Managing AWS Resources with Terraform

Michael Bright, Terraform Associate

for ArdanLabs, 2021-Jul-20

 https://linkedin.com/in/mjbright

 @mjbright

 @mjbright

Agenda

Managing AWS Resources with Terraform

Intro: Infra as Code / Terraform

AWS features

AWS concepts

AWS EC2 VMs ++

AWS S3

What is Infrastructure as Code ?

IaC

What is Infrastructure as Code (IaC) ?

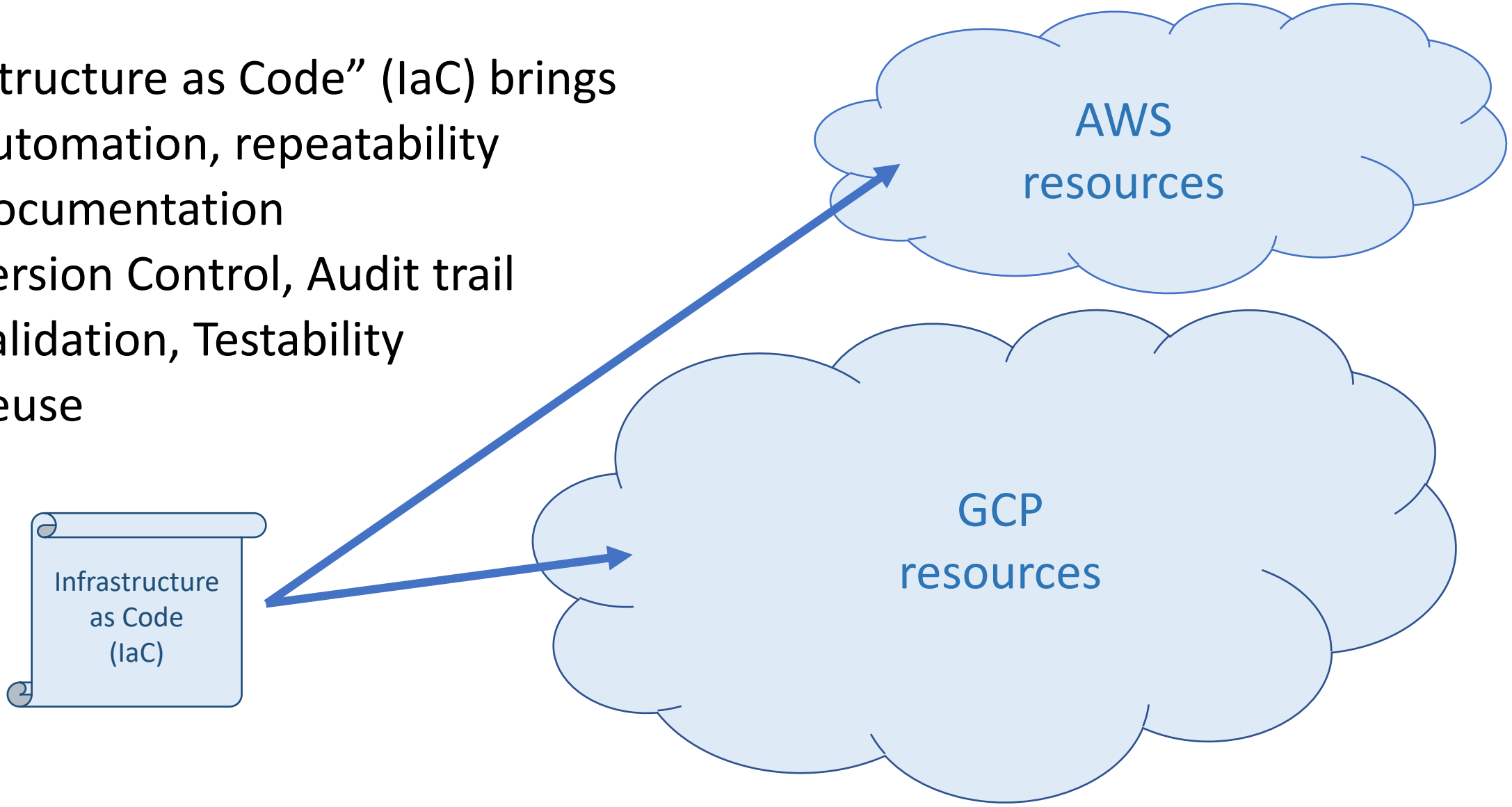
Why Infrastructure as Code ?

Why Terraform ?

