# DevOps

**In a software company oneday (Friday) evening.**

**An Email received by IT Team (Operations Team) from Development Manager.**

**“ Here is the attached ‘Tar file’ please deploy in the Test Environment”**

# Senario

**To deploy this they have lot of questions in IT Team Members.**

-> In which system has to install?

-> what are the dependencies?

-> what are the database requirements?

-> what are the security requirements?

-> who can access?

IT Team members are end up with Lots Of Questions?

# Next Monday Manager and IT Team having mud fight

IT Team Member

Manager

IT Team Member

# Blaming

**Development Team**

They don’t know, what to do except asking questions

They don’t know which is server and client.

They always request in last minute.

We are missing deadlines because of IT Team.

Because of them we got so much downtime on servers.

**Operations Team**

**Mud fight is a reason for organization Failures.**

IT Team Member

Manager

IT Team Member

**This is why DevOps Comes in**

# Let us see individual Problems

## Development Team

- Developers Team
- Build Team
- Testing Team

# What each team do individually?

## Developers Team

Developers are responsible for coding the code using

Java, .Net, Python…..

# What each team do individually?

## Build Team

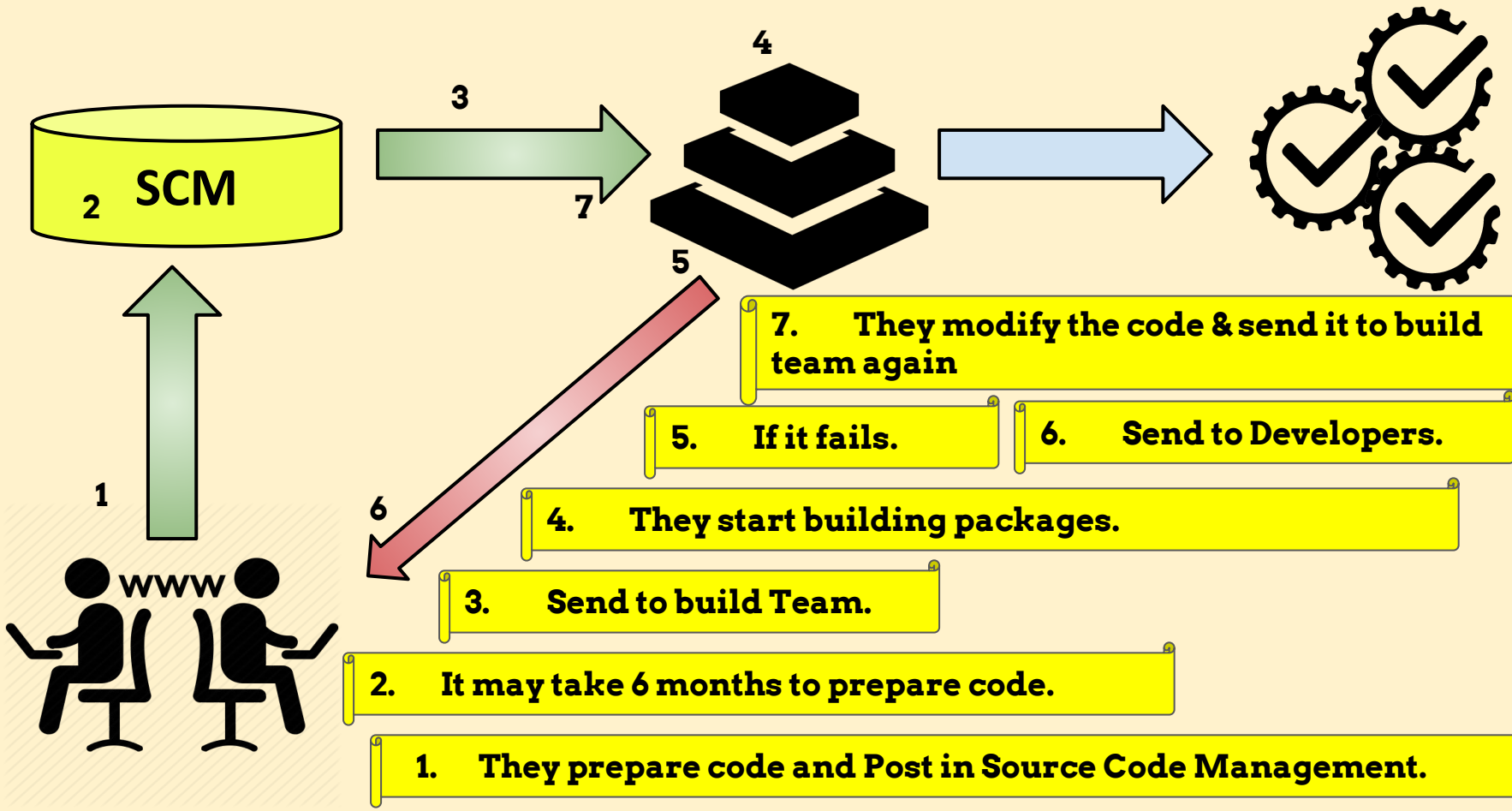
Build team members are responsible for build the code made by developers using Ant, Maven...

# What each team do individually?

## Testing Team

Testing team members are responsible for Testing the build code made by build team using
Selenium, TestingWhiz, QTP (Quick Test Professional)

