

Dipartimento di Ingegneria
Corso di laurea in Ingegneria
Informatica



Creazione di infrastrutture in Cloud attraverso il codice

Relatore
Prof. Luca Cabibbo

Laureando
Andrea Di Paola
Matr. 548039

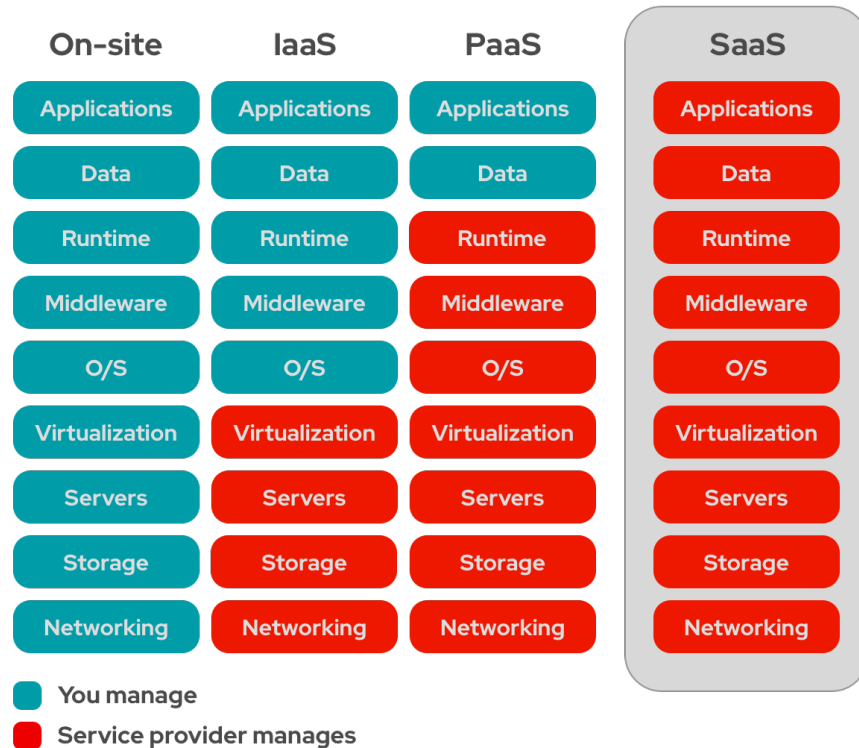
OBIETTIVI

- Acquisire conoscenze sul Cloud Computing
- Apprendere i motivi e gli approcci della Migrazione verso il Cloud
- Creare infrastrutture in Cloud
- Sperimentare automazione nel processo di provisioning di risorse



Capgemini

CLOUD COMPUTING



I modelli di Servizio:

- Infrastructure as a Service (IaaS)
- Platform as a Service (PaaS)
- Software as a Service (SaaS)

CLOUD MIGRATION

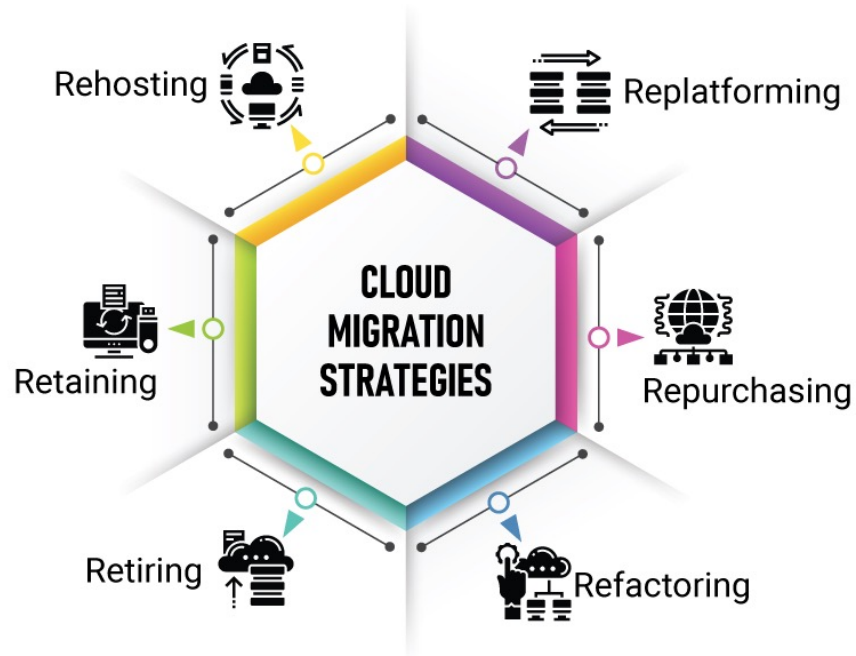
I vantaggi di spostare dati e applicazioni su Cloud:

