

AWS CloudFormation:

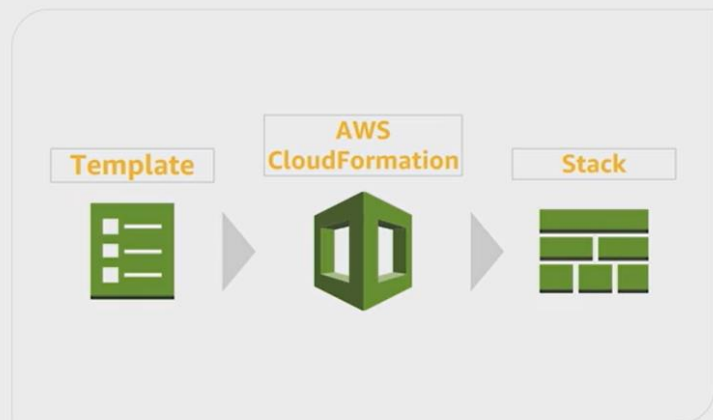
- Infrastructure as a code

Template

- Define resources to create.
- Code your infrastructure in either JSON and YAML format.

Stack

- Create from templates.
- Create multiple stacks (in multiple regions) from the same template.
- Monitor progress of stack updates: in progress, complete, failed (entire stack will roll back).



- Structure of an AWS CloudFormation Template:

Other AWS CloudFormation Elements

- Intrinsic functions
- Pseudo-parameters
- Custom resources
- Conditional expressions



AWS CloudFormation Template Sections

AWSTemplateFormatVersion: "2010-09-09"

Description: "template description"