# How Development Team works?




7. They modify the code & send it to build team again

5. If it fails.

6. Send to Developers.

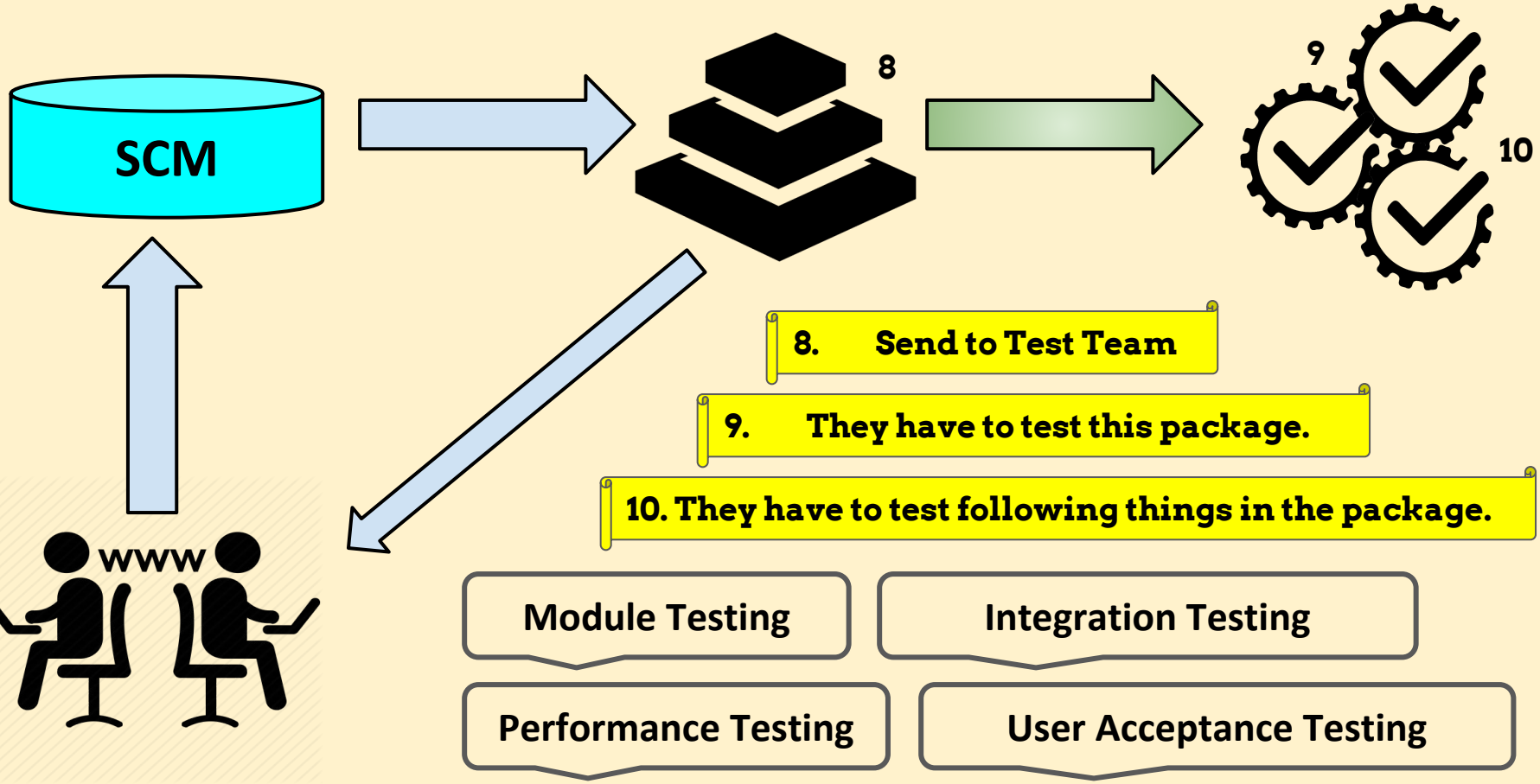
4. They start building packages.

3. Send to build Team.

2. It may take 6 months to prepare code.

1. They prepare code and Post in Source Code Management.

# How Development Team works?



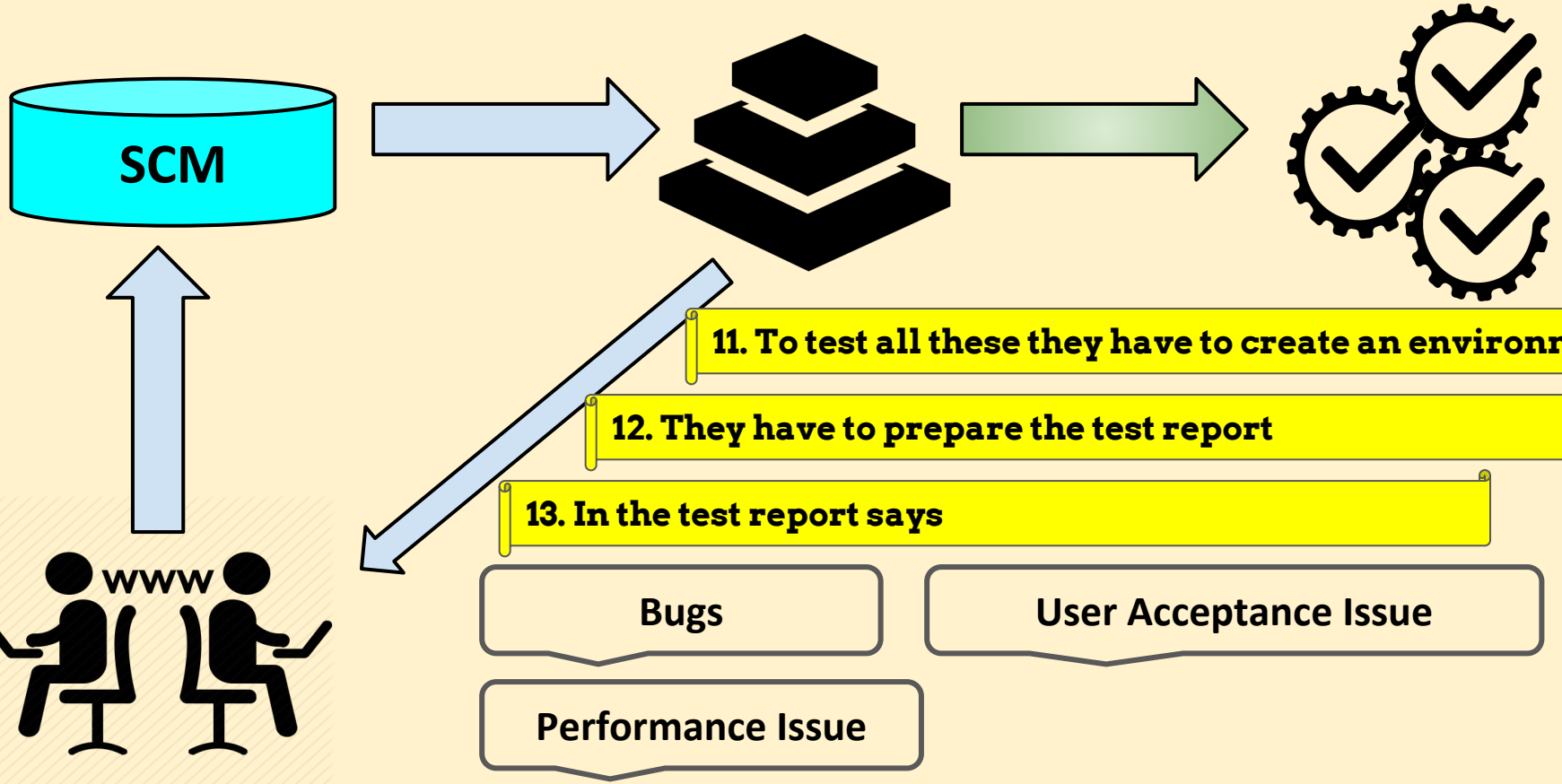

8. Send to Test Team

9. They have to test this package.

10. They have to test following things in the package.

- Module Testing
- Integration Testing
- Performance Testing
- User Acceptance Testing

# How Development Team works?



# How Development Team Works?

14. Again they send to Development team (Developers)

They have to rework on Code.