- Modalità di pagamento pay-per-use
- Alta scalabilità
- Sistemi di sicurezza offerti dal Provider
- Opportunità di modernizzare gli applicativi



CLOUD MIGRATION

TYPES OF CLOUD MIGRATION STRATEGIES



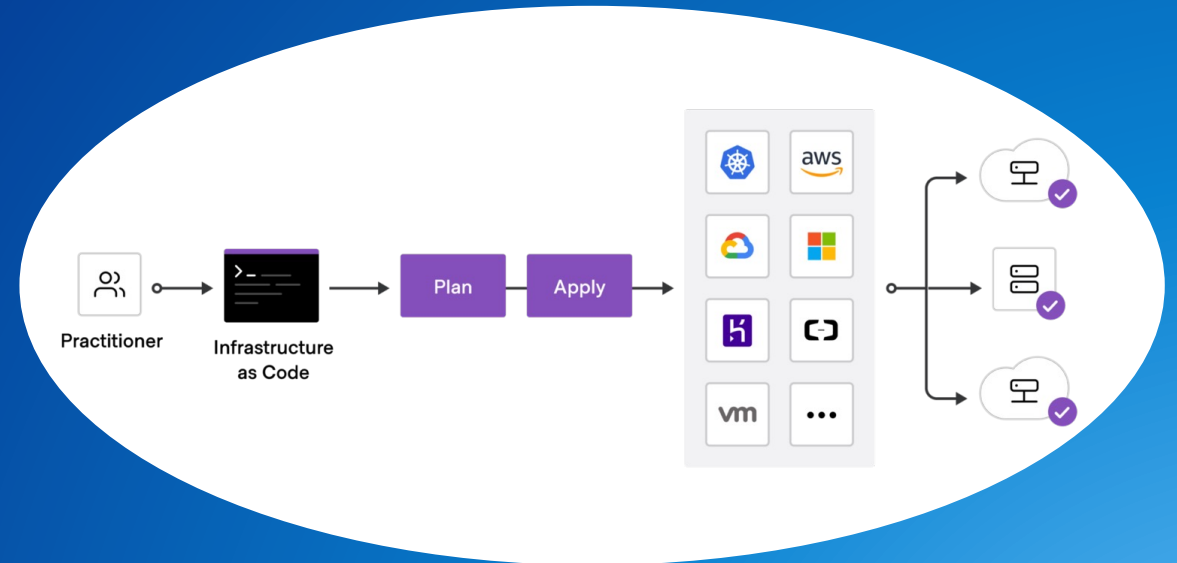
Le sei strategie di migrazione:

- Rehosting
- Replatforming
- Refactoring
- Repurchasing
- Retiring
- Retaining

INFRASTRUCTURE AS CODE

IaC è un metodo che permette di modellare infrastrutture attraverso il codice

- Riduce i tempi per i rilasci
- Riduce gli errori
- Favorisce il riutilizzo del codice



STRUMENTI DI LAVORO



AMAZON WEB SERVICES

- Modello Cloud IaaS
- Offre 150+ di servizi gestiti
- Alta affidabilità e sicurezza
- Infrastruttura dinamica

