Automation Factory

Introduction to DevOps

Ganesh Palnitkar

www.automationfactory.in

Agenda

Automation Factory

- What’s DevOps
- DevOps - Agile relation
- DevOps for me / my team ?
- DevOps challenges
- DevOps Benefits
- Best practices

Automation Factory

What’s DevOps

What’s DevOps?

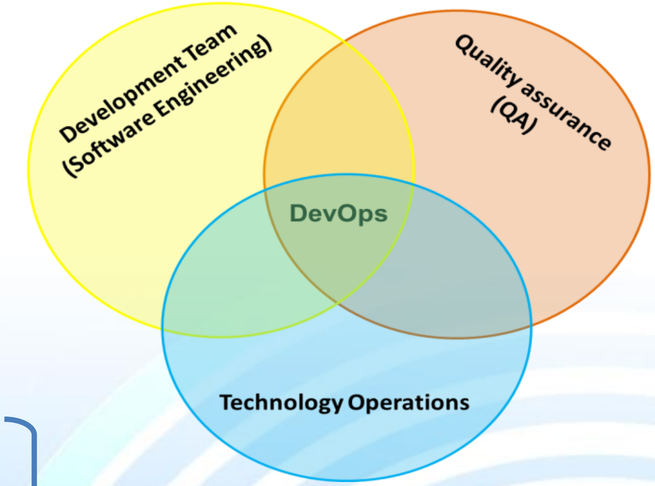
- DevOps is a software development and delivery process that emphasizes communication and collaboration between *Product Development Team, QA Team, Operations Team* and *Business owners to increase organization's ability to deliver application and services at high velocity.*

- *This speed enables organizations to better serve their customers and compete more effectively in the market.*

- **DevOps** building blocks:

