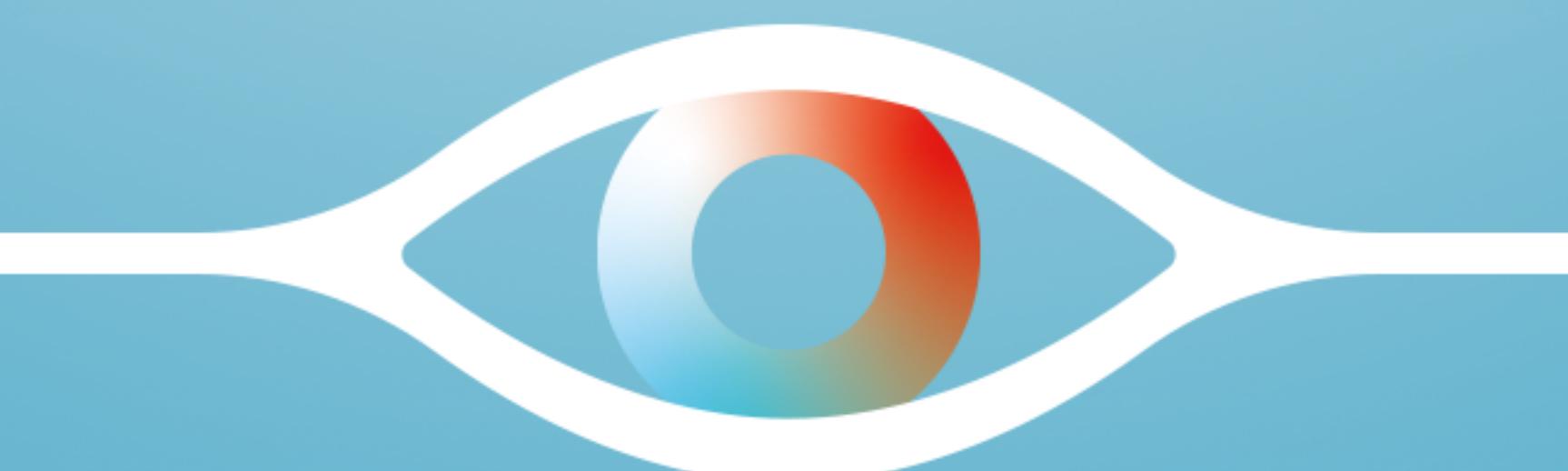


February 6th-7th

nic 20/20 VISION

Oslo Spektrum



Design, Draw, Deploy your AWS infrastructure from inception to production

Anton Babenko
@antonbabenko

February 2020

nic

Anton Babenko

AWS Community Hero / Certified Terraform fanatic since 2015

Organiser of HashiCorp UG, AWS UG, DevOps Norway, DevOpsDays Oslo

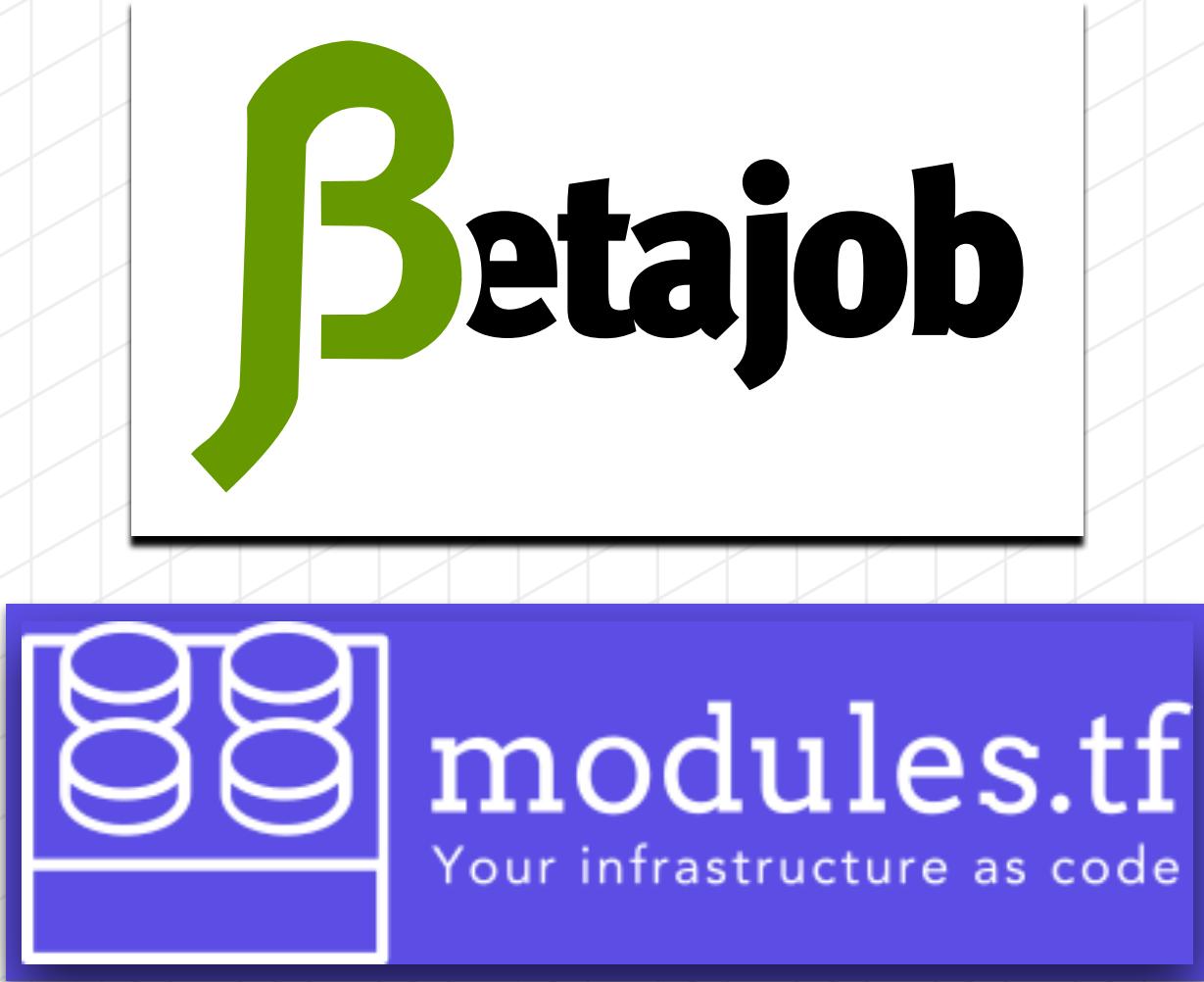
I ❤️ open-source:

- [terraform-community-modules + terraform-aws-modules](#)
- [antonbabenko/pre-commit-terraform](#) – clean code and documentation
- [antonbabenko/modules.tf-lambda](#) – generate Terraform code from visual diagrams
- [antonbabenko/terragrunt-reference-architecture](#) – Terragrunt reference architecture
- [www.terraform-best-practices.com](#)
- [@antonbabenko](#) – Twitter, GitHub, LinkedIn



What do I do?

- All-things Terraform + AWS + DevOps
- Consulting
- Workshops
- Trainings
- Mentorship



My company: betajob.com

My blog: antonbabenko.com



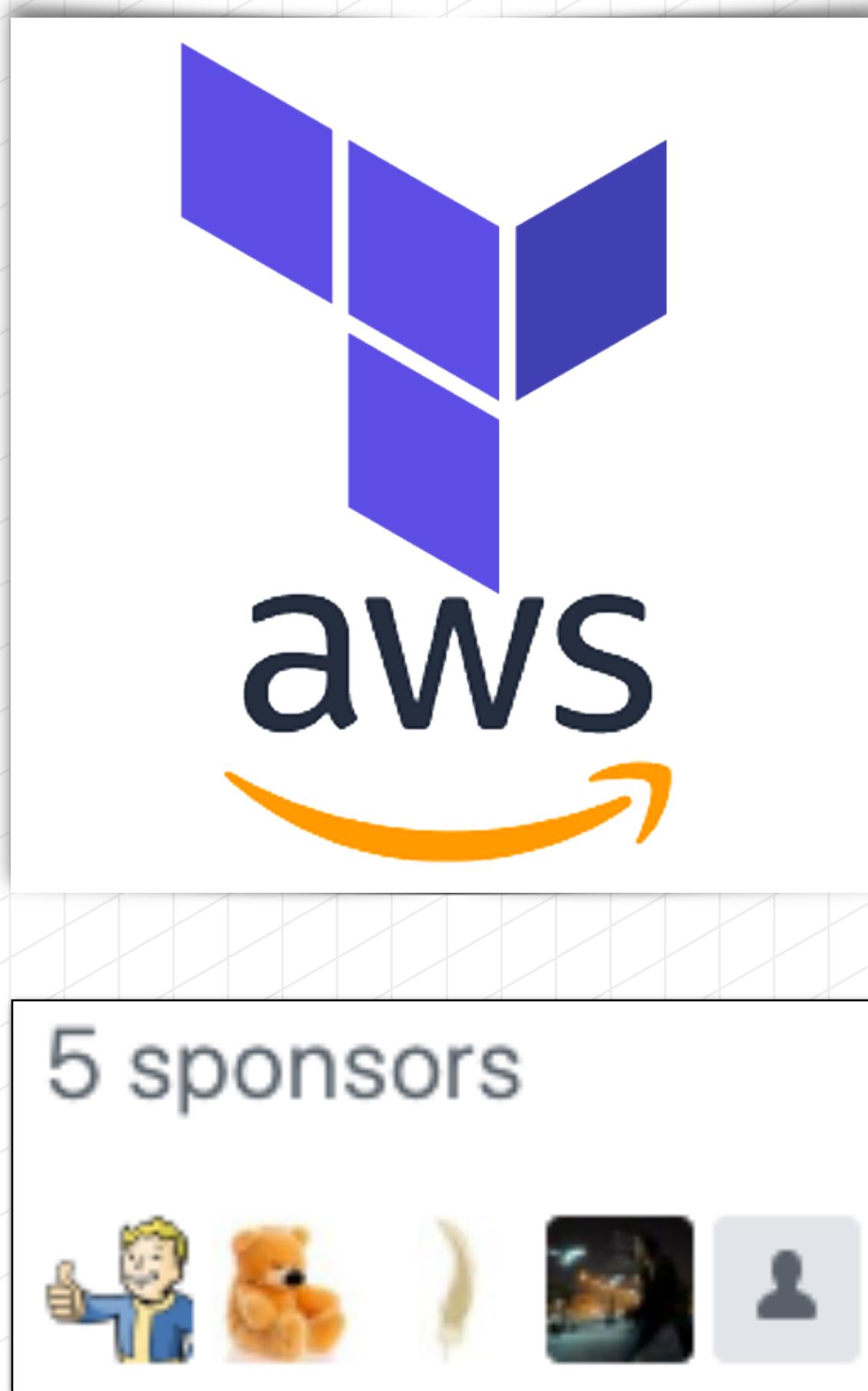
Collection of open-source Terraform AWS modules supported by the community.

More than 7 mil. downloads since September 2017.

(VPC, Autoscaling, RDS, Security Groups, ELB, ALB, Redshift, SNS, SQS, IAM, EKS, ECS, TGW...)