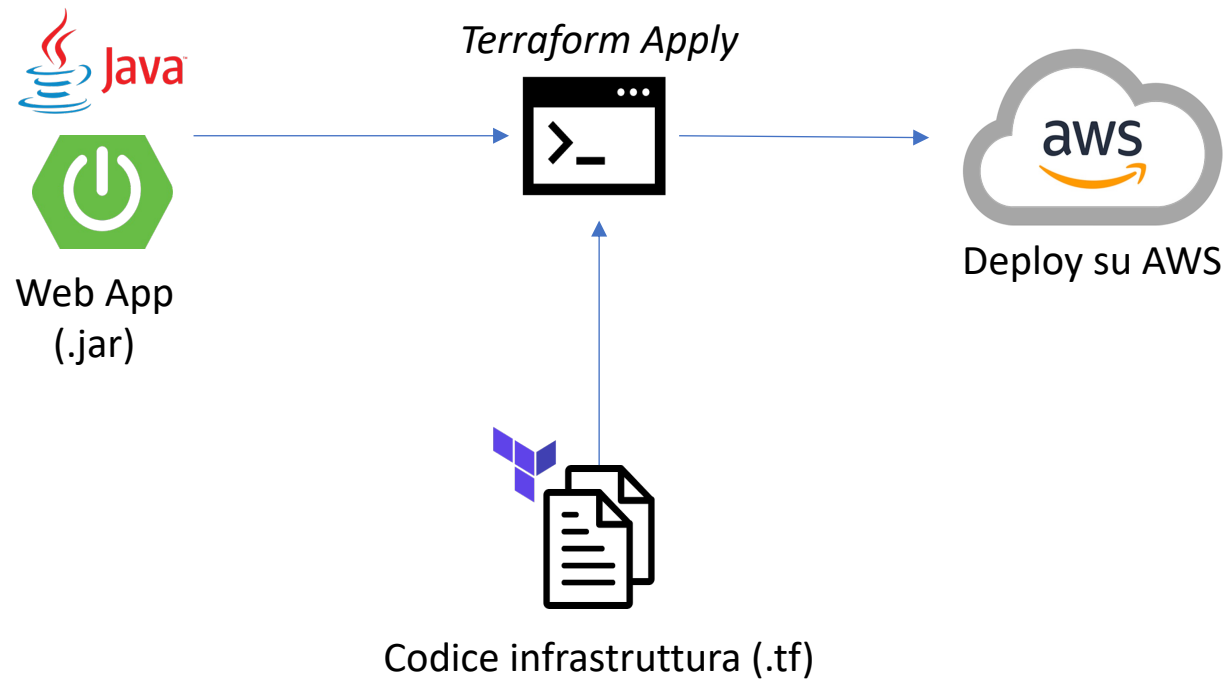
STRUMENTI DI LAVORO

TERRAFORM

- È uno strumento che consente di applicare Infrastructure as code.
- È un linguaggio di programmazione dichiarativo.
- Consente di lavorare con un'ampia scelta di providers cloud.
- Tiene traccia del provisioning delle risorse.



MIGRAZIONE WEB APP



Come sperimentazione, si può sfruttare Terraform per il provisioning delle risorse e il deployment automatizzato di una web app Spring Boot scritta in java

PROCEDURA

1. Inizializzazione

Definizione del Cloud provider, del data center e delle variabili a supporto delle risorse (es. credenziali AWS).

```
init.tf > ...
1 terraform {
2   required_providers {
3     aws = {
4       source = "hashicorp/aws"
5       version = "=3.75.2"
6     }
7   }
8
9   required_version = ">= 0.14.9"
10 }
11
12 provider "aws" {
13   region = var.aws_region
14   access_key = var.aws_access_key
15   secret_key = var.aws_secret_key
16 }
17
```

```
variables.tf > ...
1 variable "aws_region" {
2   type = string
3   default = "eu-central-1"
4 }
5
6 variable "aws_az" {
7   type = string
8   default = "eu-central-1a"
9 }
10
11 variable "aws_access_key" {
12   type = string
13 }
14
15 variable "aws_secret_key" {
16   type = string
17 }
18
```