  - Code
  - Build
  - Test

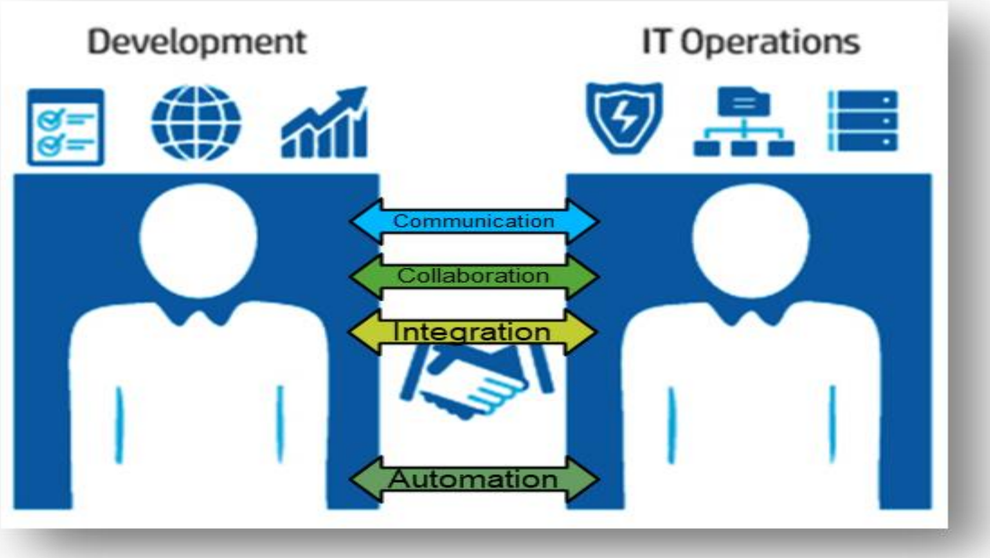
  Development and QA Team



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

  Operations Team

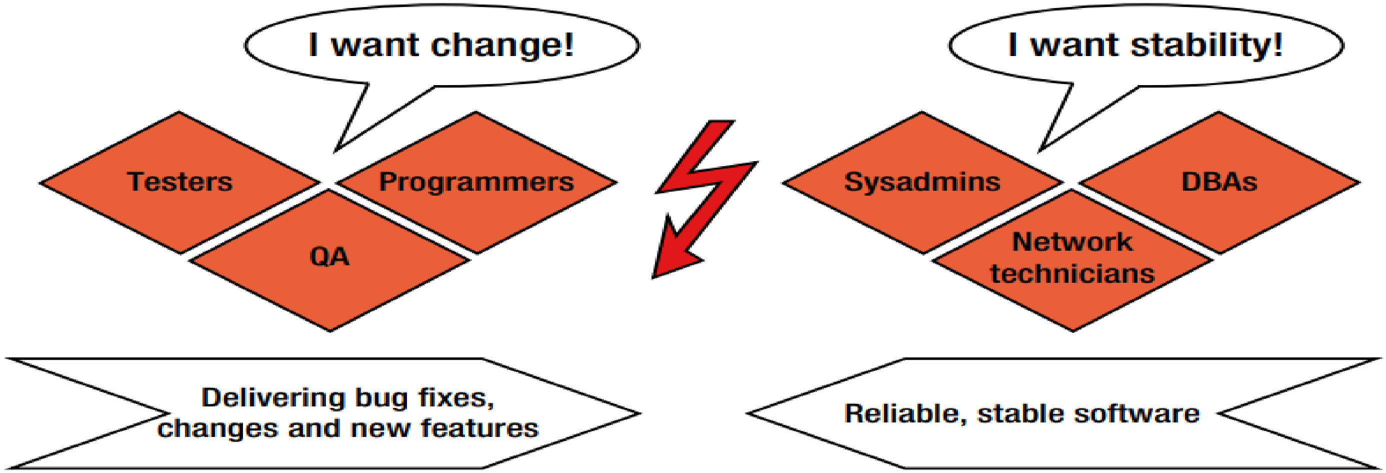
DevOps Culture



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

