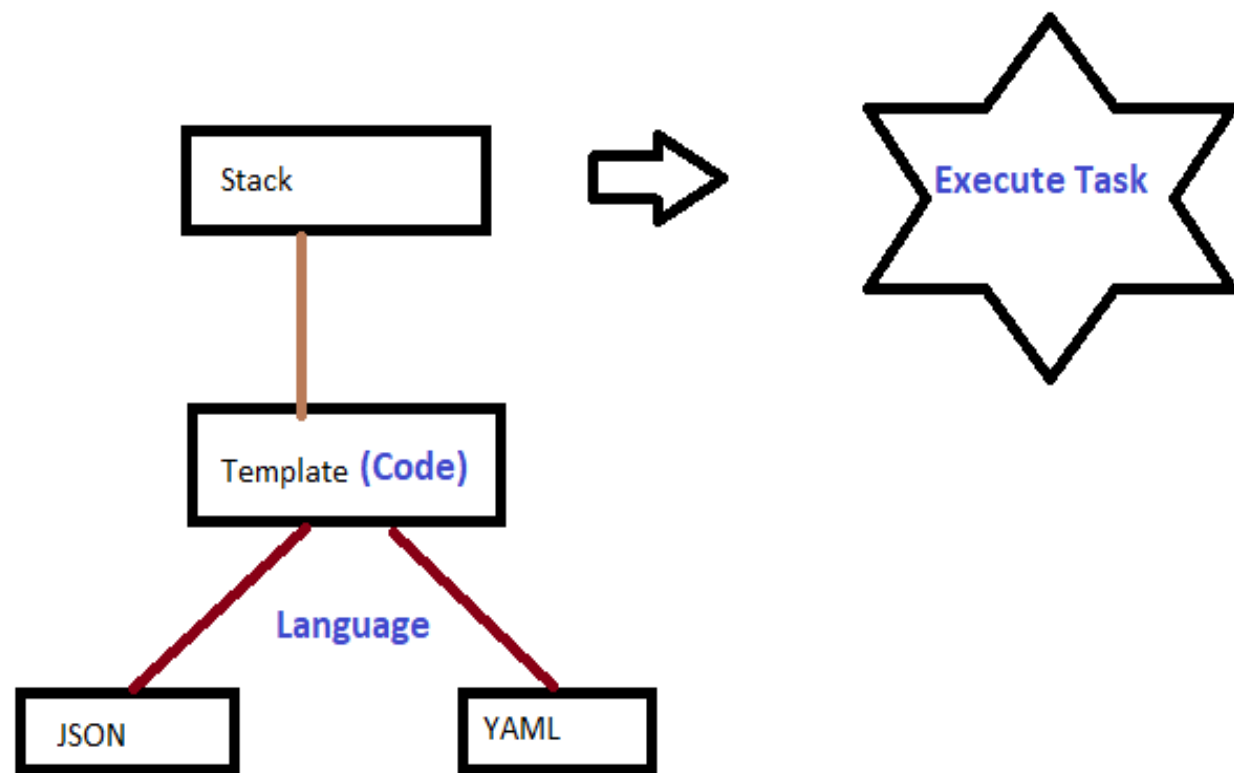




AWS
CloudFormation



Topics to be covered--Cloudformation

- 1) Introduction to IaasC
- 2) How to Use cloudformation sample template
- 3) Create a S3 bucket using cloudformation
- 4) Configure Wordpress server using cloudformation.
- 5) Configure Lamp Server
- 6) Create EC2 Instance using cloudformation.

What is configuration management?

- ✓ Configuration management is a process for maintaining computer systems, servers, and software in a desired, consistent state. It's a way to make sure that a system performs as it's expected to as changes are made over time.
- ✓ Automation plays an essential role in server configuration management. It's the mechanism used to make the server reach a desirable state, previously defined by provisioning scripts using a tool's specific language and features.
- ✓ Another common term used to describe the automation features implemented by configuration management tools is *Server Orchestration* or *IT Orchestration*, since these tools are typically capable of managing one to hundreds of servers from a central controller machine.

Configuration Management Benefits

- ✓ The primary benefit of configuration management is consistency of systems and software. With configuration management, you no longer guess or hope that a configuration is current. It is correct because the configuration management system ensures that it is correct.
- ✓ When combined with automation, configuration management can improve efficiency because manual configuration processes are replaced with automated processes. This also makes it possible to manage more targets with the same or even fewer resources.