Introduction

"Infrastructure as Code" (IaC) brings

- Automation, repeatability
- Documentation
- Version Control, Audit trail
- Validation, Testability
- Reuse



Terraform Concepts

Concepts & Benefits

Desired state

Idempotency

Repeatable, & auditable

Multi-environment
- via provider plugins

Terraform Workflow

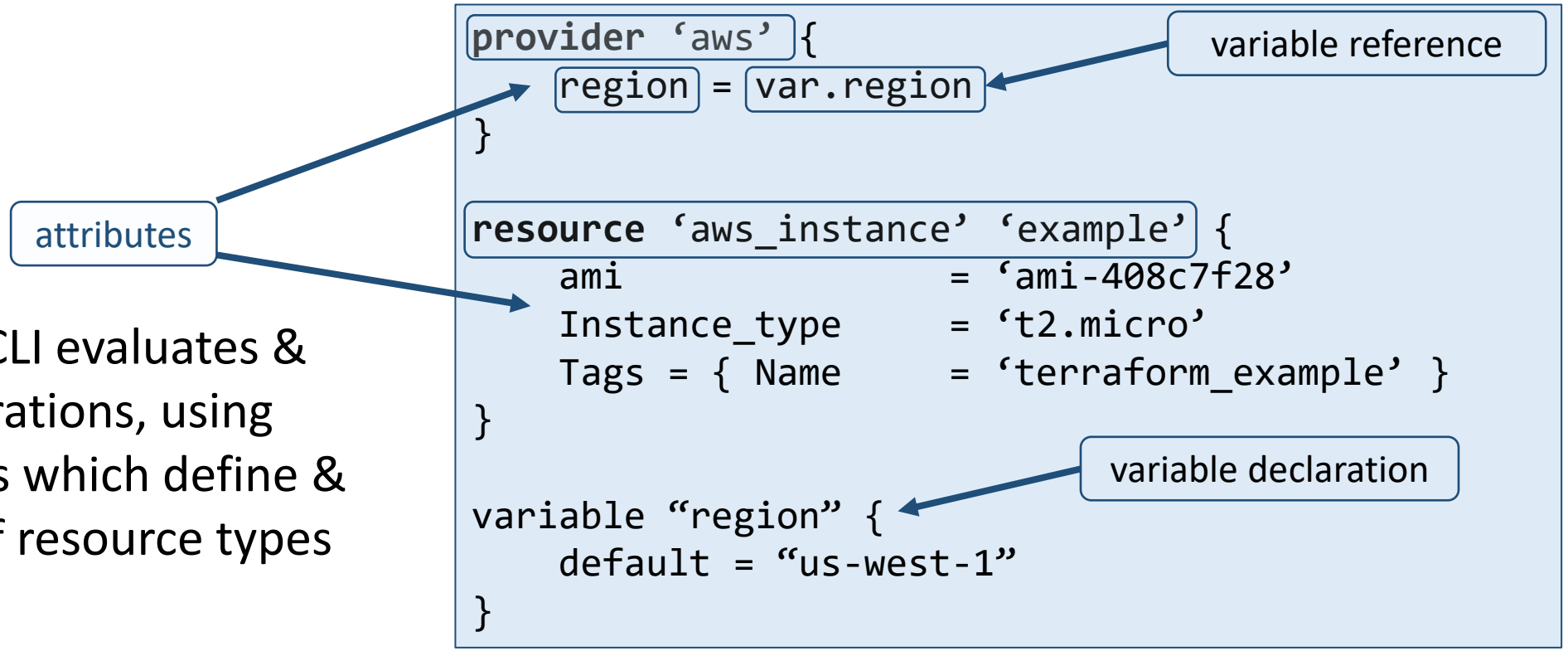
Use of Terraform is according to the workflow

- terraform init - download & initialize specified providers
- terraform plan - parse and check config
- terraform apply - create or update all resources
- terraform destroy - destroy all resources

https://www.terraform.io/docs/commands/index.html

Example Terraform Configurations

Configurations use Hashicorp Configuration Language (HCL v2) & specify a provider & several resources



The Terraform CLI evaluates & applies configurations, using provider plugins which define & manage a set of resource types

JSON format can also be used – files named as .tf.json

Terraform Template Files

Template file naming

File naming is unconstrained, so be consistent
– common practice is to use at least 3 files per module:

- *providers.tf*: *[only in root module]* declares providers & versions to use

- *main.tf*: resource definitions
- *outputs.tf*: exported data items
- *variables.tf*: input variables, with optional default value