15. If requirements are **not changed** then the product will be Ready.

Developers → Build → Testing

16. If requirements are **changed** then the product will go back to Developers.

Developers → Build → Testing

## To avoid these issues?

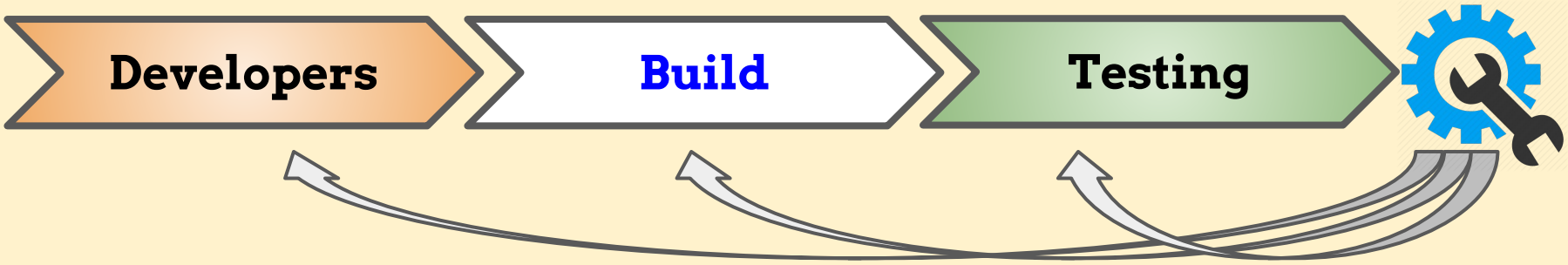
# Agile Model Comes Up

In this model they won't build application fully, they divided applications in many small modules.

If the requirement is change then modification of application is very easy by add patches and Removing bugs.

To avoid these issues?

# Agile Model Comes Up




But there is a still latency for build an application.

Let us see

# Operation Team (IT Team)

## Server Team

## Network Team

## Virtualization Team

## Storage Team

## Security Team

Let us see

# Operation Team (IT Team)

Size of Operation team 1-5% of Development Team

Let us see

# Responsibility of IT Team

## Server Team

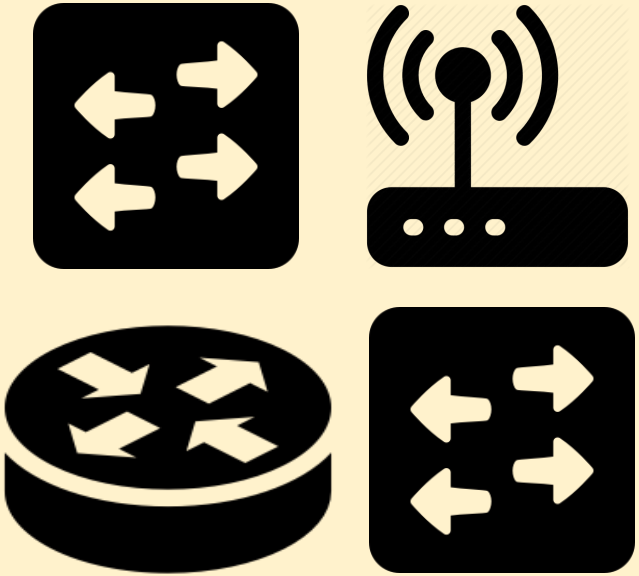
They are responsible for maintaining servers.

Let us see

# Responsibility of IT Team

## Network Team

They are responsible for maintaining Switches and Router.

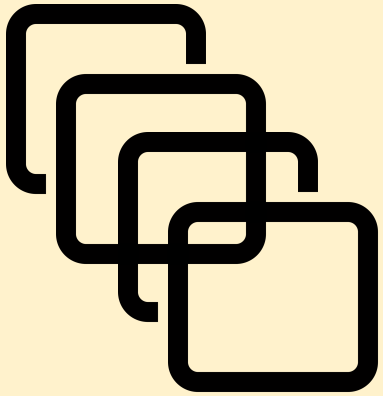


Let us see

# Responsibility of IT Team

## Virtualization Team

They are responsible for maintaining Virtual Machines and its Hypervisors.

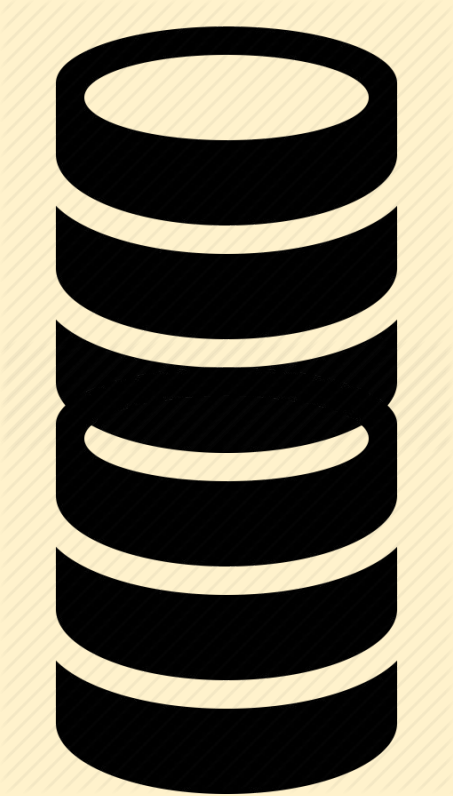


Let us see

# Responsibility of IT Team

## Storage Team

They are responsible for maintaining NAS box and Storage Servers up on networks.