- aws.tf
- credentials.auto.tfvars
- HelloWorld.jar
- init.tf
- variables.tf

PROCEDURA

2. Le risorse

Risorsa AWS	Codice Terraform
Virtual Private Cloud (VPC)	aws_vpc
Subnet	aws_subnet
Internet Gateway	aws_internet_getaway
Gruppo di sicurezza	aws_security_group
Route table	aws_route_table
Ricerca dell'AMI (Immagine della Macchina Virtuale)	aws_ami
Istanza Amazon EC2	aws_instance

PROCEDURA

3. Creazione App Server

```
resource "aws_instance" "app_server" {  
  ami                = data.aws_ami.ubuntu.id  
  instance_type      = "t2.micro"  
  associate_public_ip_address = true  
  subnet_id          = aws_subnet.server_subnet.id  
  vpc_security_group_ids = [aws_security_group.allow_HTTP_and_SSH.id]  
  key_name            = "terraformKeys"
```

Definizione dell'immagine e il tipo di macchina virtuale

Copia del file jar attraverso connessione SSH

```
# Runs as root  
user_data = <<SCRIPT  
#!/bin/bash  
apt update -y  
apt upgrade -y  
apt install -y openjdk-11-jre-headless  
java -jar /home/ubuntu/HelloWorld.jar  
SCRIPT
```

Script per installare java ed eseguire l'applicativo

```
provisioner "file" {  
  source      = "HelloWorld.jar"  
  destination = "/home/ubuntu/HelloWorld.jar"  
  
  connection {  
    type      = "ssh"  
    host      = self.public_ip  
    user      = "ubuntu"  
    private_key = file("/Users/andreadipaola/Documents/terraformKeys.pem")  
  }  
}
```

DIMOSTRAZIONE

Il comando *terraform apply* costruisce l'infrastruttura

→ `terraform_webserver terraform apply`

☑	Name	ID Istanza	Stato dell'ista...	Tipo di istanza	Verifica dello st...	Stato dell'all...	Availability Zone	DNS IPv4 pubblico
☑	app_server	i-045734a62d34bc655	🟢 In esecuzione	t2.micro	🟢 2/2 controlli supe	Nessun al...	eu-central-1a	-