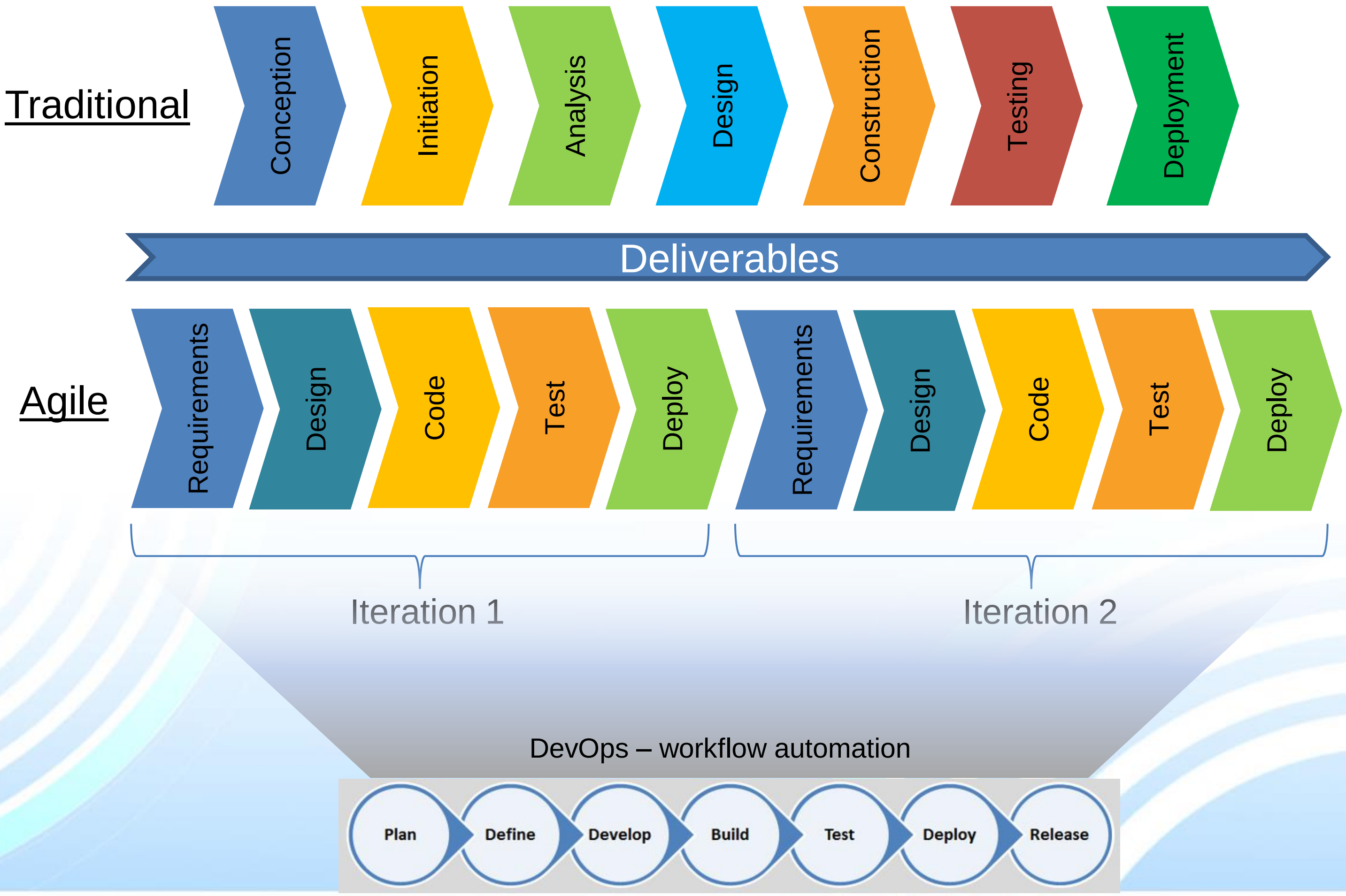
Development V/s Operations



- **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

Automation Factory



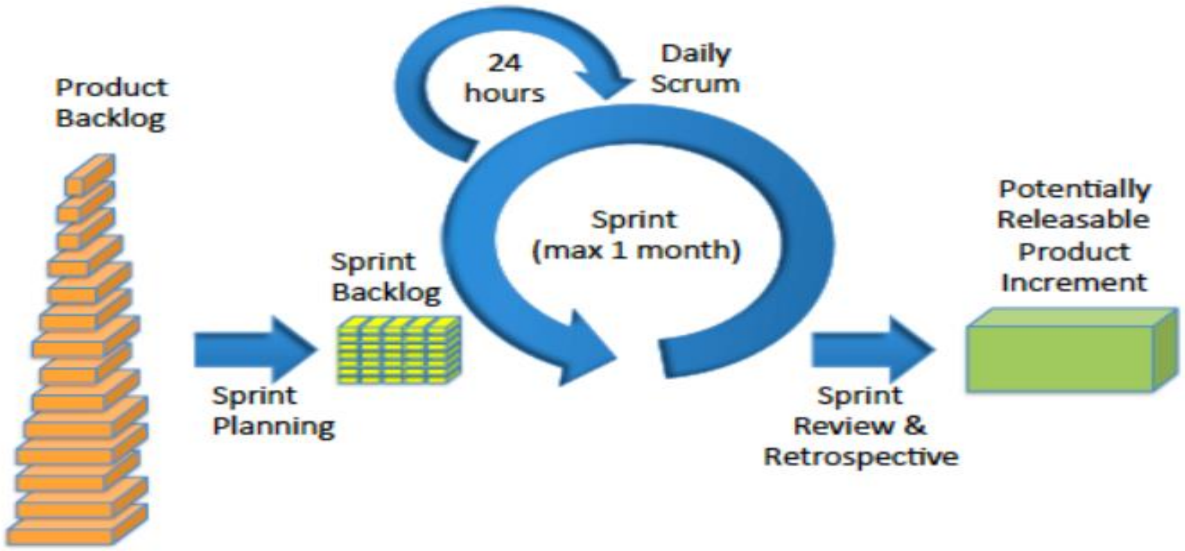
What’s Agile Practice

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.

