Contents

[Overview 1](#_Toc68261772)

[Basics & Terminology 2](#_Toc68261773)

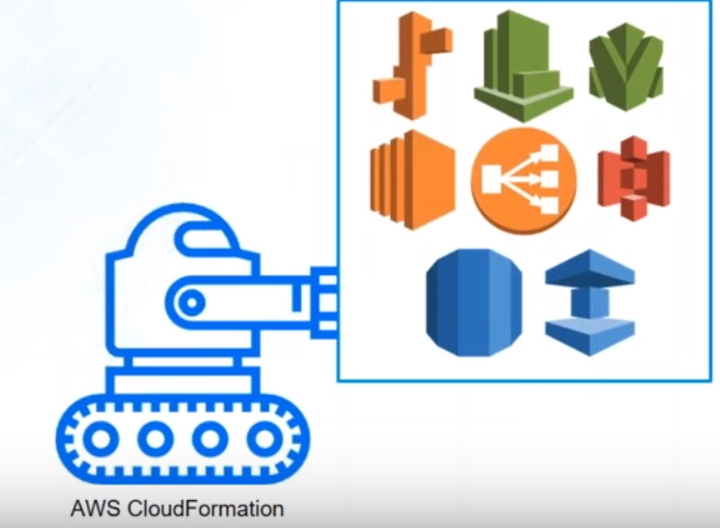
[Work Flow 3](#_Toc68261774)

[Structure of Template 4](#_Toc68261775)

[References 4](#_Toc68261776)

# Overview

* A Complex application on AWS can have many resources and managing all these resources can be a mundane task.
* ClodFormation is an Infrastructure as code tool for AWS
* CloudFormation provides AWS Infrastructure
  + Safe
  + Repeatable
  + Minimal manual actions
  + No custom scripts

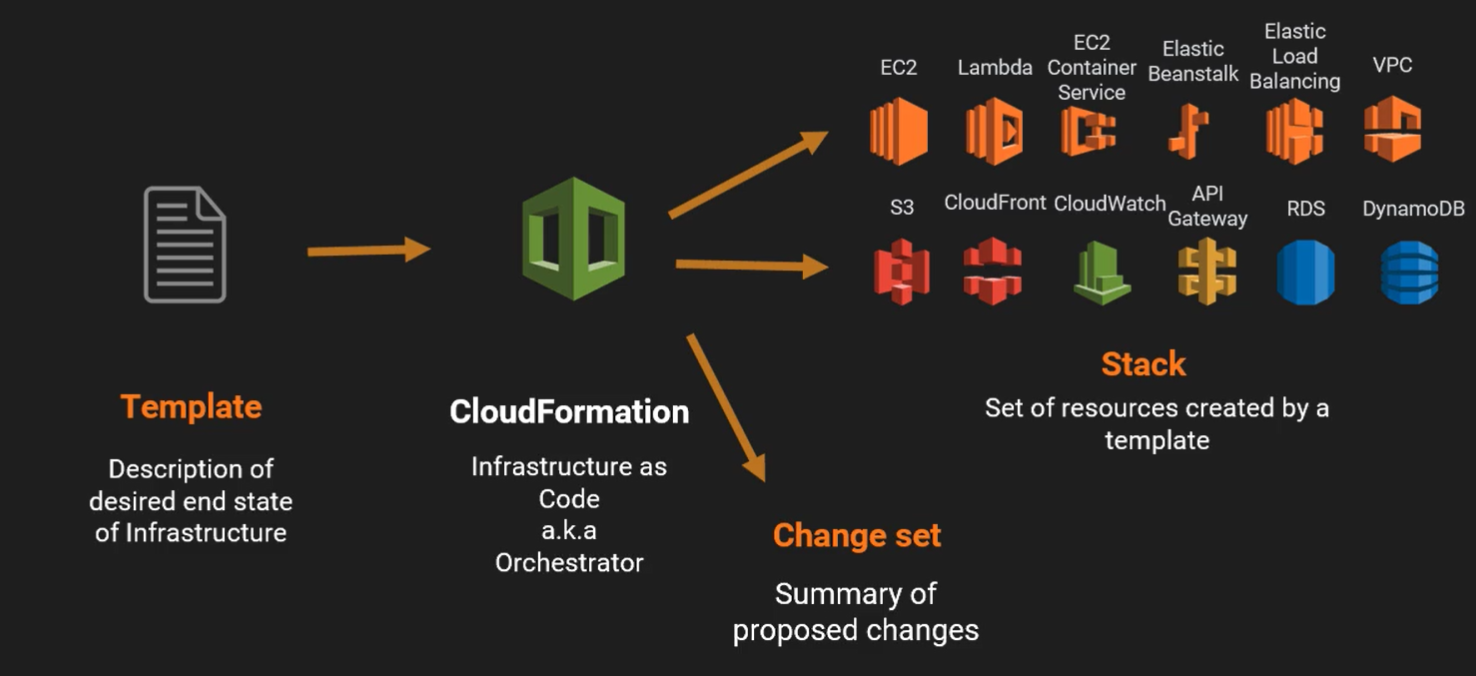


* AWS CloudFormation is a service that helps you model and setup your AWS resources so that you can spend less time managing those resources and more time focusing on your application that run in AWS.
* For creating templates in AWS CloudFormation you use a JSON or YAML script.
* CF Template gives easier way to create aws application/infrastructure creational/update templates.
* It allows us to define parameters
* Manage dependencies
* Easily control and track changes of Infrastructure
* Rollback or delete Cleanly
* Template can be shared with others to create the application/infra in any account.

# Basics & Terminology

* Evaluate the Scenario:
  + Console: Creation is simple but repetition can be difficult
  + CLI: Scripting might require some prior knowledge, but changes will be difficult to handle in repetition
  + CF Template: Create is time consuming but repetition is effective and easy.
* Template:
  + Text file - JSON/YAML format which has resources, parameters etc
  + Input into CloudFormation
  + Describes end state of the Infrastructure
* Stack:
  + When CloudFormation executes a template, it creates a stack.
  + To update the resources within a template, you need to update a stack.
  + A Set of related resources as a single unit is called a stack
* Change Set –
  + Before updating a stack, you can generate a change set.
  + A Change set allows you to see how the changes will impact your running resources.
  + This can be very important for live systems.
* Resource: Any thing which can be created in AWS
* Parameter: Input from user
* Output: result of execution of Template

# Work Flow



# Structure of Template

* AWS Template Format Version (2010-09-09)
* Description
* Metadata
* Parameters
* Mappings
* Conditions
* Outputs
* Resources (The Only Section that is required)

# References

* Github URL for CF Templates

<https://github.com/skatta3/intro-to-CloudFormation_AC>

* Template Anatomy

<https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/template-anatomy.html>