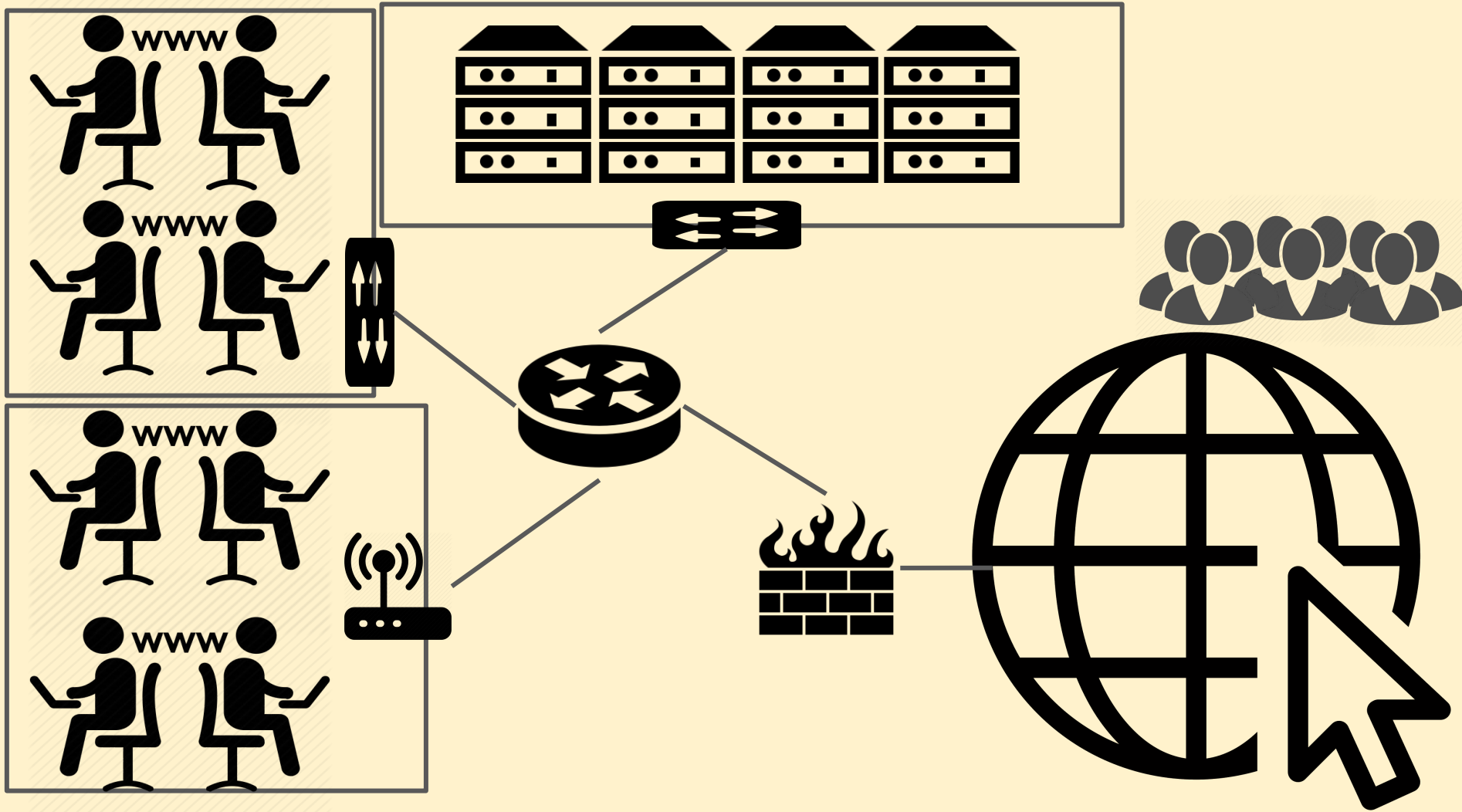


Let us see

# Responsibility of IT Team

## Security Team

They are responsible for maintaining Firewalls and providing Security to enterprises.


www
www
www
www

# Responsibility of IT Team

**To deploy an application they have to do:**

- **Servers, Network Connectivity, OS, patching, supporting softwares and other activities.**
- **To access application they have to open ports on Firewalls. (Internet/Intranet)**
- **Storage allocation to store all logs and Transactions.**

**To do all these tasks**

- **They have to maintain "Runbook" (Documentation). It must be very detailed. It should be repeatable.**

| | Development | Operations |
| --- | --- | --- |
| Build | They must have successful build | They must have stable build |
| Change | Requirements are changing frequently | Requirements are changing infrequently |
| Standard | Agile | ITIL |
| Target | Develop a new feature | Increase the server, network uptime. |