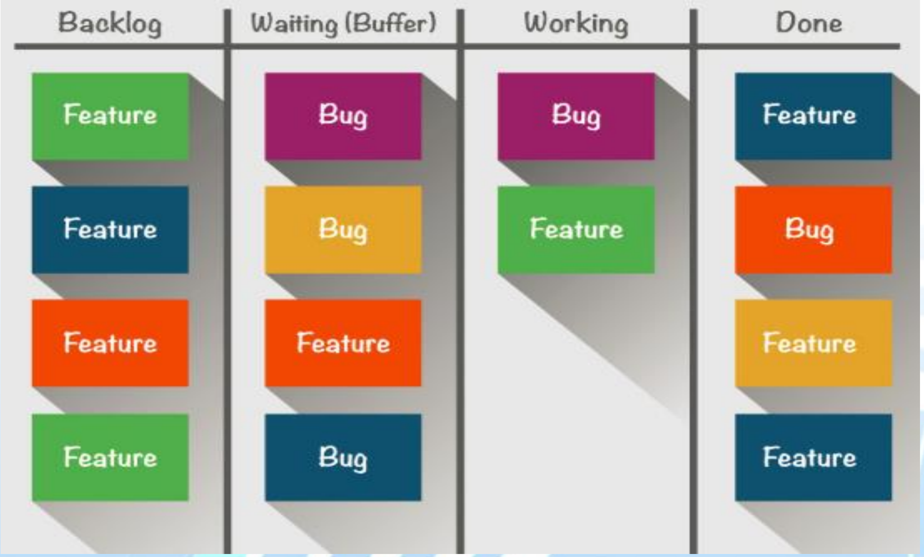
Agile Scrum Framework at a glance



- 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

Kanban

- The Kanban methodology is way less structured than Scrum.
- One can apply Kanban principles to any process that's already running.
- Kanban board acts as the charter. Board has states as columns, which every work item passes through – from left to right.
  - *Backlog,*
  - *In-progress,*
  - *Ready for release,*
  - *Done / Completed.*



- Daily stand-up meetings

Automation Factory

Making Agile successful

DevOps in Business

DevOps Drivers:

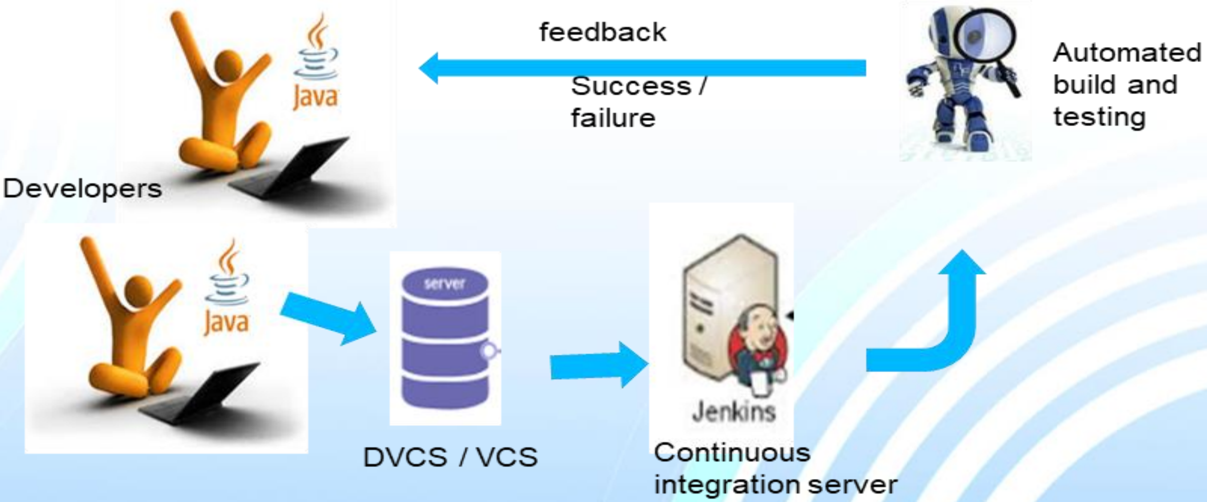
- Market competition
- Ever-changing business needs
- Quick to market requirement
    - Tight delivery deadlines
    - “The code works on my machine” – blame game
    - Disconnect bet’n Development and Operations team.
- Conflict Scenarios
    - Conflict during deployment
    - Conflict after deployment
    - Conflict about performance
- Advantages of agile processes like Scrum, Kanban are often nullified because of the obstacles to collaboration, processes, and tools that are build up in front of operations. Because of this achieving delivery timelines for a sprint becomes challenging.

Scenario

	Development Team	QA Team	Operations (Environment and deployment)
Team Size	5	3	3
Delivery timeline	12 months		
Development methodology	Agile		
Scenario-1	New release for deployment	→	Failed deployment
Scenario-2	New release deployed		Performance issue in production
Scenario-3	- New release - Bug fixed sent for testing	→ Failing Tests	Product failed in prod. Code bug

DevOps for QA

- 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.