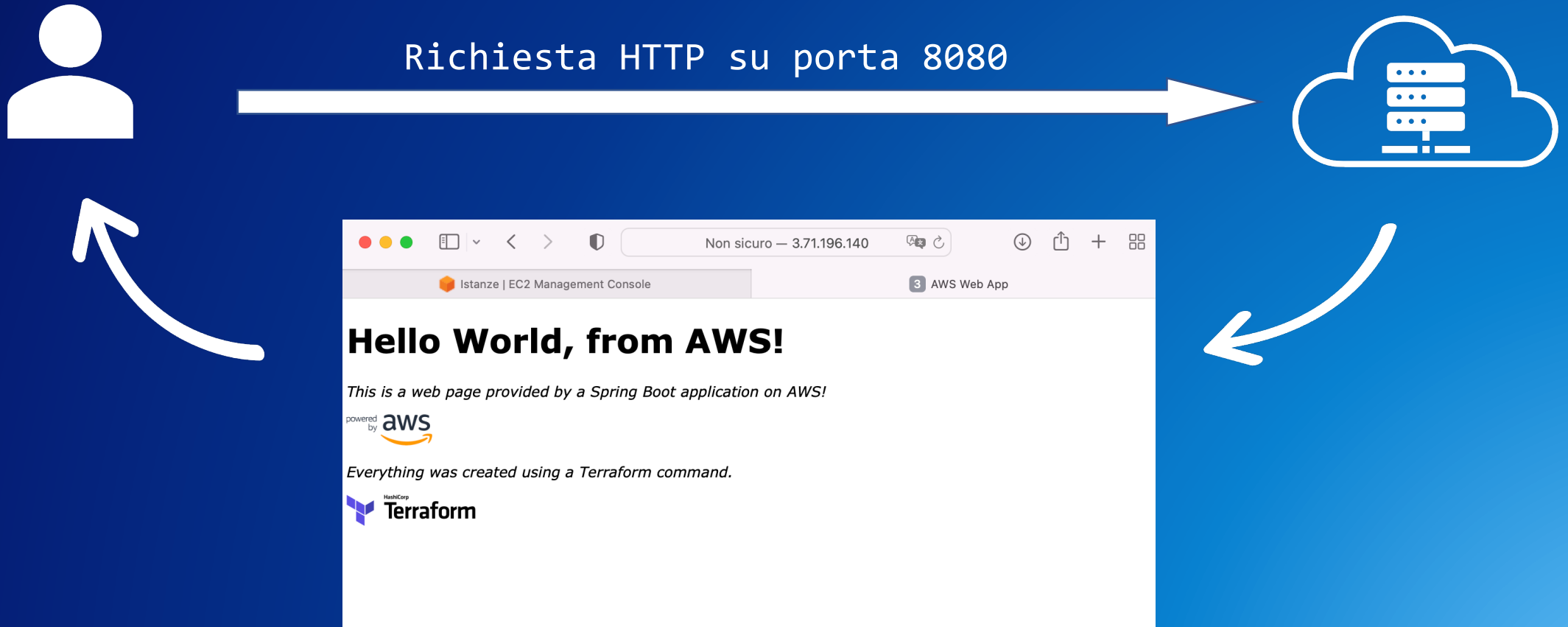
Istanza: i-045734a62d34bc655 (app_server)	
Dettagli	Sicurezza Reti Storage Verifiche di stato Monitoraggio in corso Tag
▼ Riepilogo dell'istanza Informazioni	
ID Istanza i-045734a62d34bc655 (app_server)	Indirizzo IPv4 pubblico 🟢 In esecuzione
Indirizzo IPv6 -	Indirizzo IPv4 privato 3.71.196.140 indirizzo aperto
Tipo di nome host Nome IP: ip-192-168-0-168.eu-central-1.compute.internal	Nome DNS IP privato (solo IPv4) ip-192-168-0-168.eu-central-1.compute.internal
Risponde al nome DNS della risorsa privata -	Tipo di istanza t2.micro
	Indirizzi IPv4 privati 192.168.0.168
	DNS IPv4 pubblico -
	Indirizzi IP elastici -

```
aws_vpc.server_vpc: Creating...
aws_vpc.server_vpc: Creation complete after 2s [id=vpc-057efea20cd83e0a6]
aws_internet_gateway.server_igw: Creating...
aws_subnet.server_subnet: Creating...
aws_security_group.allow_HTTP_and_SSH: Creating...
aws_subnet.server_subnet: Creation complete after 1s [id=subnet-094bbdf6709e0595d]
aws_internet_gateway.server_igw: Creation complete after 1s [id=igw-0768911af2e777176]
aws_route_table.server_rt: Creating...
aws_route_table.server_rt: Creation complete after 2s [id=rtb-050a39fdbcb9f21443]
aws_route_table_association.route_table_association: Creating...
aws_route_table_association.route_table_association: Creation complete after 0s [id=rtbassoc-0e182eea077705955]
aws_security_group.allow_HTTP_and_SSH: Creation complete after 3s [id=sg-0ec4900bef9b0da6d]
aws_instance.app_server: Creating...
aws_instance.app_server: Still creating... [10s elapsed]
aws_instance.app_server: Still creating... [20s elapsed]
aws_instance.app_server: Provisioning with 'file'...
aws_instance.app_server: Still creating... [30s elapsed]
aws_instance.app_server: Still creating... [40s elapsed]
aws_instance.app_server: Creation complete after 43s [id=i-045734a62d34bc655]

Apply complete! Resources: 7 added, 0 changed, 0 destroyed.
+ terraform_webserver
```

Accedendo alla console AWS
si possono monitorare le risorse

DIMOSTRAZIONE



CONCLUSIONI

Nel contesto di un progetto di Migrazione verso il cloud:

- Sono stati studiati gli approcci di migrazione
- Sono state effettuate delle sperimentazioni di automazione dell'infrastruttura

Sviluppi futuri prevedono la creazione di infrastrutture complesse che permettono di eseguire applicazioni a micro servizi eseguiti su cluster di VMs o Containers

GRAZIE
PER L'ATTENZIONE!