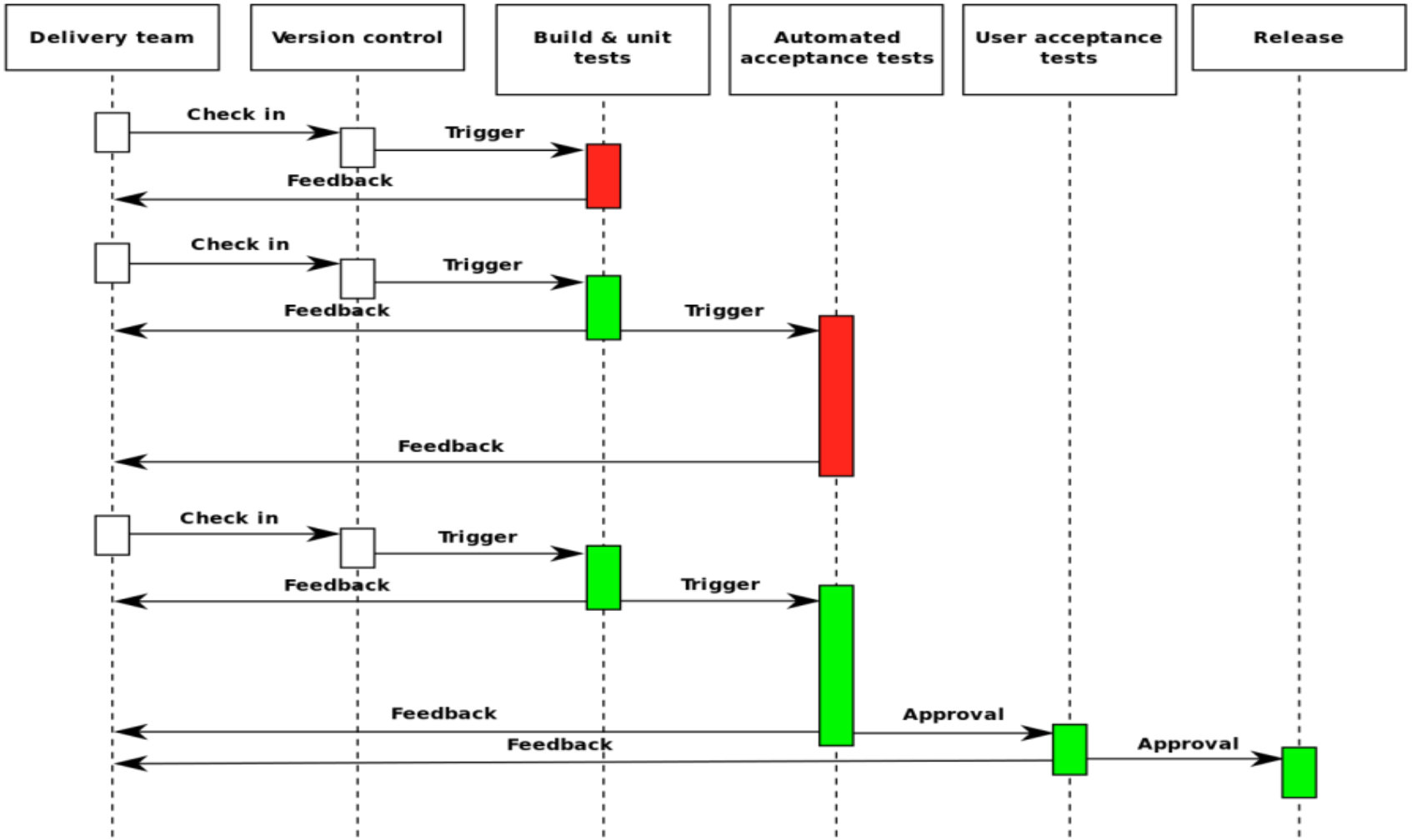
DEVOPS FOR DEV

Continuous Testing enabled with,

- Continuous integration.
  - Continuous integration is software development practice in which team members integrate their work frequently, leading multiple integrations per day. Each integration helps to reveals integrations errors in build success / failures as quickly as possible. This helps in significantly reducing integration problems and delivery timeline.