DevOps Concepts: Pets vs. Cattle



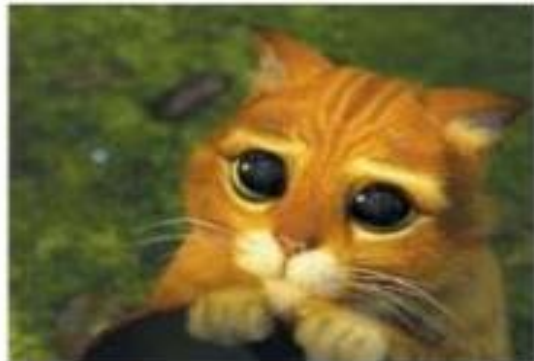
pets

vs



cattle

Service Model



- Pets are given names like `pussinboots.cern.ch`
- They are unique, lovingly hand raised and cared for
- When they get ill, you nurse them back to health



- Cattle are given numbers like `vm0042.cern.ch`
 - They are almost identical to other cattle
 - When they get ill, you get another one
-
- Future application architectures should use Cattle but Pets with strong configuration management are viable and still needed

Infrastructure as Code



Infrastructure as Code (IaC)

- IaC allows developers to modify infrastructure in a way that makes provisioning automated, faster, and repeatable. It's a key component of Agile and DevOps practices such as version control, continuous integration, and continuous deployment.
- Infrastructure as code can help with the following:
 - ✓ Improve speed:
 - ✓ Improve reliability:
 - ✓ Prevent configuration drift:
 - ✓ Support experimentation, testing, and optimization:

Infrastructure as Code



CloudFormation

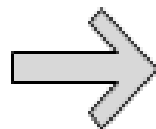
- ✓ Amazon Web Services CloudFormation is a free service that provides Amazon Web Service (AWS) customers with the tools they need to create and manage the infrastructure a particular software application requires to run on Amazon Web Services.
- ✓ An important advantage of CloudFormation is that it allows developers to automate service provisioning steps in a fairly simple way. There is no extra charge for AWS CloudFormation; customers only pay for the AWS resources that are required to run their applications.

CloudFormation

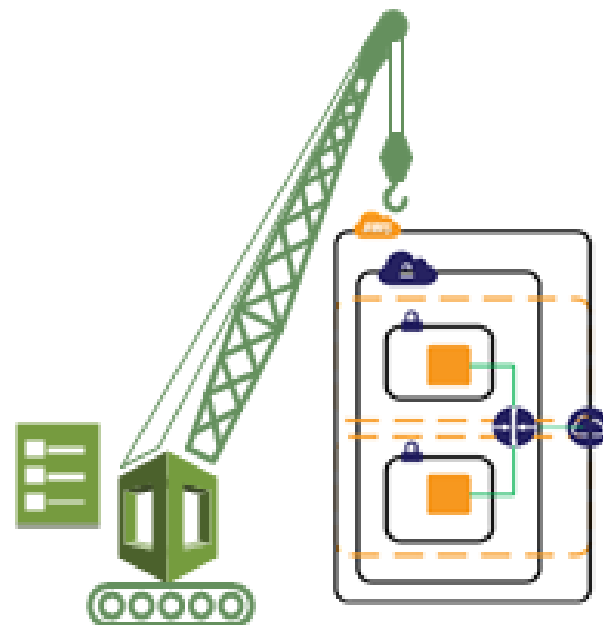
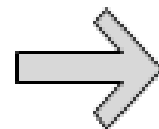
- ✓ CloudFormation has two parts: **templates** and **stacks**. A template is a JavaScript Object Notation (JSON) text file. The file, which is declarative and not scripted, defines what AWS resources or non-AWS resources are required to run the application.
- ✓ When the template is submitted to the service, CloudFormation creates the necessary resources in the customer's account and builds a running instance of the template, putting dependencies and data flows in the right order automatically. The running instance is called a stack.
- ✓ Customers can make changes to the stack after it's been deployed by using CloudFormation tools and an editing process that is similar to version control. When a stack is deleted, all related resources are deleted automatically as well.