**Both team success metrics are different.**

## What is the solution?

**They both merged together work for an organization success.**

## How to make Organization Success?

**By implementing**

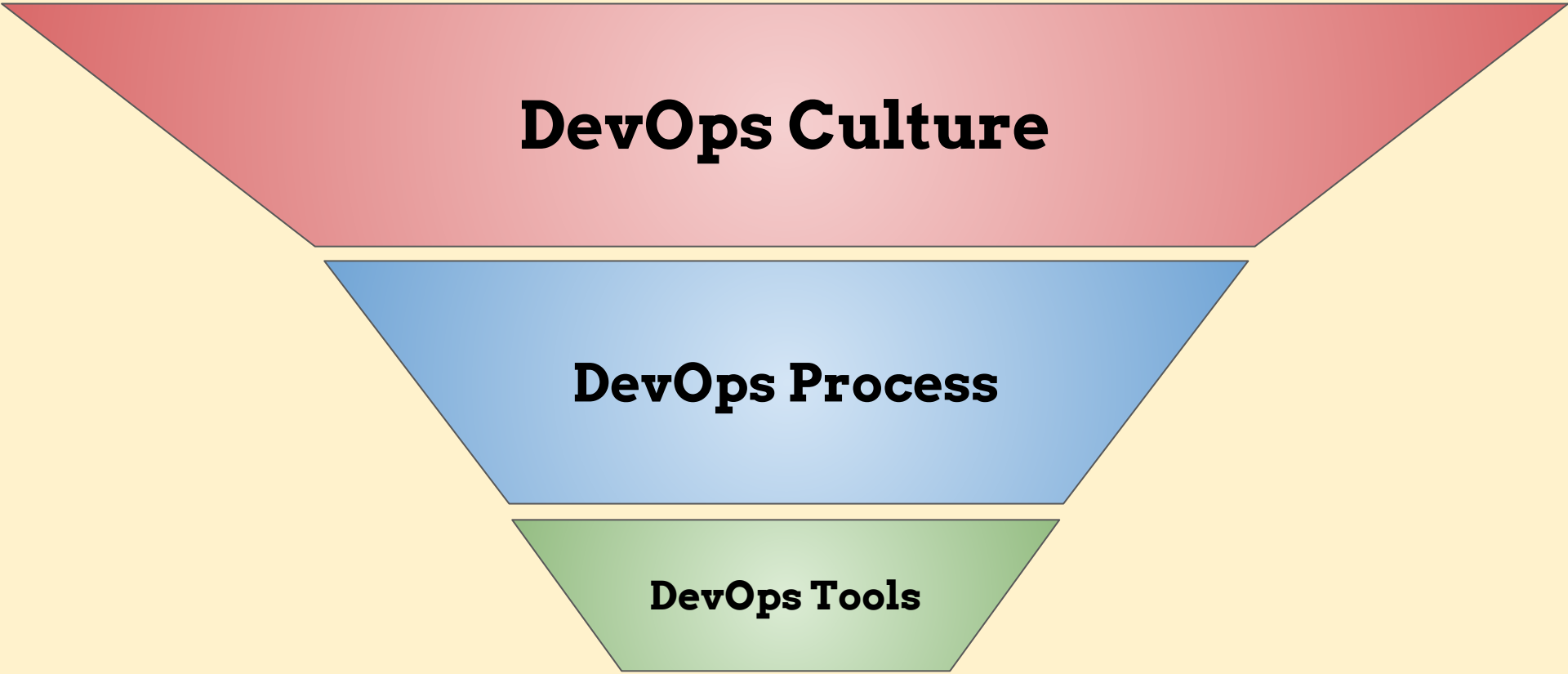
# What is DevOps

**DevOps is a software development methodology that combines software development with information technology operations.**

**The goal of DevOps is the shorten the systems development life cycle while also delivering features, fixes, and updates frequently in close alignment with business objectives.**

**The DevOps approach is to include automation and event monitoring at all steps of software build**

# Parts of DevOps



# Parts of DevOps

## DevOps Culture

### Communication

Both team have to communicate.

### Collaboration

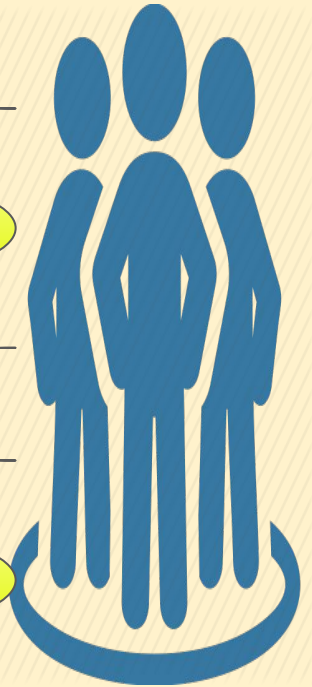
Both team have to collaborate.

### Integration

Ops team have to understand Dev.

### Automation

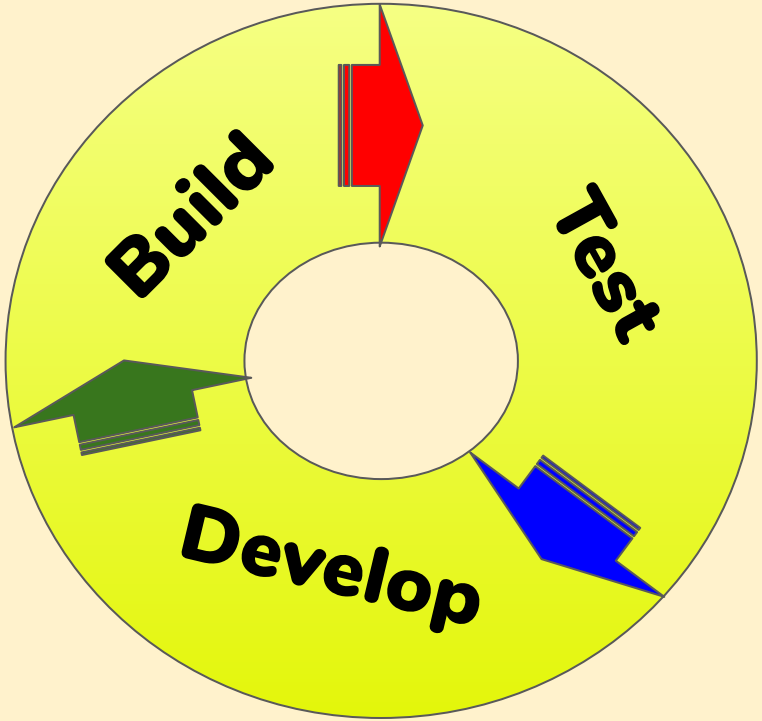
Develop and Automate things.



# Parts of DevOps

## DevOps Process

## Continuous Integration




Increase the productivity.

Target reachability is easy.

Application on Date.

# Parts of DevOps



## Configuration Management

It is a process of configure to all VM/PM by using single command.

## Continuous Deployment

Taking images of servers to redeploy.

## Monitoring

Continuous Monitoring Servers and other resources.