DEVOPS FOR QA – CONTI. DEL (CD)



Source: *Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation*

DEVOPS FOR IT OPERATIONS

Integrated environment provisioning

- Dynamic environment provisioning
- Containerized app deployment and Data Center Management

Continuous application deployment

- Single click deployment

Continuous monitoring –

- Performance monitoring
- System and application monitoring
- Log analysis

DEVOPS FOR BUSINESS OWNERS

Quick to Market

- Agility

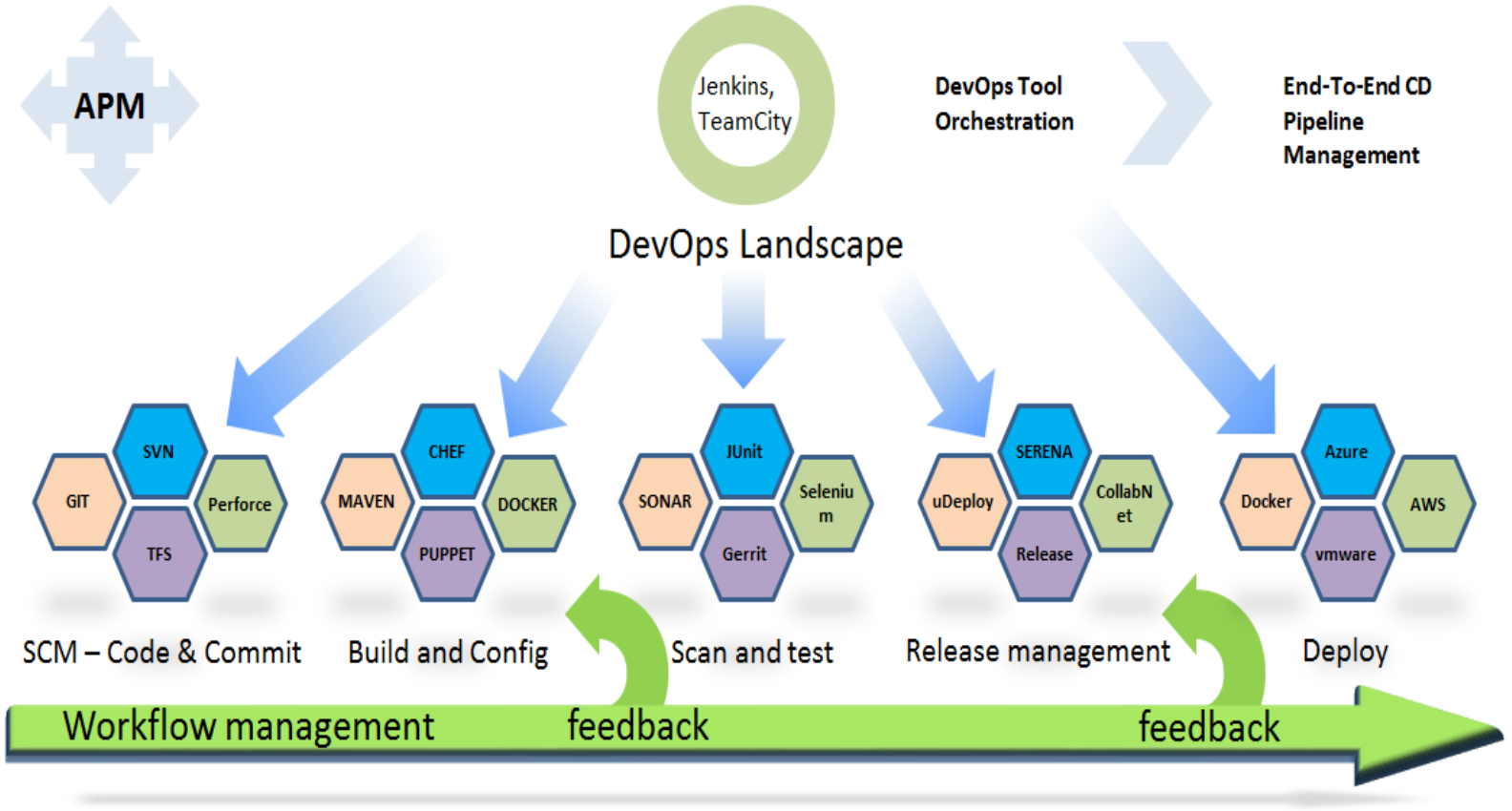
Environment stability

- Fast recovery
- Fully automated deployments

Customer satisfaction

- Improvement in product quality
- Quick turn around time

DEVOPS LANDSCAPE



CHALLENGES IN IMPLEMENTING DEVOPS!

Establishing DevOps Culture

Implementing Change in Application Development Environment.

Environment Upgradation (standardization)

Application Complexity

Budget

Availability of Skillset

BEST PRACTICES IN DEVOPS

Active partnership and close coordination among the stake holders in establishing DevOps culture.

Implement DevOps in totality. Avoid partial implementation, can become a reason for failure.

Choose right tool for each phase in DevOps implementation.

Best Practices

Options of substituting a exiting tools should be taken solicitously. No Fancy ideas.

Give equal importance to log analysis, report generation and circulation.

Mindset to adapt to changes.

DEVOPS OPERATIONAL BENEFITS

Increased Agility

To enable instant change deployment

Improved Quality

To improve end user satisfaction

Improve Innovation

To increase innovation cycle

Reduced Outages

Less outages in production (about 80% outages are change related)

THANK YOU