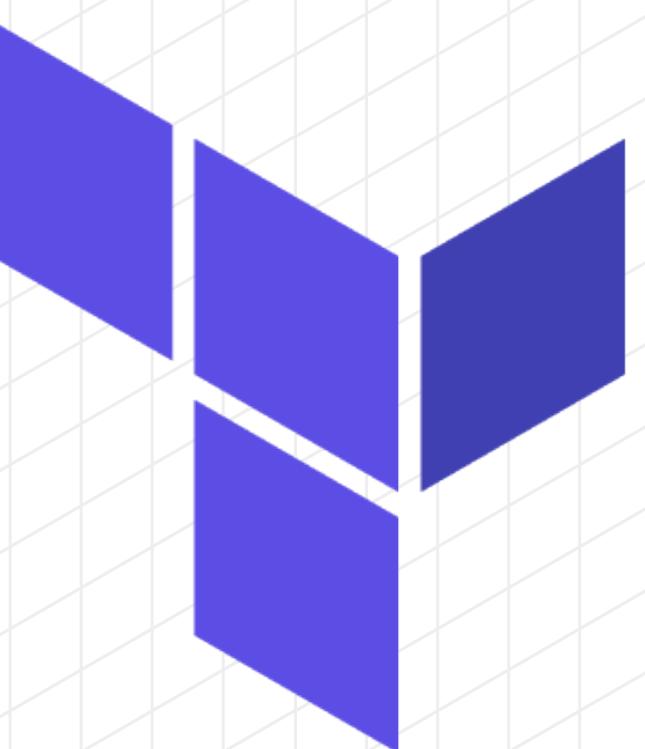
github.com/terraform-aws-modules

registry.terraform.io/modules/terraform-aws-modules



7+ million provisions, 1000+ pull-requests and issues resolved...

Become a sponsor – github.com/sponsors/antonbabenko



HashiCorp

Terraform

Write, plan and manage infrastructure as code

www.terraform.io

nic

@antonbabenko

Infrastructure as code makes DevOps possible

Key benefits:

- Treat infrastructure like application code
- Always know what changed
- Validate infrastructure before deployment

101 shades of infrastructure

- as powerpoint (aka ClickOps)
- as json/yaml files
- as configuration files
- as structured key-values configuration files
- as code

101 shades of infrastructure

- as powerpoint (aka ClickOps) – *infrastructure-2010-fixed.ppt*
- as json/yaml files – AWS CloudFormation, Azure Resource Manager
- as configuration files – Terraform, Azure Resource Manager
- as structured key-values configuration files – Terragrunt
- as code – Pulumi, AWS CDK



"I have an idea!"