- *modules/*: *contains any sub-modules*

Terraform – Getting started

Installing

Download Terraform for your OS/architecture

Unzip the downloaded zip file and place the terraform static binary in your PATH, e.g.

```
$ wget https://releases.hashicorp.com/terraform/1.0.2/terraform_1.0.2_linux_amd64.zip
$ unzip terraform_1.0.2_linux_amd64.zip
$ chmod +x terraform
$ mv terraform /usr/local/bin
$ terraform version
Terraform v1.0.2 on linux_amd64
```

macOS
64-bit | Arm64

FreeBSD
32-bit | 64-bit | Arm

Linux
32-bit | 64-bit | Arm | Arm64

OpenBSD
32-bit | 64-bit

Solaris
64-bit

Windows
32-bit | 64-bit

https://www.terraform.io/downloads.html

AWS Resources

AWS Resources manageable by Terraform

ACM PCA API Gateway Application Autoscaling AppMesh AppSync Athena Autoscaling Backup Batch Budgets Cloud9 CloudFormation CloudFront CloudHSM v2 CloudTrail CloudWatch CodeBuild CodeCommit CodeDeploy CodePipeline Cognito Config Cost & Usage Report Data Lifecycle Manager (DLM) Database Migration Service (DMS) DataPipeline DataSync Device Farm Directory Service Direct Connect DynamoDB Accelerator (DAX) DocumentDB **EC2 ECR ECS** EFS **EKS** ElastiCache Elastic Beanstalk **Elastic Load Balancing v2 (ALB/NLB)** Elastic Map Reduce (EMR) ElasticSearch Elastic Transcoder Firewall Manager (FMS) File System (FSx) Gamelift Glacier Global Accelerator Glue GuardDuty HoneyComb **IAM** IoT Inspector Kinesis Kinesis Firehose KMS **Lambda** License Manager Lightsail Macie MQ MediaPackage MediaStore Managed Streaming for Kafka (MSK) Neptune OpsWorks Organizations Pinpoint Pricing QuickSight RAM **RDS** Redshift Resource Groups **Route53** Resolver **S3** Sagemaker Secrets Manager Security Hub SES Service Catalog Service Discovery Service Quotas Shield SimpleDB SNS SQS SSM Step Function (SFN) Storage Gateway SWF Transfer **VPC** WAF WAF Regional WorkLink WorkSpaces XRay