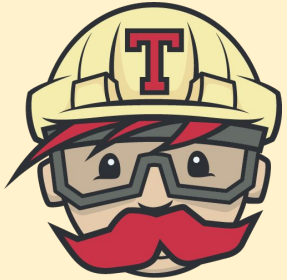
# Parts of DevOps

## DevOps Tools

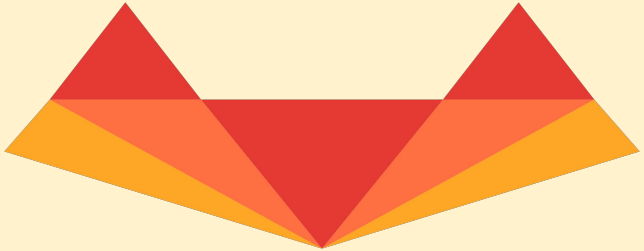

## Source Code Management


# Parts of DevOps

DevOps Tools




CONTINUOUS INTEGRATION

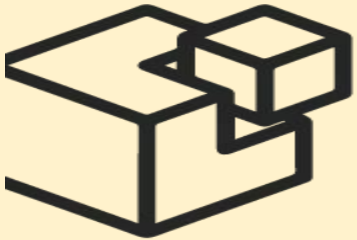


# Parts of DevOps

## DevOps Tools


CHEF™

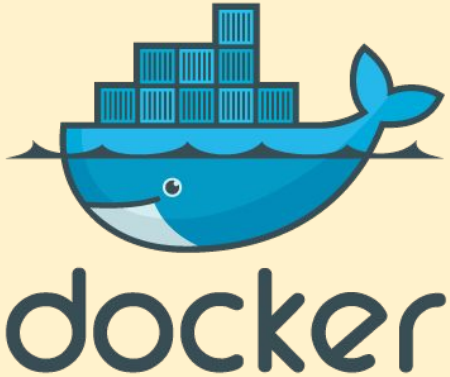
## CONFIGURATION MGMT



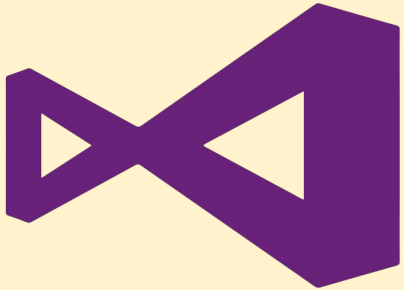
SALTSTACK

# Parts of DevOps

## DevOps Tools


## CONTINUOUS DEPLOYMENT



# Parts of DevOps

DevOps Tools

solarwinds

PRTG NETWORK MONITOR

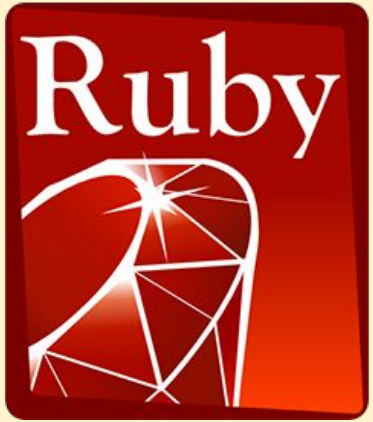
CONTINUOUS MONITORING

Nagios®

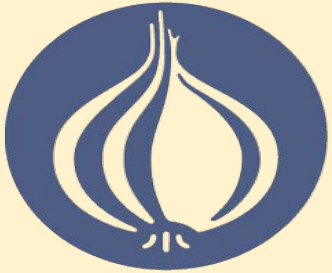
ZABBIX

# Parts of DevOps

## DevOps Tools



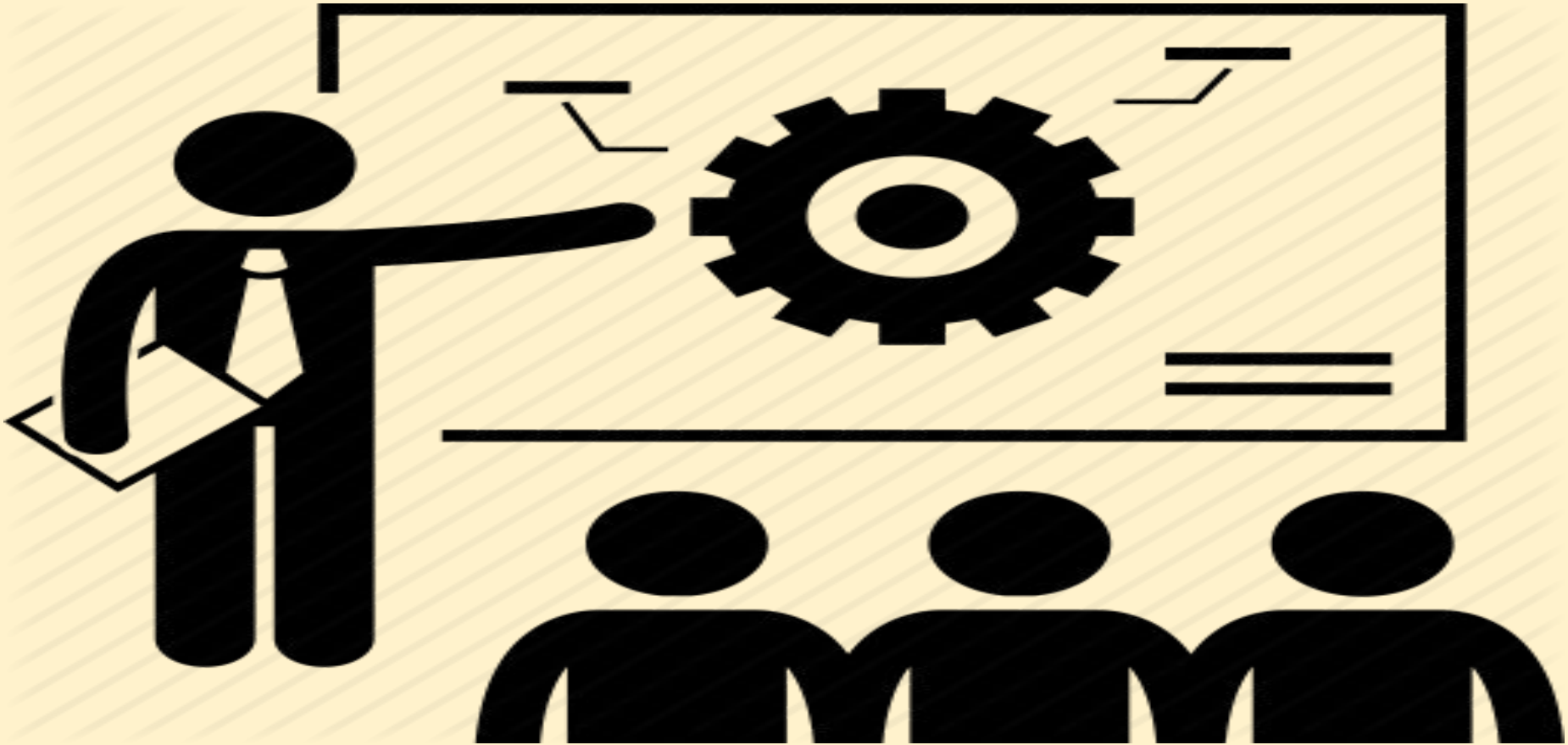

## AUTOMATION TOOLS



Perl

ANSIBLE

# Demo



# Who is DevOps Engineer?

A DevOps Engineer is an IT professional who works with software developers, system operators and other production IT staff to administre code releases.

A DevOps Engineer will work with development team staff to tackle the coding and scripting needed to connect elements of code, or software development kits.