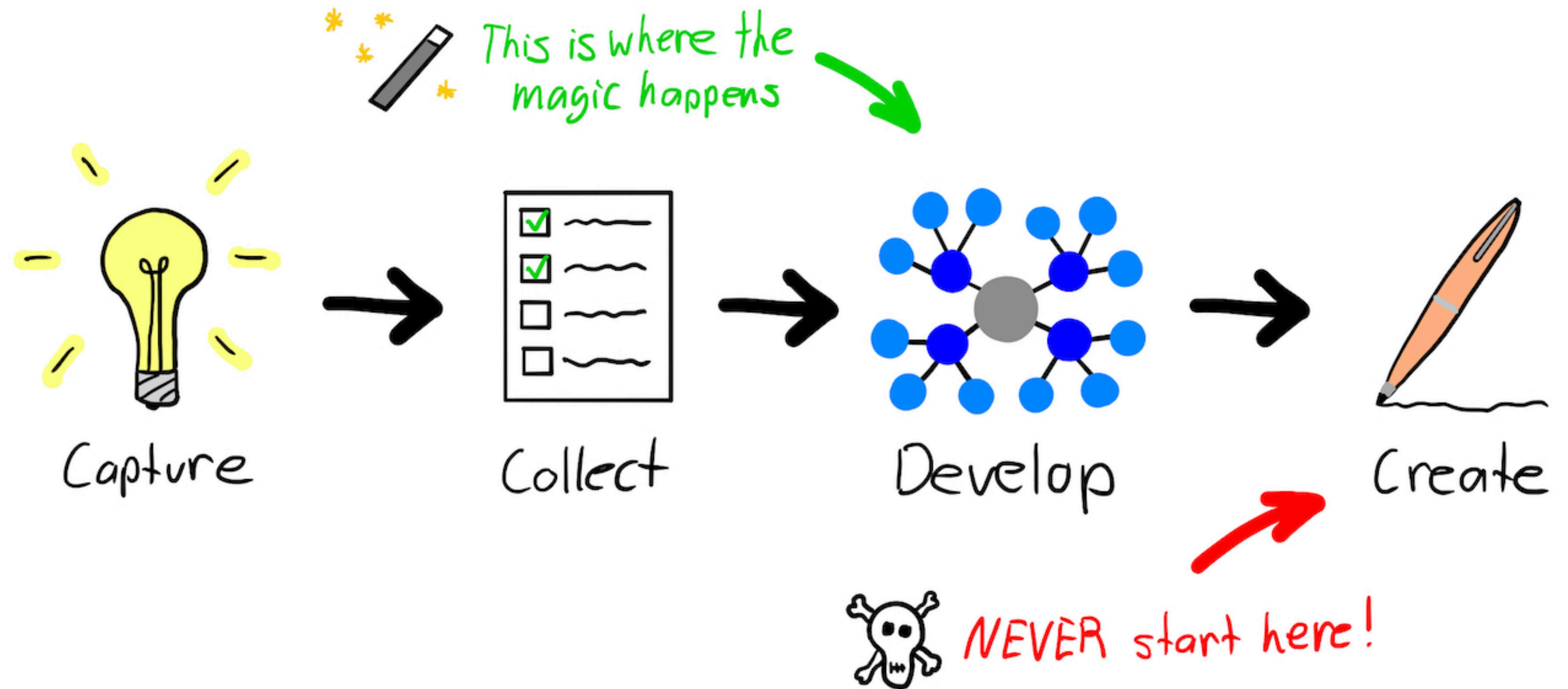
@antonbabenko



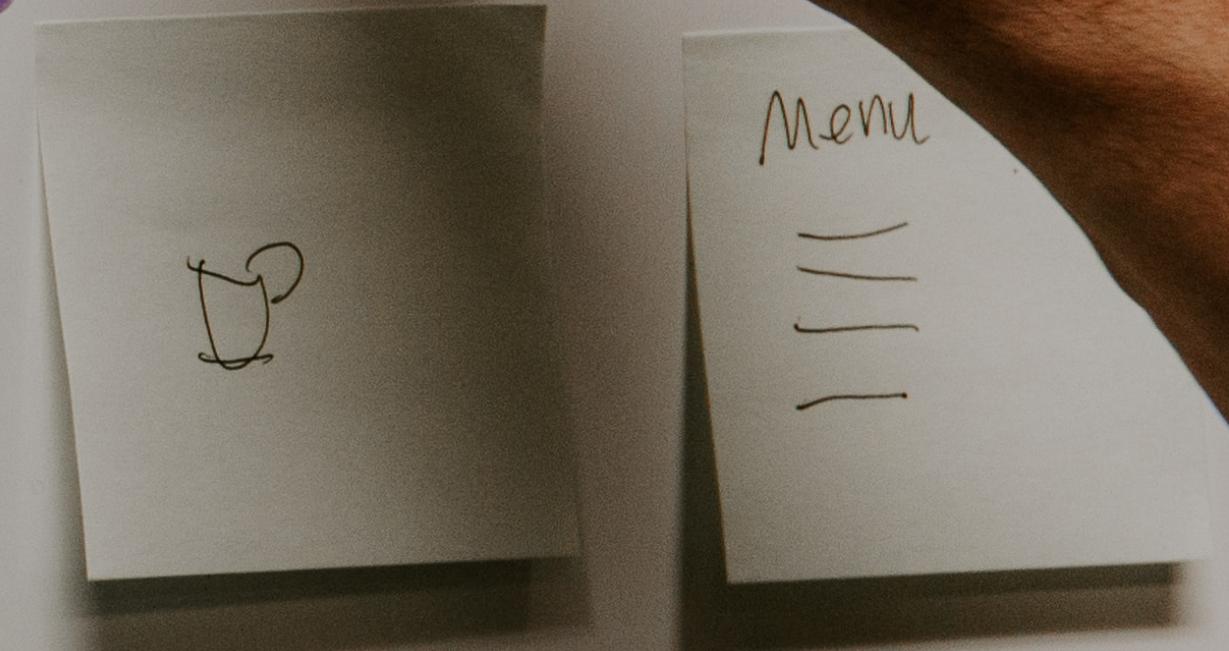
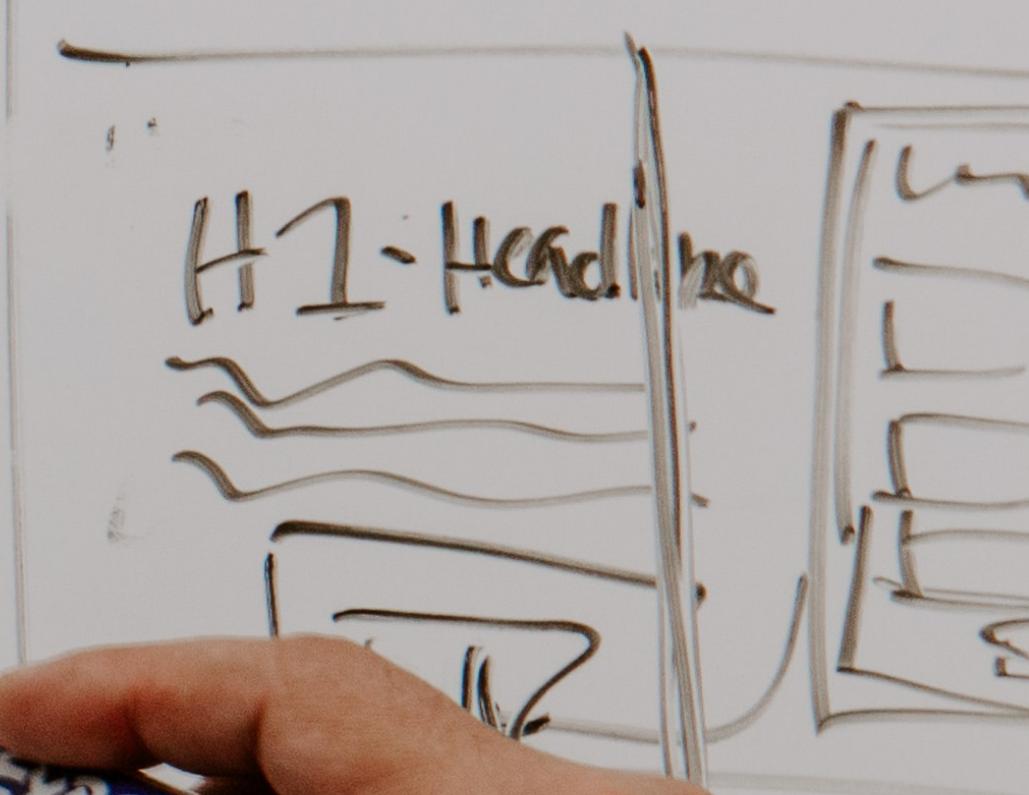
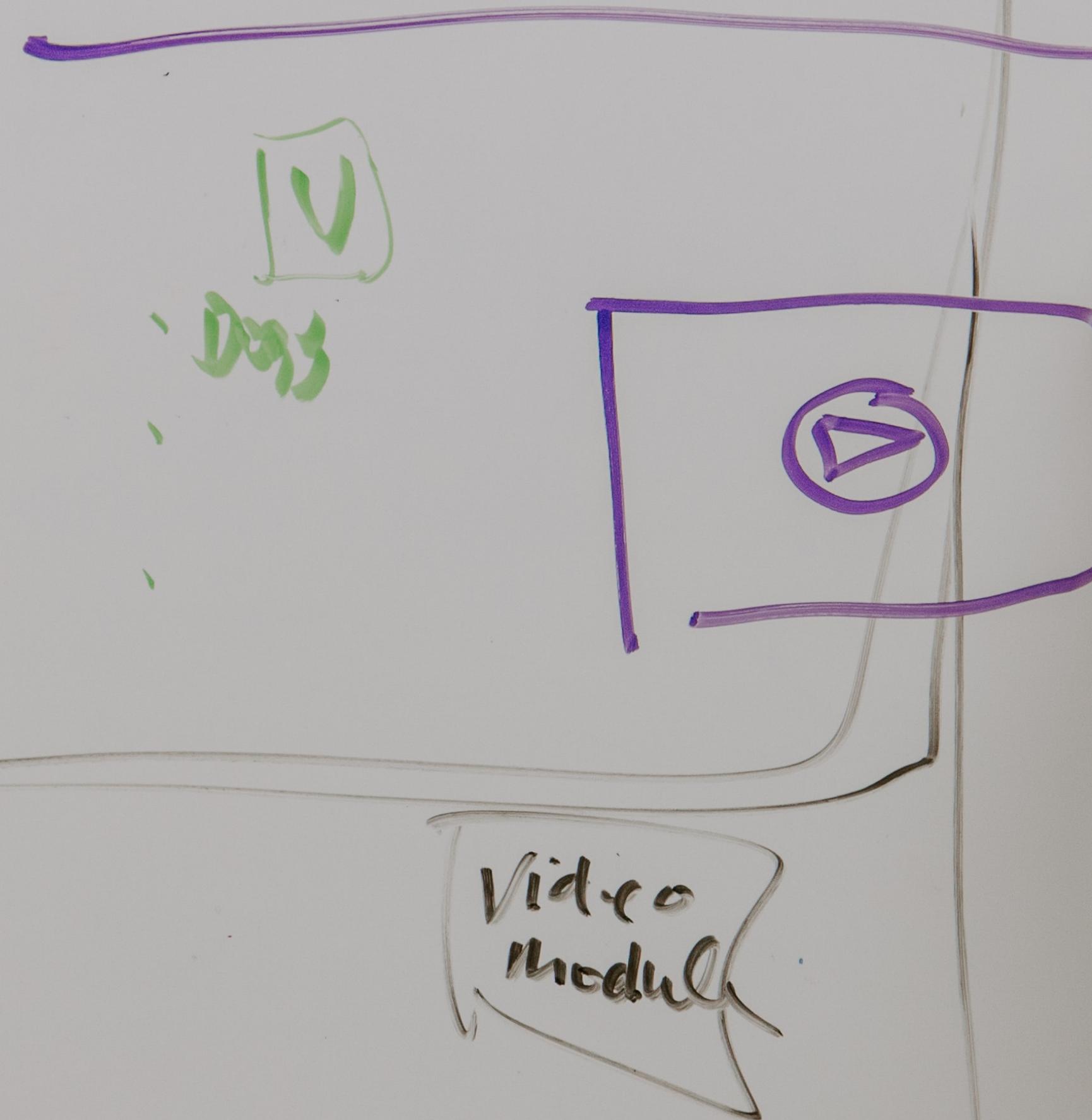
What's next?