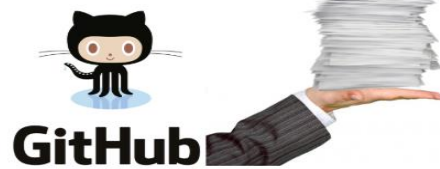
1 Create or use an existing template



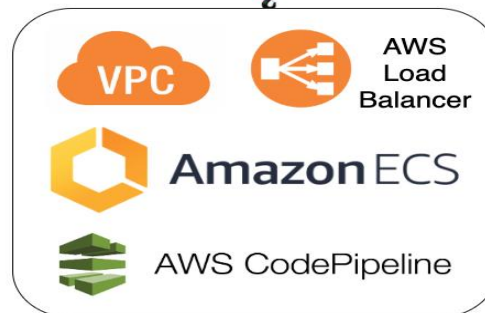
2 Save locally or in S3 bucket



3 Use AWS CloudFormation to create a stack based on your template. It constructs and configures your stack resources.



Source
Code



AWS
Cloudformation

CloudFormation Lab--1

- 1) Create a S3 bucket using cloudformation
- 2) Configure Wordpress server using cloudformation.
- 3) Configure Lamp Server
- 4) Create EC2 Instance using cloudformation.
- 5) How to Use cloudformation sample template

CloudFormation Lab--1

1) Create a S3 bucket using cloudformation.

Sol: Open cloudwatch --create Stack --create template in designer --select again

CloudFormation > Stacks > Create stack

Step 1
Specify template

Step 2
Specify stack details

Step 3
Configure stack options

Step 4
Review

Create stack

Prerequisite - Prepare template

Prepare template
Every stack is based on a template. A template is a JSON or YAML file that contains configuration information about the AWS resources you want to include in the stack.

☐ Template is ready

☐ Use a sample template

☒ Create template in Designer

Create template in Designer

Use the AWS CloudFormation Designer to graphically design your stack on a simple, drag-and-drop interface. The Designer automatically updates and validates the template JSON or YAML.

Create template in designer

S3 URL: Will be generated when sample template is created in Designer

View in Designer

Cancel Next

CloudFormation Lab--1

Left side scroll down –expand S3—drag bucket and keep in Middle windows –then click on create stack—next—give stack name—next—next-create stack

Go to S3 and check the created bucket

Resource types

- ▶ RDS
- ▶ Redshift
- ▶ ResourceGroups
- ▶ Route53
- ▼ S3
 - AccessPoint
 - Bucket**
 - BucketPolicy
- ▶ SDB
- ▶ SES
- ▶ SNS
- ▶ SQS

File: 'new.template'

S3B113SW Bucket

Properties Metadata DeletionPolicy DependsOn Condition

S3B113SW

```
1 {  
2   "Resources": {  
3     "S3B113SW": {  
4       "Type": "AWS::S3::Bucket",
```

CloudFormation Lab--2

1) Configure Wordpress server using cloudformation.

Open Cloudformation—create stack – Use a sample template—select wordpress(simple) --next

CloudFormation > Stacks > Create stack

Step 1
Specify template

Step 2
Specify stack details

Step 3
Configure stack options

Step 4
Review

Create stack

Prerequisite - Prepare template

Prepare template
Every stack is based on a template. A template is a JSON or YAML file that contains configuration information about the AWS resources you want to include in the stack.

☐ Template is ready ☒ Use a sample template ☐ Create template in Designer

Select a sample template

View more sample templates [↗](#)

Sample templates
This collection of sample templates will help you get started with AWS CloudFormation and quickly build your own templates

WordPress blog ▼

S3 URL: https://s3.ap-south-1.amazonaws.com/cloudformation-templates-ap-south-1/WordPress_Single_Instance.template [View in Designer](#)

Cancel [Next](#)

CloudFormation Lab--2

1) Fill the database detail –next—next--create

CloudFormation > Stacks > Create stack

Step 1
Specify template

Step 2
Specify stack details

Step 3
Configure stack options

Step 4
Review

Specify stack details

Stack name

Stack name

mywordpressserver

Stack name can include letters (A-Z and a-z), numbers (0-9), and dashes (-).

Parameters

Parameters are defined in your template and allow you to input custom values when you create or update a stack.

DBName
The WordPress database name

wordpressdb1

DBPassword
The WordPress database admin account password

DBRootPassword
MySQL root password

DBUser
The WordPress database admin account username

InstanceType
WebServer EC2 instance type

t2.micro

KeyName
Name of an existing EC2 KeyPair to enable SSH access to the instances

webserverkey.pem

SSHLocation
The IP address range that can be used to SSH to the EC2 instances

0.0.0.0/0

Cancel Previous Next