# What are the Roles and Responsibilities ?

Able to perform system troubleshooting and problem solving across platform and application domains.

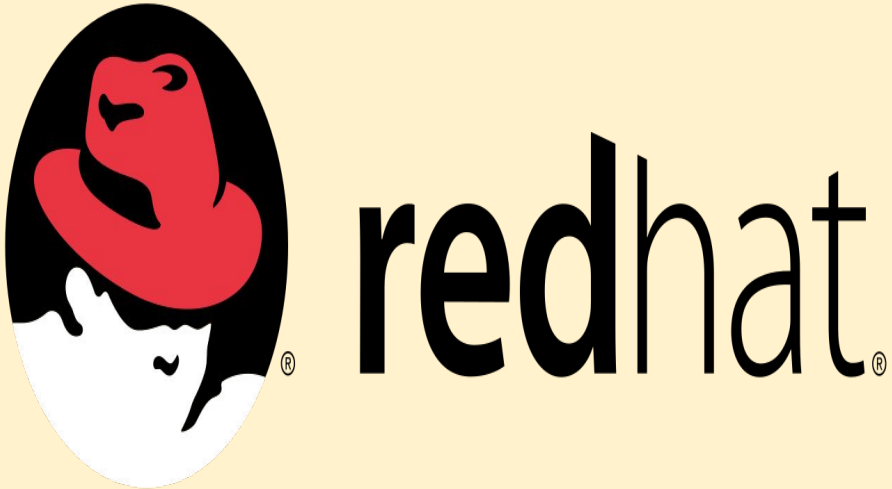
Manage project effectively through open standards-based platforms.

Improve quality and reduce development cost with collaboration.

Analyse, design and evaluate automation scripts and systems.

# DevOps Certifications






# DevOps Certifications


## AWS Certified DevOps Engineer

# DevOps Certifications


## Implementing Microsoft Azure DevOps Solutions

# DevOps Certifications

RHC of Expertise in Platform as a Service
RHC of Expertise in Containerized Application Development
RHC of Expertise in Ansible Automation
RHC of Expertise in Configuration Management
RHC of Expertise in Container Administration