@antonbabenko



1. w
2. w
3. w
- w
- w
- w
- w
- w
- w
- w



AWS vs Azure vs GCP? PHP? Go? Serverless?

What is infrastructure?

“Infrastructure is all what is needed to support the computing requirements of an application.”

How about DDD for infrastructure?

Design, Draw and Deploy

Design, Draw, and Deploy your AWS infrastructure



@antonbabenko

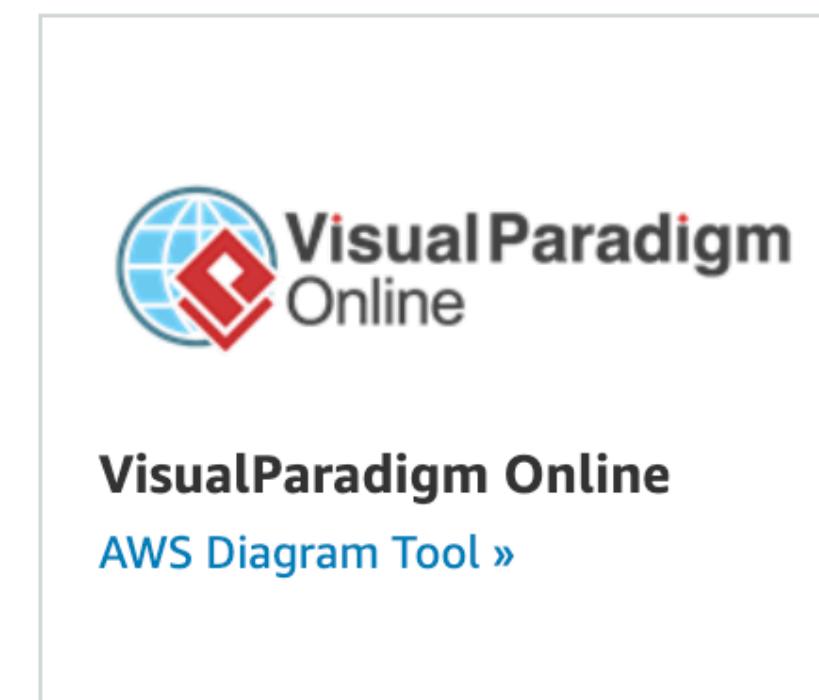
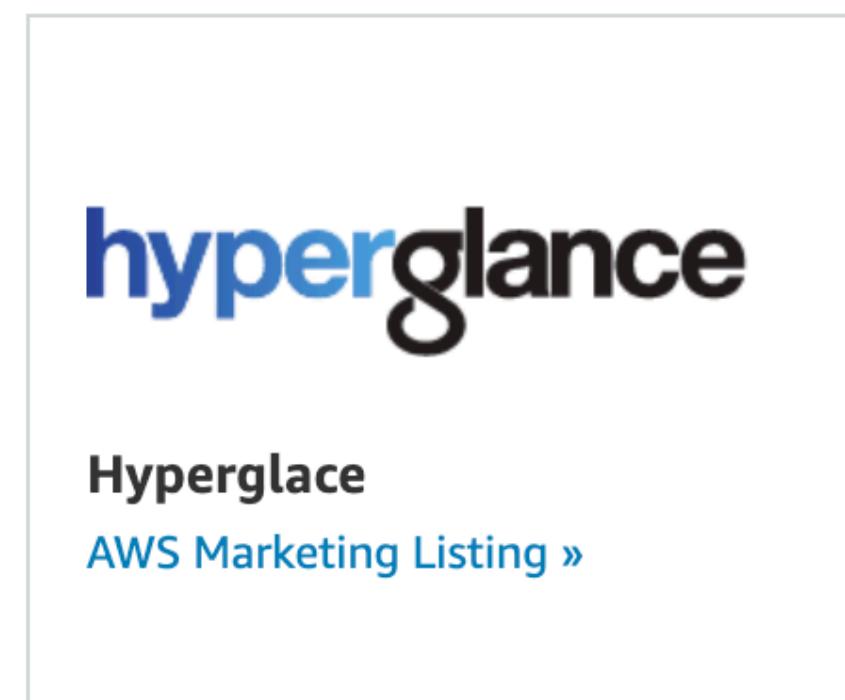
Design your AWS infrastructure

