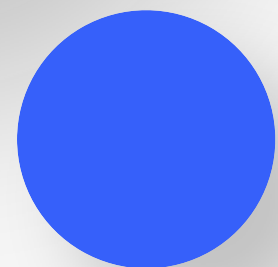




PROJECT AWW



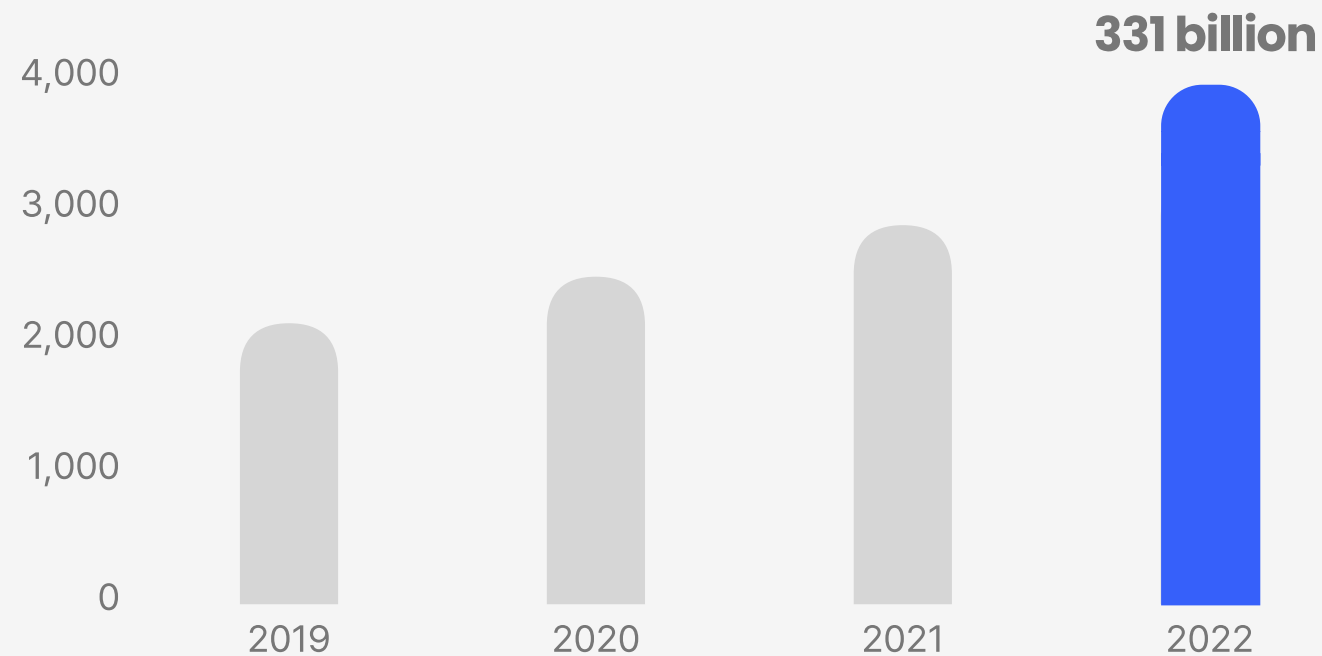
laCDOG



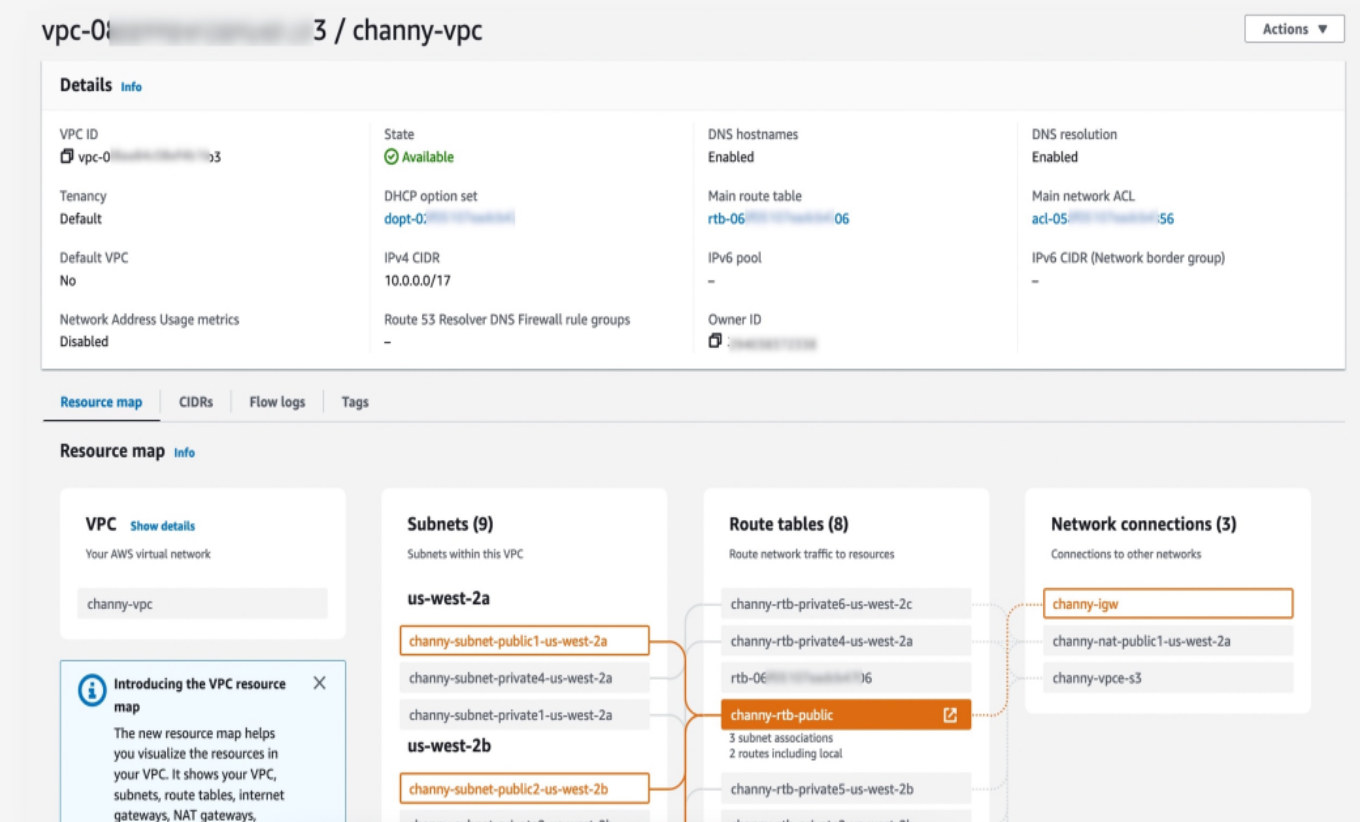
laC 기반 인프라 시각화 서비스 & 패키지

Project Motive

프로젝트 수행 동기



전세계 클라우드 매출 통계(Gatner)



Background

현재 클라우드는 누구나 다 사용하는 기술로 자리잡았으며, 고도화됨에 따라 인프라는 복잡해 지는 추세이다. 이러한 고도화된 인프라를 콘솔로 관리하는데는 설계 및 반복작업에 여러 어려움이 존재한다.



Project Motive

프로젝트 수행 동기

Rank
1위



2022 Github
오픈소스 트렌드

36% 성장

Terraform
39.5K