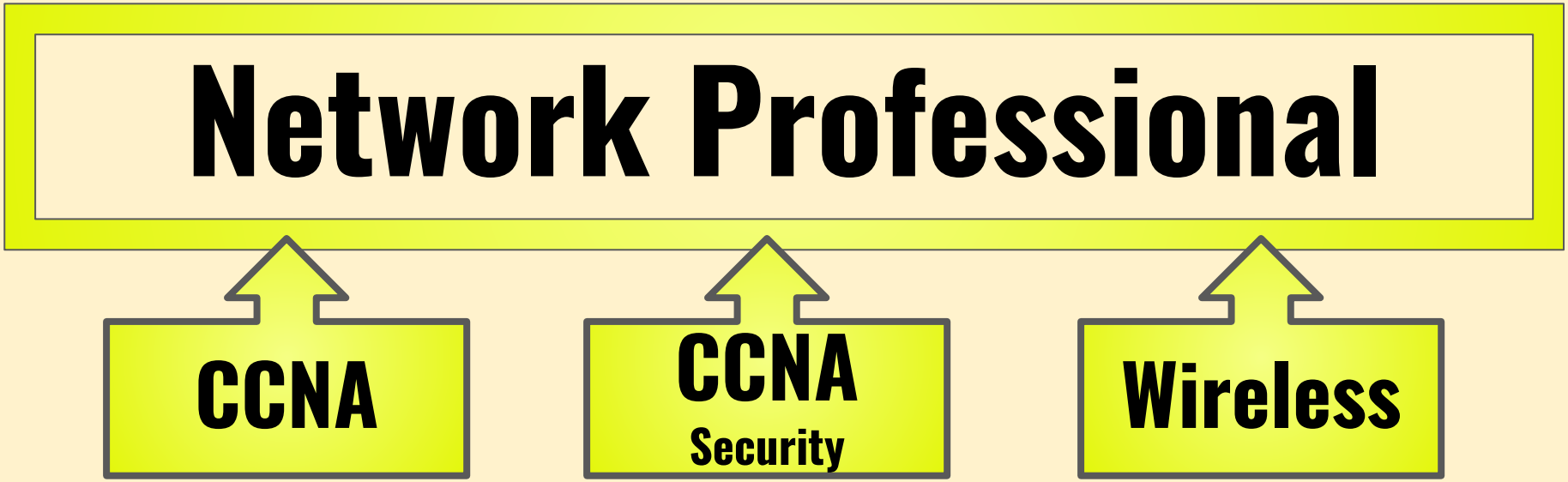
redhat®

# DevOps Certifications

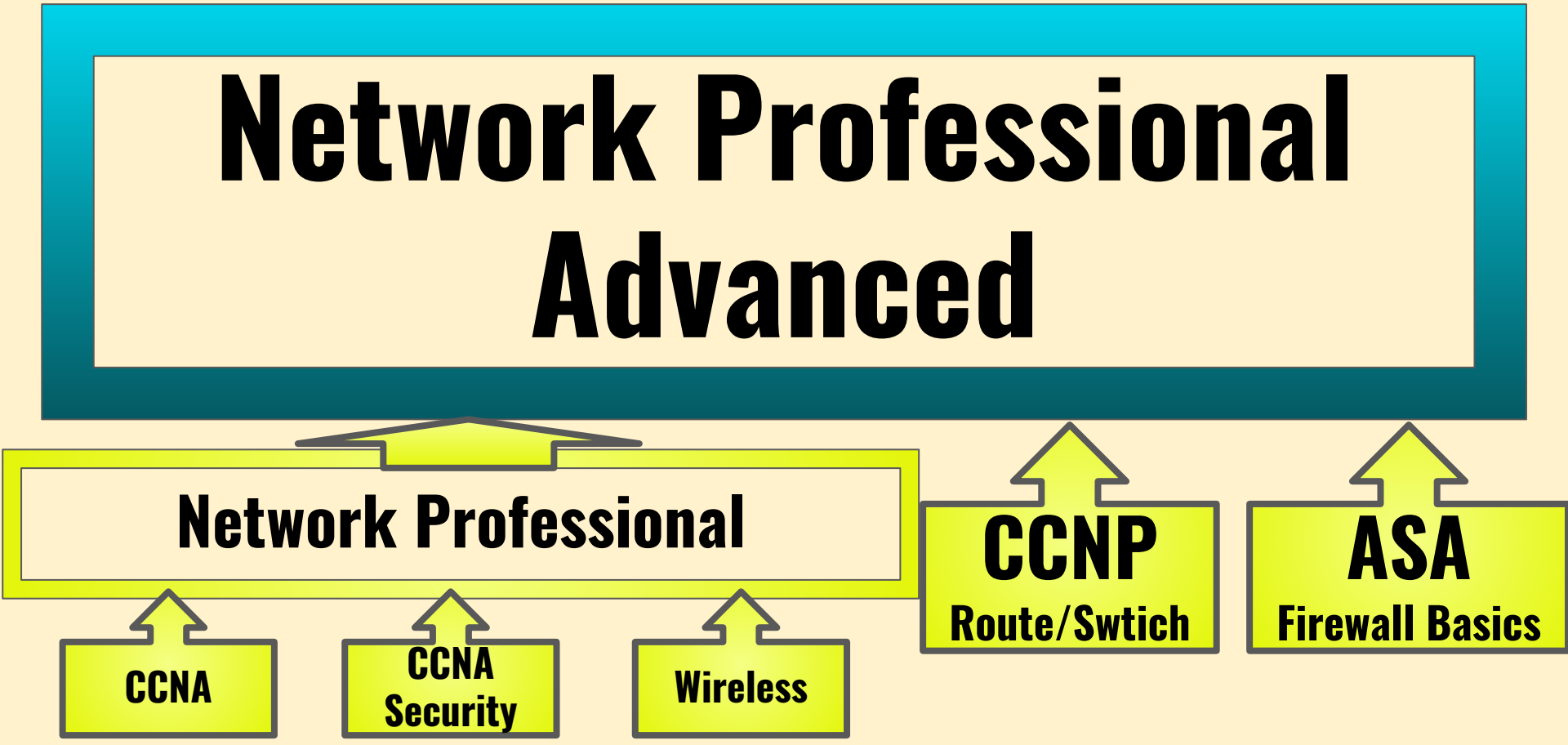
DevOps Foundation Certified
DevOps Test Engineering
DevOps Leader….

DEVOPS INSTITUTE

# New and Upgraded Courses



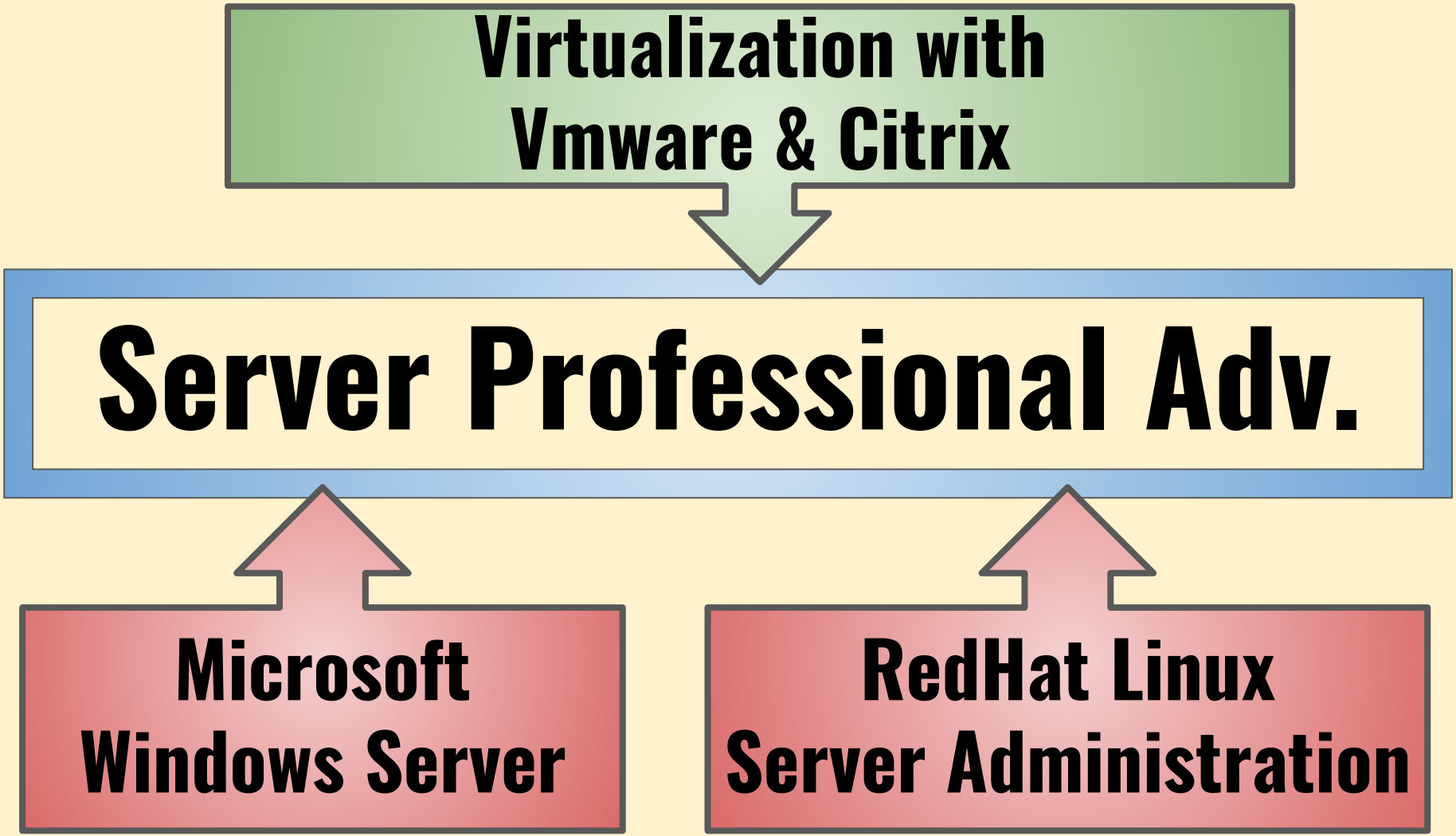
# New and Upgraded Courses



# New and Upgraded Courses

## Server Professional

- Microsoft Windows Server
- RedHat Linux Server Administration


Virtualization with
Vmware & Citrix
Server Professional Adv.
Microsoft
Windows Server
RedHat Linux
Server Administration

# Cloud Administration

Amazon Web Services

Microsoft Azure

# Thank You