- ✓ AWS Well-Architected Framework – <https://wa.aws.amazon.com/>
- ✓ AWS Architecture Center – <https://aws.amazon.com/architecture/>
- ✓ AWS Whitepapers & Guides – <https://aws.amazon.com/whitepapers/>



@antonbabenko

Draw your AWS infrastructure





Current blueprint

Web App Reference Architecture



WHAT'S NEW



DESIGN LIVE BUDGET



AWS Components

Common

BLOCK

TEXT LABEL

ICON

IMAGE

AREA

Compute

AUTO SCALING

Auto scale EC2 group capacity

EC2

Elastic virtual server

LAMBDA

Run code in response to events

ZONE

Availability zone

Storage

EBS

EC2 block storage volume

S3

Simple object storage

EFS

Elastic file system

CLOUDFRONT

Content delivery network

Networking

ELB

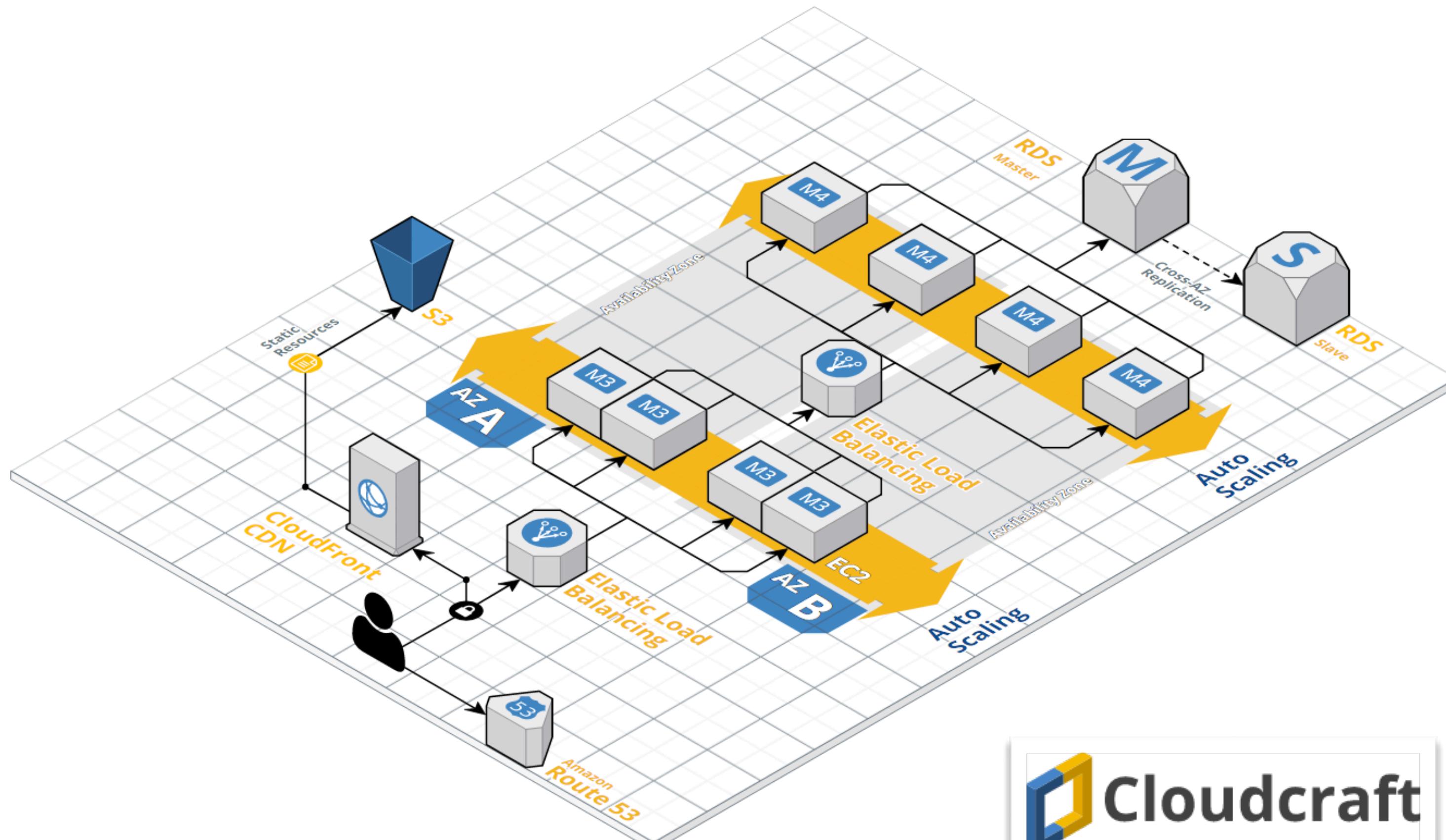
Elastic load balancer

ROUTE 53

Managed DNS service

VPC GATEWAY

Access instances in your VPC



Drag to create multi-selection

Source: cloudcraft.co – the best way to draw AWS diagrams

@antonbabenko

Your AWS infrastructure

- Design
- Draw
- Deploy – What to do with deployment?