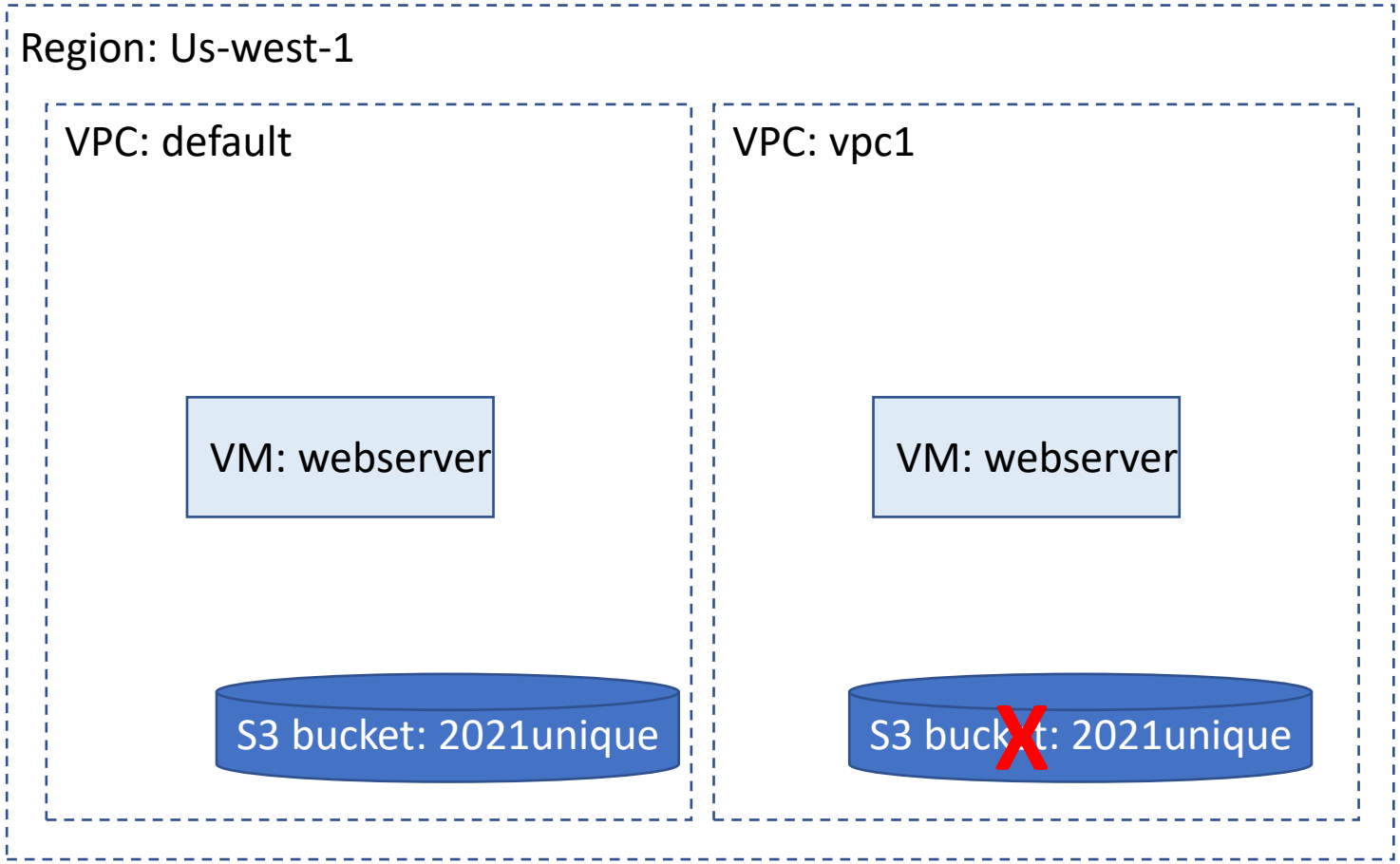
AWS Concepts

Concepts - Regions

Regions, Availability Zones, VPCs

EC2 VMs

S3 object storage

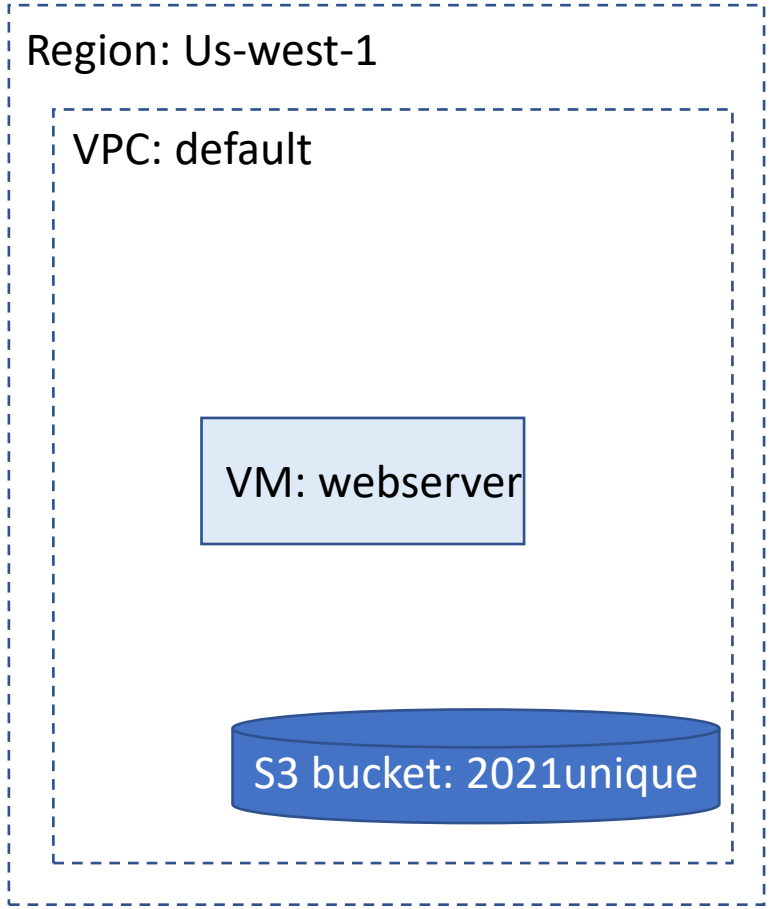


AWS Concepts

Concepts – S3 Object Storage

A bucket is a container for objects stored in S3
Each object is contained in a bucket

S3 buckets
- are Region specific
- have a globally unique name



SQS – Simple Queue Service can be used to monitor S3 bucket activity

Demo time

Want more ?

Come back for the Tuesday 24th August session on *"Serverless deployments with Terraform"*

Sign up for training in the fall

https://learn.hashicorp.com/terraform

Thank you !

 https://linkedin.com/in/mjbright

 @mjbright

 @mjbright

Thank you !

 https://linkedin.com/in/mjbright

 @mjbright

 @mjbright