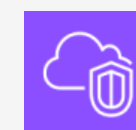
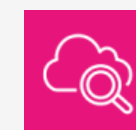
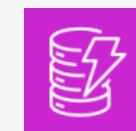
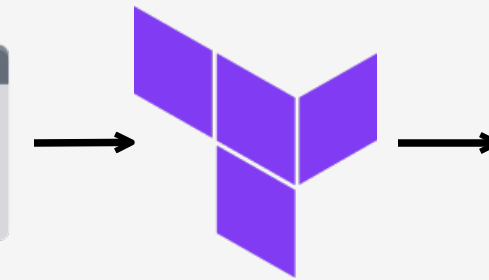
laC 관련 툴의
많은 star 수 존재

135.1k



Effect

복잡한 구성을 해결하기 위해 laC(Infrastructure as a Code)로 여러 구성사항을 코드로 정의한 것을 사용해 버전관리, 자동화, 협업, 재사용에 큰 이점을 보이고 있다.





Project Motive •

프로젝트 수행 동기

laC 도입이 적은 이유

선언형 언어라는 특징으로 코드만 보고
인프라의 구성을 이해하기 어렵다.
인프라의 전체적인 형태, 네트워크 흐름, 권한 관리를
파악하기 어려워 laC 구성상 오류가 많이 발생한다.