Metadata: # template metadata

Parameters: # set of parameters

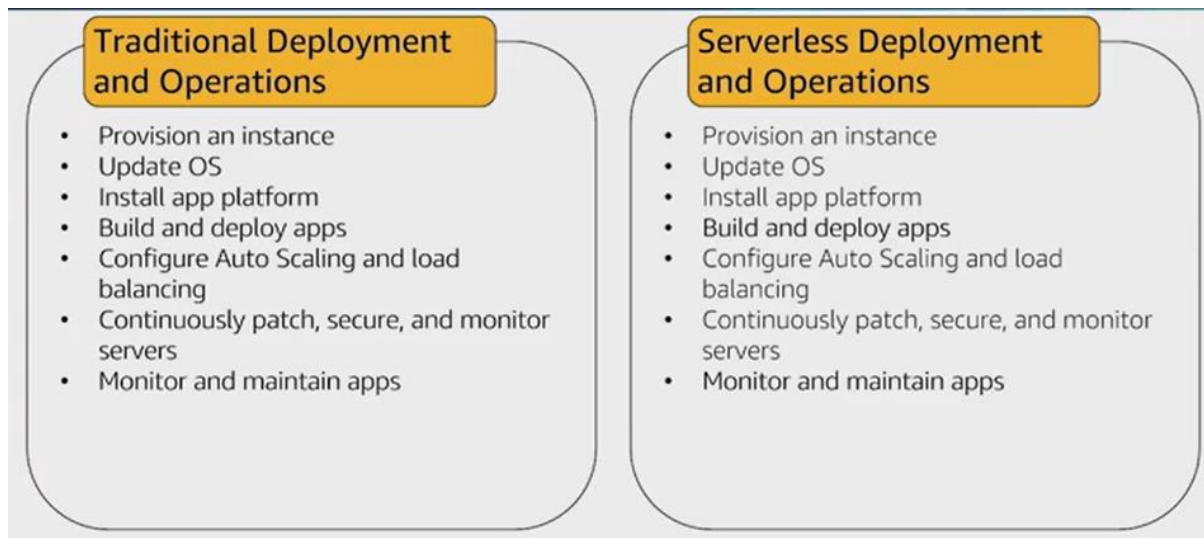
Mappings: # set of mappings

Conditions: # set of conditions

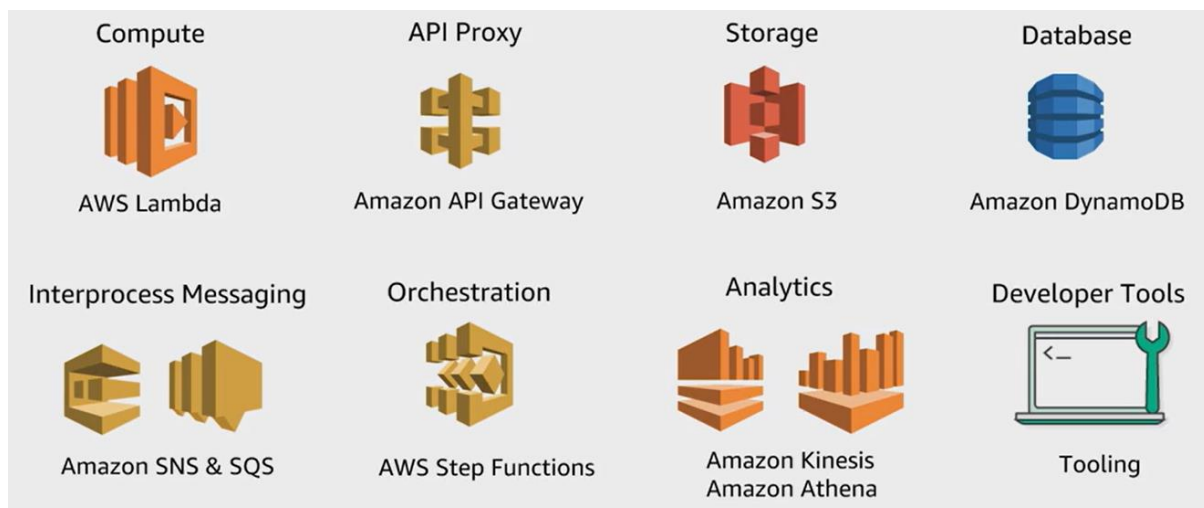
Resources: # set of resources

Outputs: # set of outputs

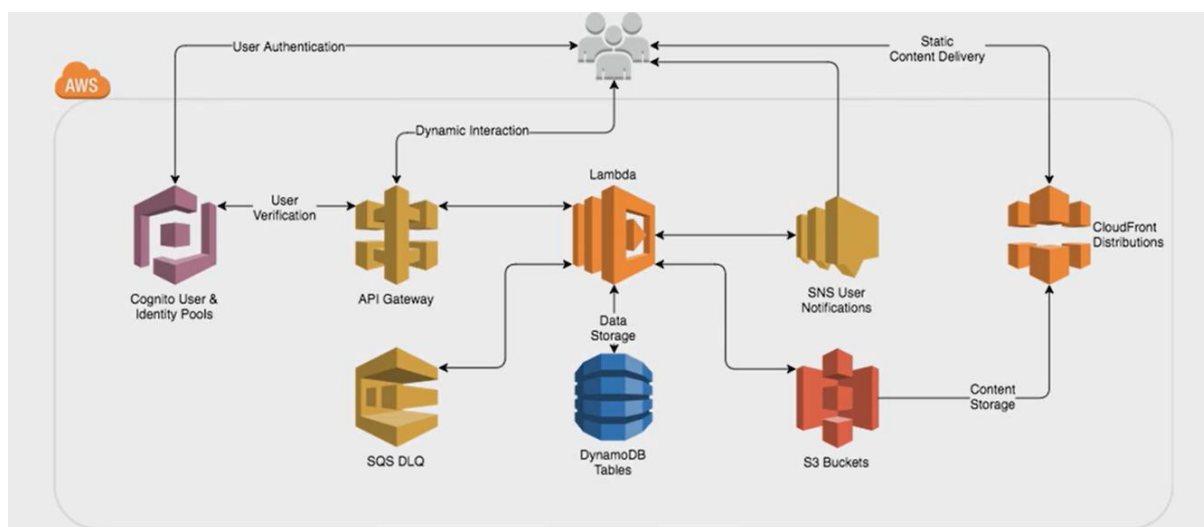
Serverless computing:



AWS serverless platform:



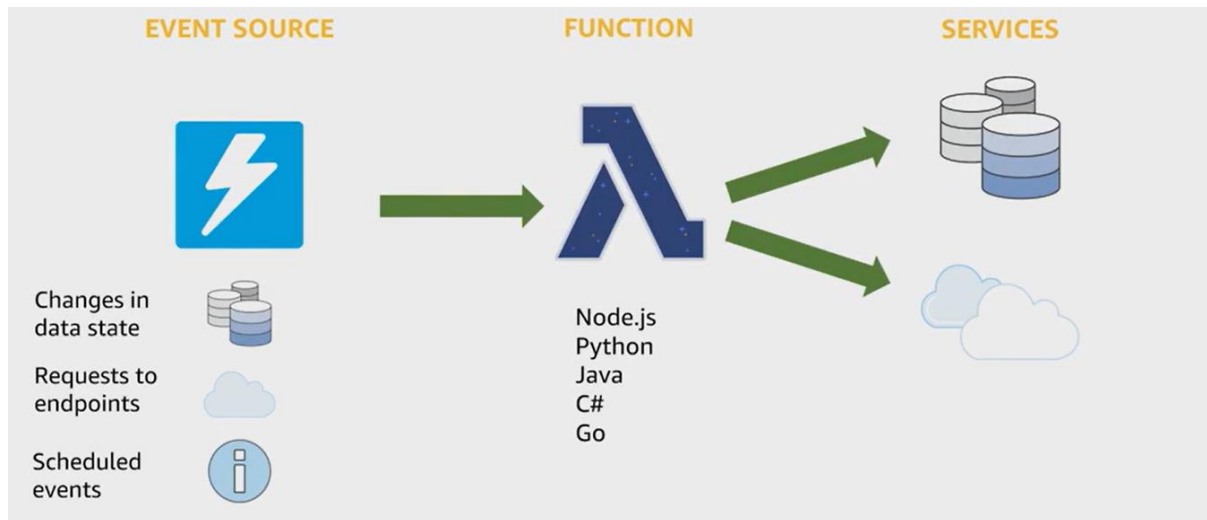
Serverless Architecture:



Serverless Application Model (SAM):

- CloudFormation Extension optimised for serverless
- New serverless resource types: functions, APIs and Tables
- Supports anything CloudFormation supports.

AWS lambda: Serverless applications:



Examples of event sources that trigger AWS Lambda:



Test Axioms:

- Elastic Load Balancing and Auto Scaling are designed to work together.
- Scaling out is better than scaling up.
- AWS Elastic Beanstalk allows you to focus on building your application.
- CloudFormation templates allow you to have a definition of resources to create.
- A serverless application is typically a combination of Lambda + other AWS services.