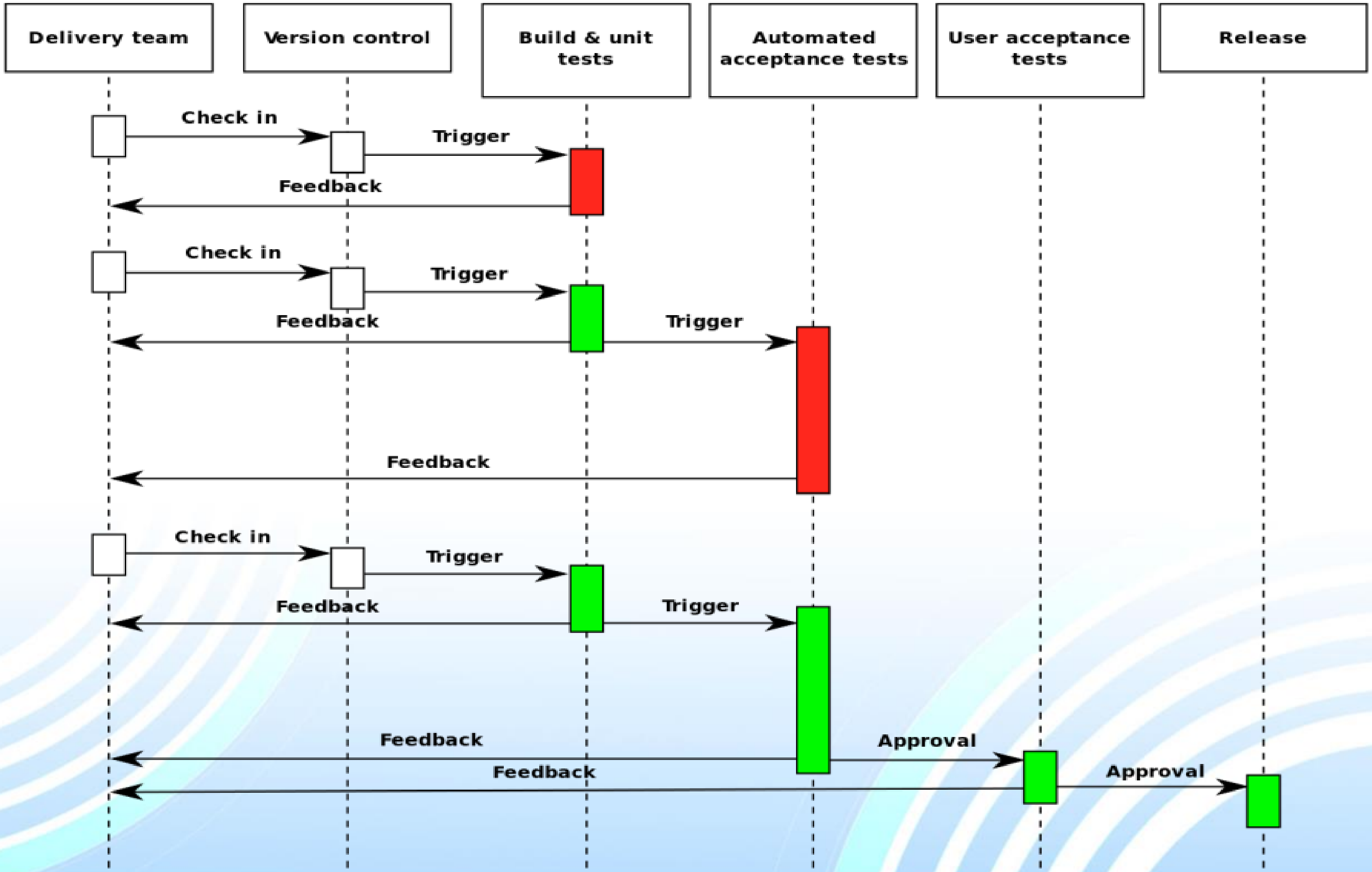
DevOps for IT Operations

- 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

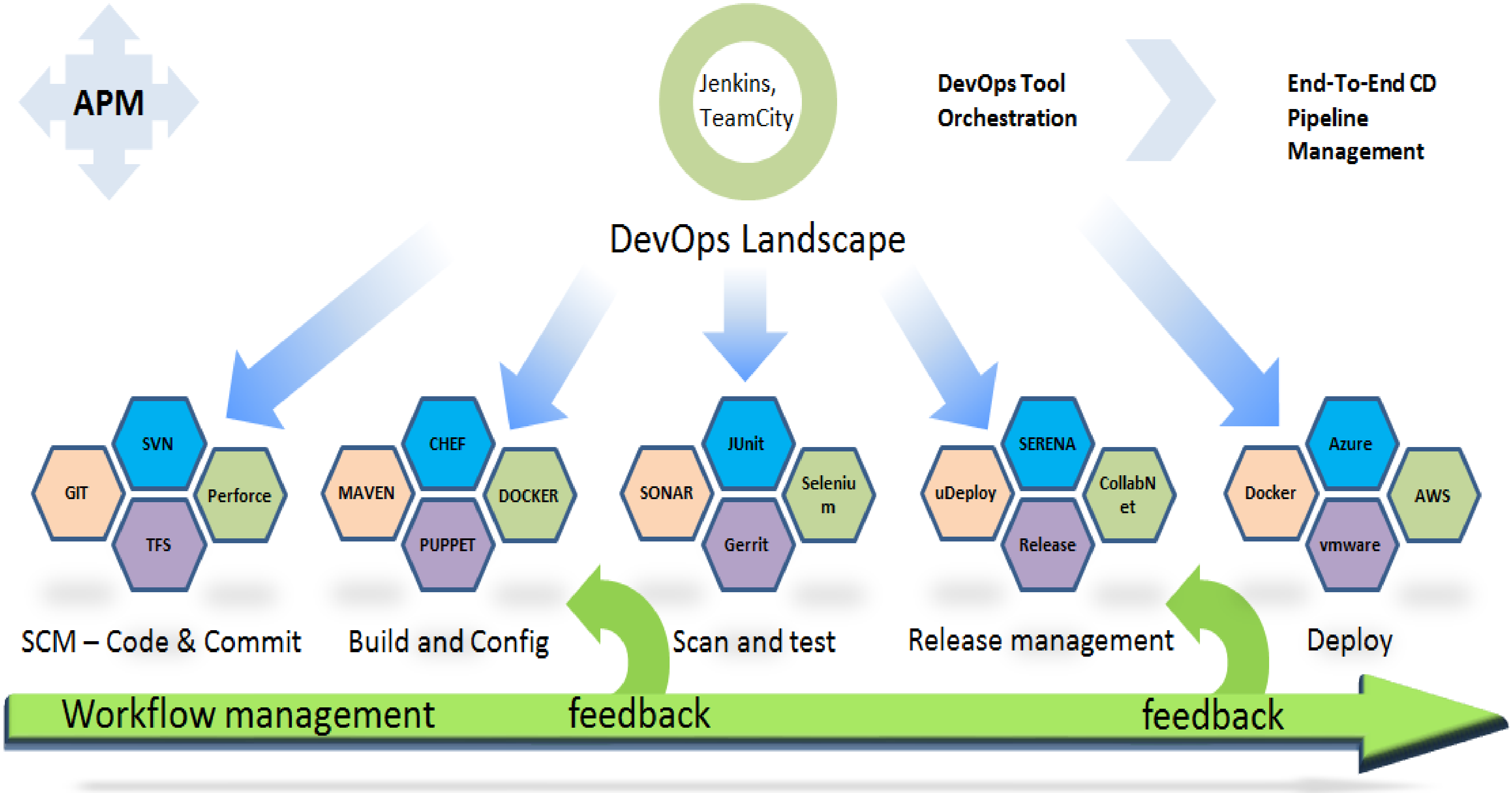
- Quick to Market
  - Agility
- Environment stability
  - Fast recovery
  - Fully automated deployments
- Customer satisfaction
  - Improvement in product quality
  - Quick turn around time

Continuous delivery pipeline



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

DevOps landscape



Common goals of an enterprise DevOps practice

- Increased deployment frequency
- Reduced lead time for changes
- Faster recovery when problems occur
- More robust and better integrated security
- A “shift left” in quality – start testing in an earlier phase.
- Fast feedback loops and effective communication between teams and departments

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.
- 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

Automation Factory

Increased Agility:

To enable instant change deployment

Improved Quality:

To improve end user satisfaction

Improve Innovation

To increase innovation cycle

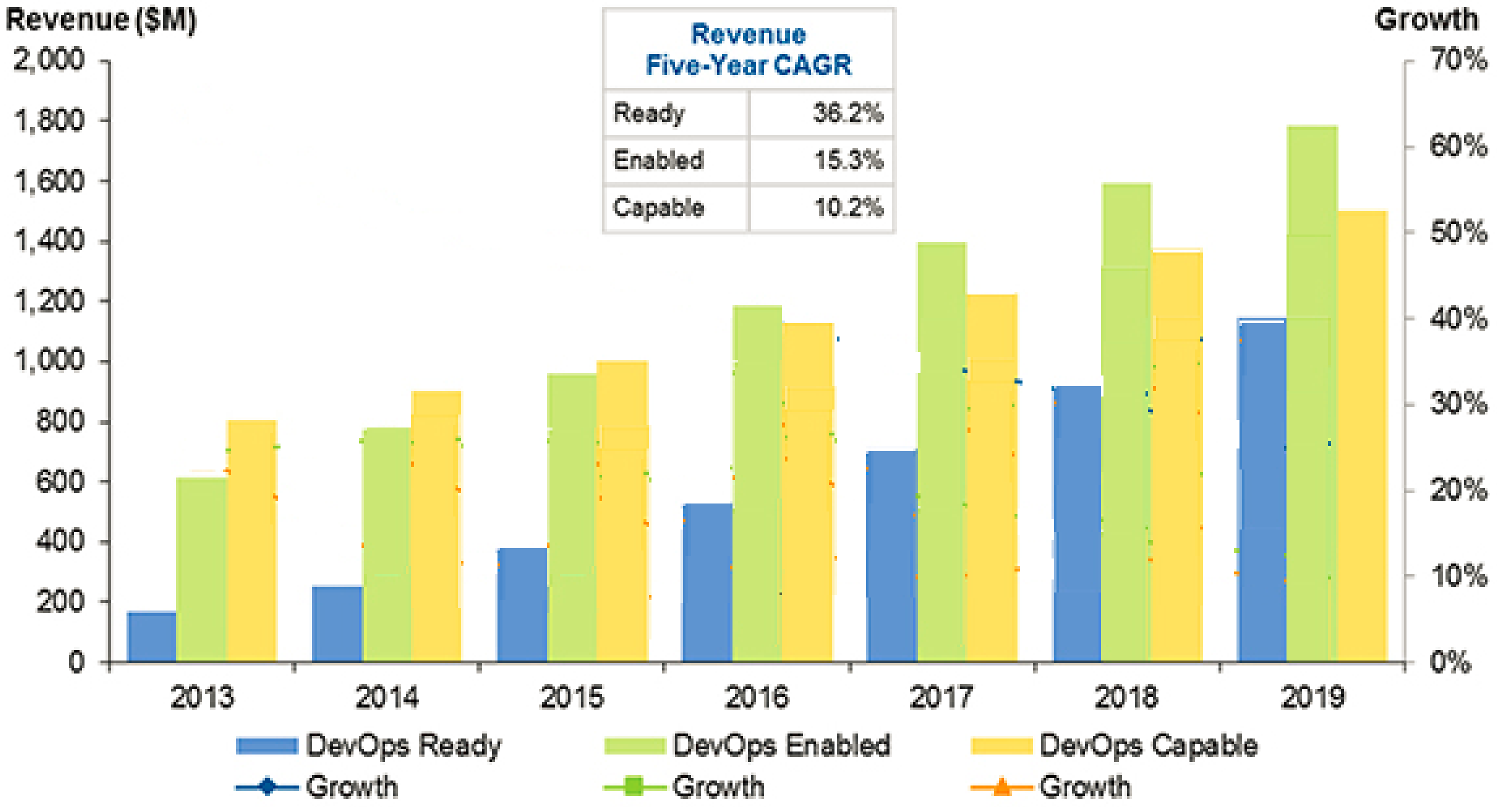
Reduced Outages

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

DevOps - Career path

- ❑ DevOps Engineer : supporting DevOps implementation and tools.
- ❑ DevOps Architect : designing DevOps solutions for product development tracks.
- ❑ DevOps terminology came into practice around 2009 and since 2012-13 industry has seen fast growth in number of companies adopting DevOps.
- ❑ According to Gartner, organizations around the world are increasingly adopting the DevOps culture and by the end of 2016, 25 percent of top global 2000 organizations would have adopted DevOps as a mainstream strategy.
- ❑ Forrester estimates that around 50 % of top 100 companies would adopt DevOps practice by 2017.
- ❑ DevOps certifications are still evolving. Certifying bodies like, PeopleCert and Exin and actively introducing certification tracks which are updated every quarter.

DevOps Awareness Pattern in IT Industry



Revenue Five-Year CAGR	
Ready	36.2%
Enabled	15.3%
Capable	10.2%

Source: Gartner 2016

Automation Factory

Thank You