Requirements & Preferences

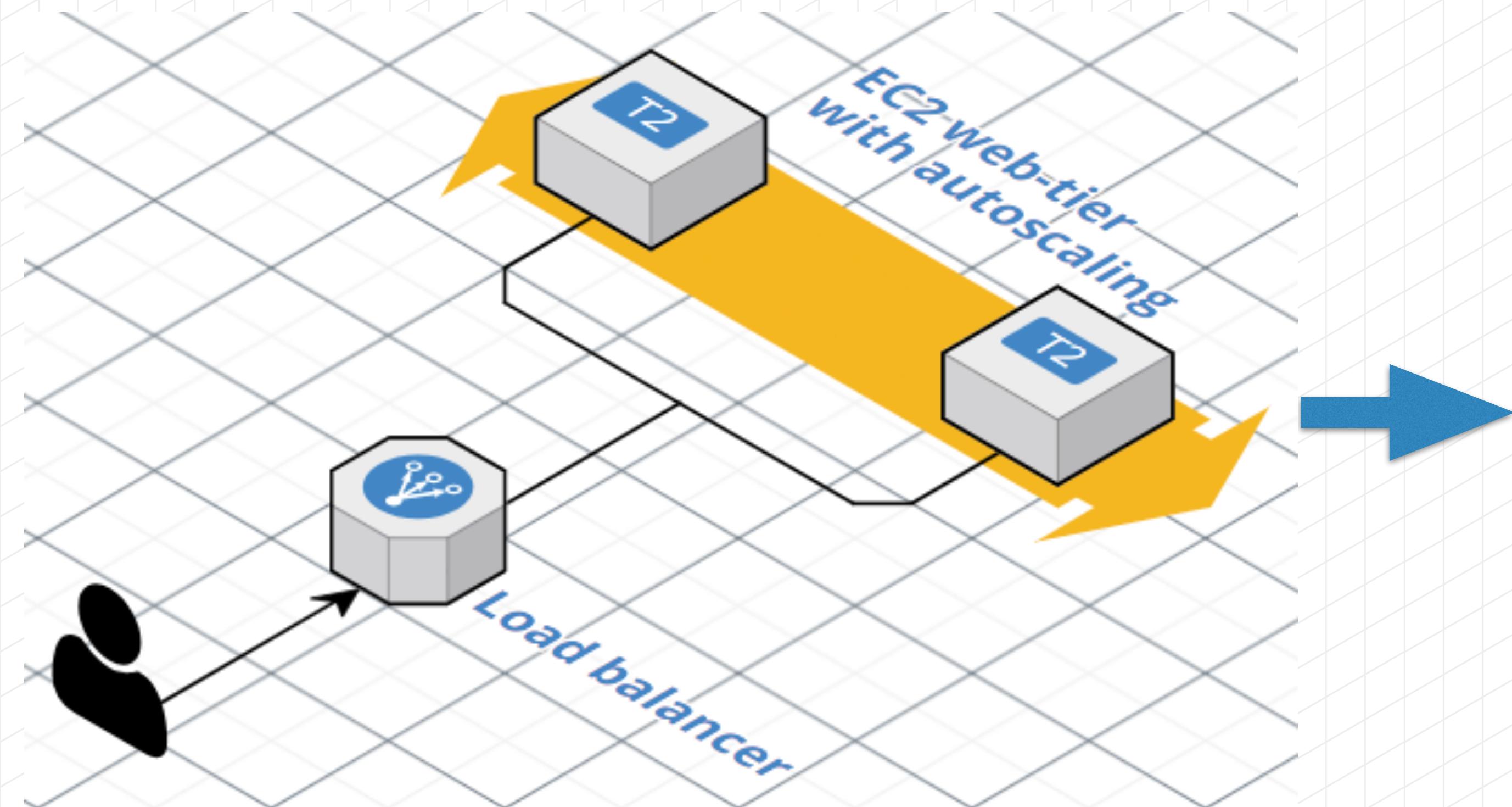
- Infrastructure as code
- Reproducible
- Automation
- Proven technologies
- Active community
- Available commercial support
- Follow best-practices
- Reduce copy-paste

What should you care about?

- Declarative and/or imperative approach
- Push vs pull model
- Lines of code
- Testability
- Programming language
- OOP or DSL
- Popularity

What should you **REALLY** care about?

- Y Declarative and/or imperative approach
 - Y Push vs pull model
 - Y Lines of code
 - Y Testability
 - Y Programming language
 - Y OOP or DSL
 - Y Popularity
- Simplicity**
- Velocity**
- Maintainability**



```

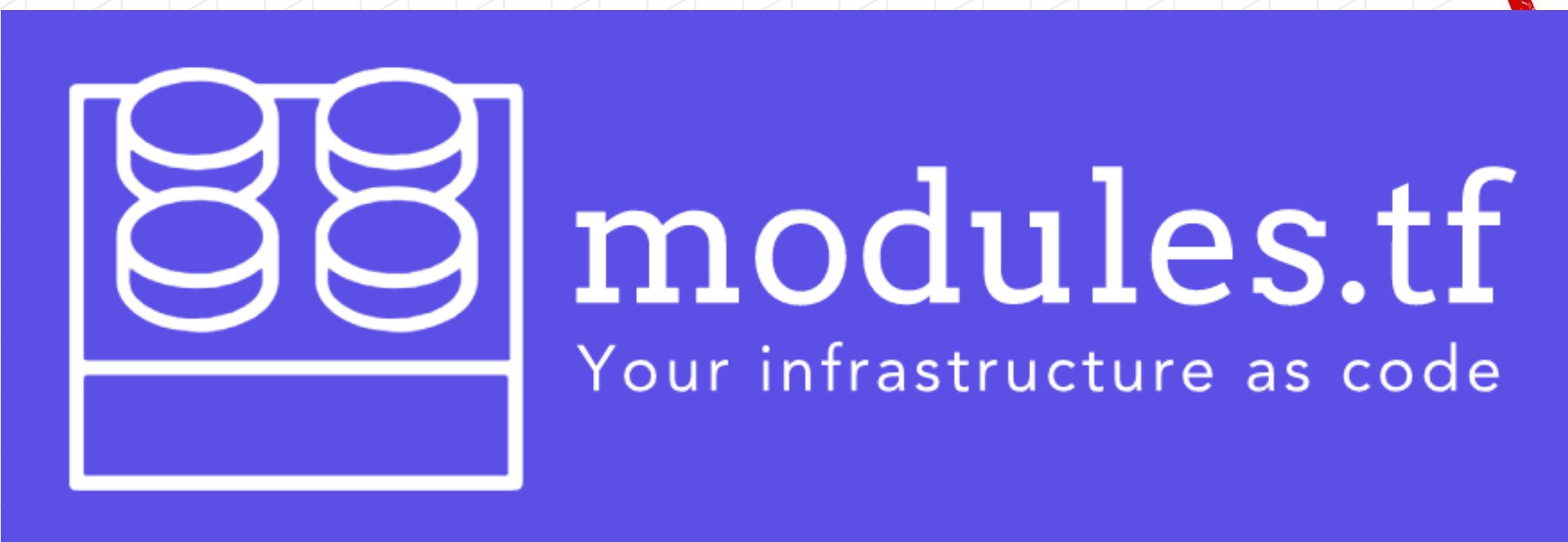
1 variable "aws_region" {
2   description = "Region where resources should be created"
3   default      = "eu-west-1"
4 }
5
6 provider "aws" {
7   region = var.aws_region
8 }
9
10 resource "aws_s3_bucket" "this" {
11   bucket = "my-bucket-${random_pet.bucket.id}"
12 }
13
14 resource "random_pet" "bucket" {
15   keepers = {
16     aws_region = var.aws_region
17   }
18
19   length = 1
20 }
21
22 output "this_s3_bucket_id" {
23   description = "ID of S3 bucket"
24   value       = aws_s3_bucket.this.id
25 }

```



@antonbabenko

100% Opensource



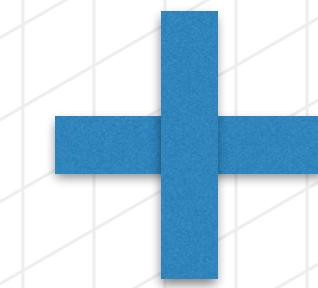
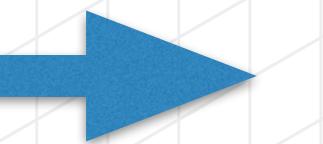
100% FREE

Infrastructure as code generator – from visual diagrams to Terraform

Source code – <https://github.com/antonbabenko/modules.tf-lambda>

Generated infrastructure code – <https://github.com/antonbabenko/modules.tf-demo>

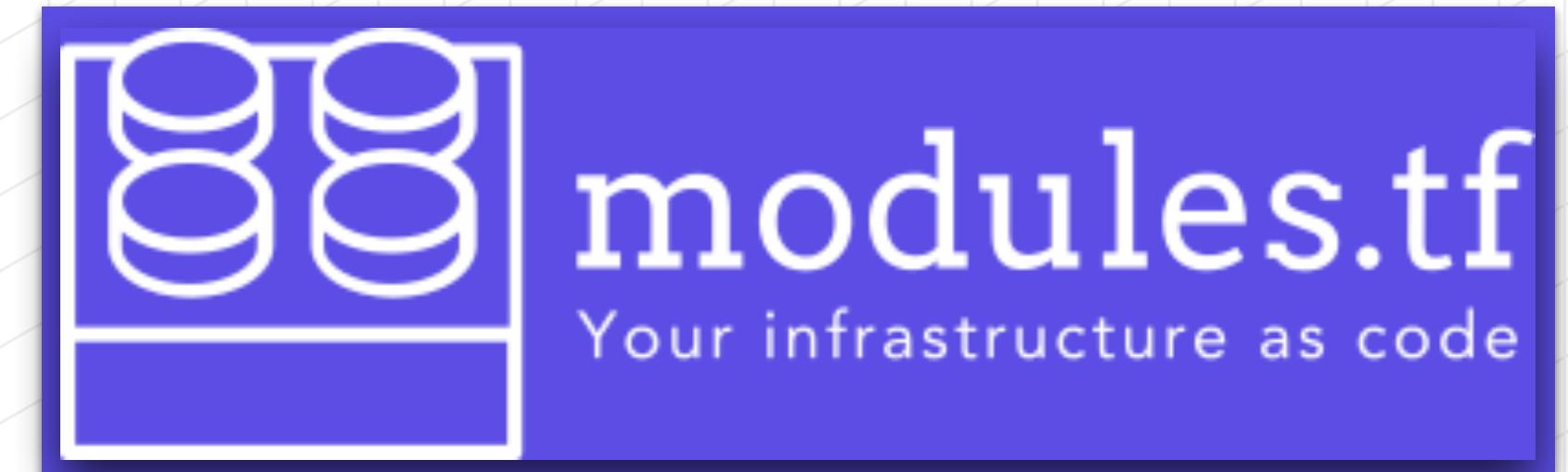
@antonbabenko



- ✓ [cloudcraft.co](#) – design, plan and visualize
- ✓ [terraform-aws-modules](#) – building blocks of AWS infrastructure
- ✓ [Terraform](#) – manage infrastructure as code
- ✓ [Terragrunt](#) – orchestrate Terraform configurations

Demo

- ✓ A picture's worth a thousand words
- ✓ Design, draw and deploy infrastructure as code



nic

@antonbabenko

modules.tf – generated code

- ✓ Potentially ready-to-use Terraform configurations
- ✓ Suits best for bootstrapping
- ✓ Enforces Terraform best-practices and segregation of duties between Terraform developers and users
- ✓ Batteries included (terraform-aws-modules, terragrunt, pre-commit-terraform)
- ✓ 100% free and open-source (<https://github.com/antonbabenko/modules.tf-lambda>)
- ✓ Released under MIT license



@antonbabenko

Thanks!

github.com/antonbabenko

twitter.com/antonbabenko

antonbabenko.com

nic