Key Keywords

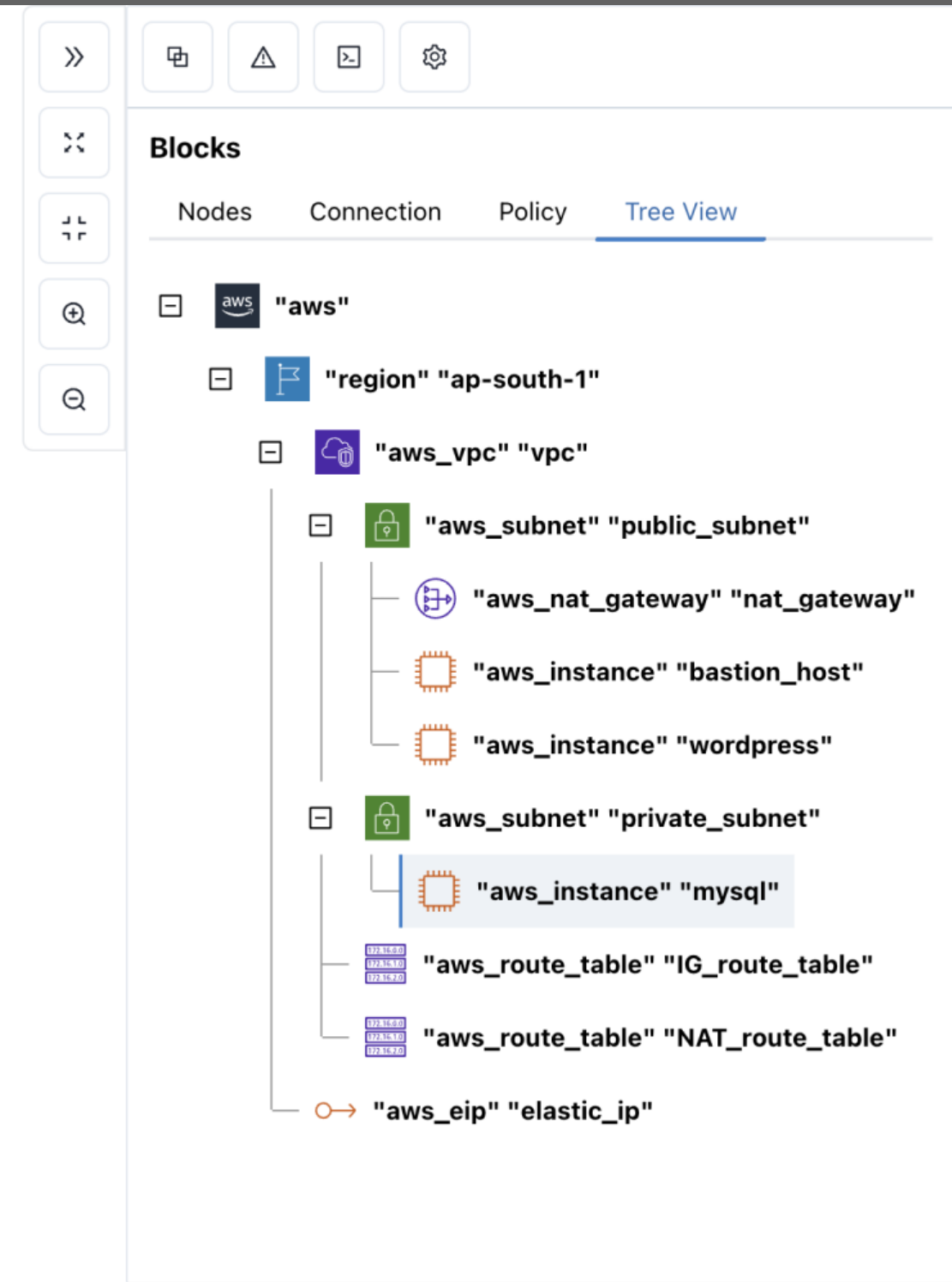
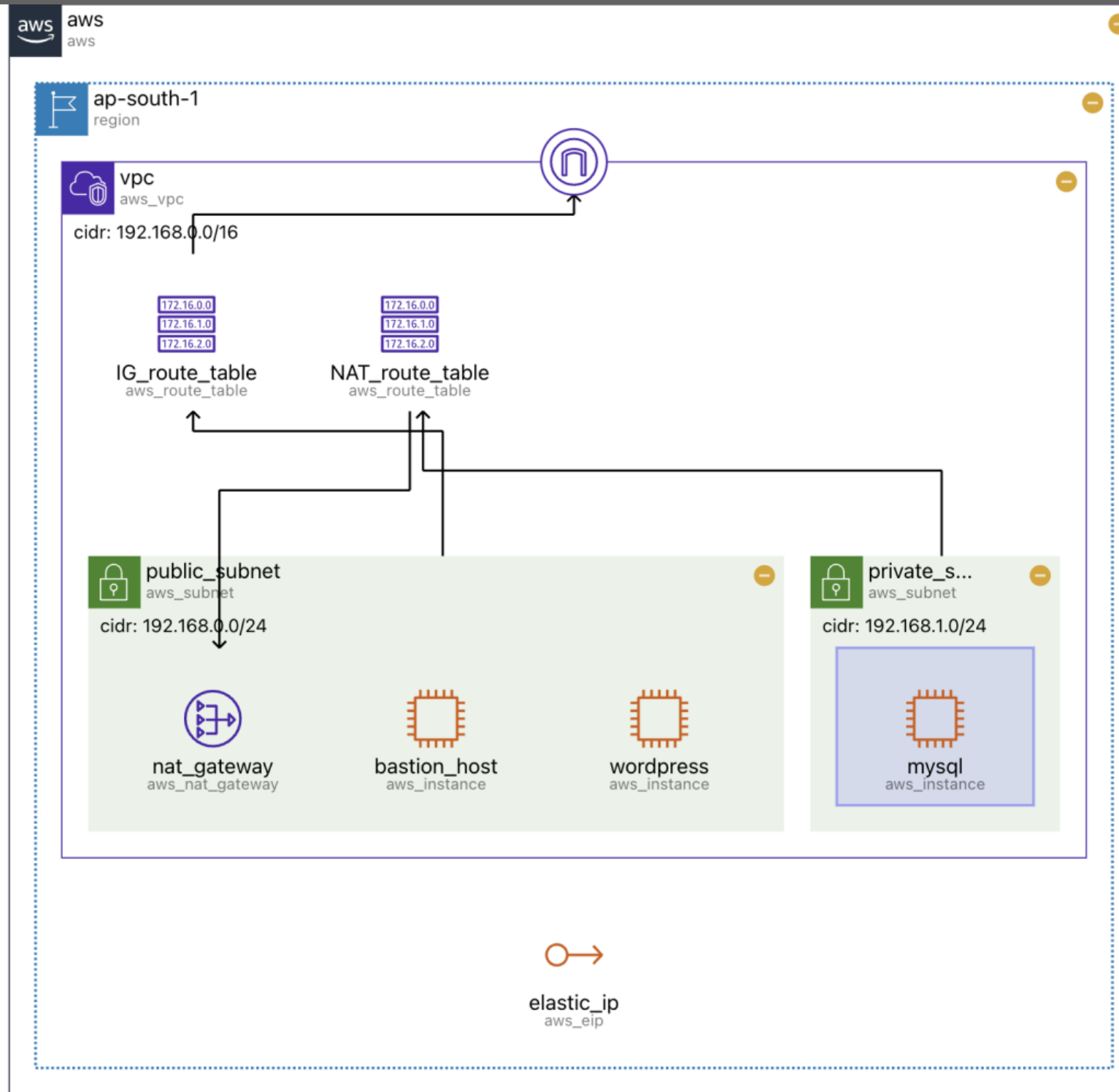
기존 laC 시각화 솔루션의 한계

Terraform의 기본 기능인 graph 기능을 바탕으로 하는
의존성 시각화, 실무에서 자주 사용되지 않는
드래그 앤 드롭 형식의 시각화 서비스만 존재
plan 데이터만을 이용, 느리고 보안에 취약



Conclusion

laC만을 이용한 인프라 시각화 제공으로, 사용자가 조금 더 쉽게 laC를 이용해
클라우드 환경을 구성할 수 있도록 한다.



laC 작성 중 빠르게 인프라를 확인할 수 있는 시각화 여러가지 기능 제공



Solution

프로젝트 내 해결사항

- ☒ aws
- ☒ ncp
- ☒ multi

파일 업로드

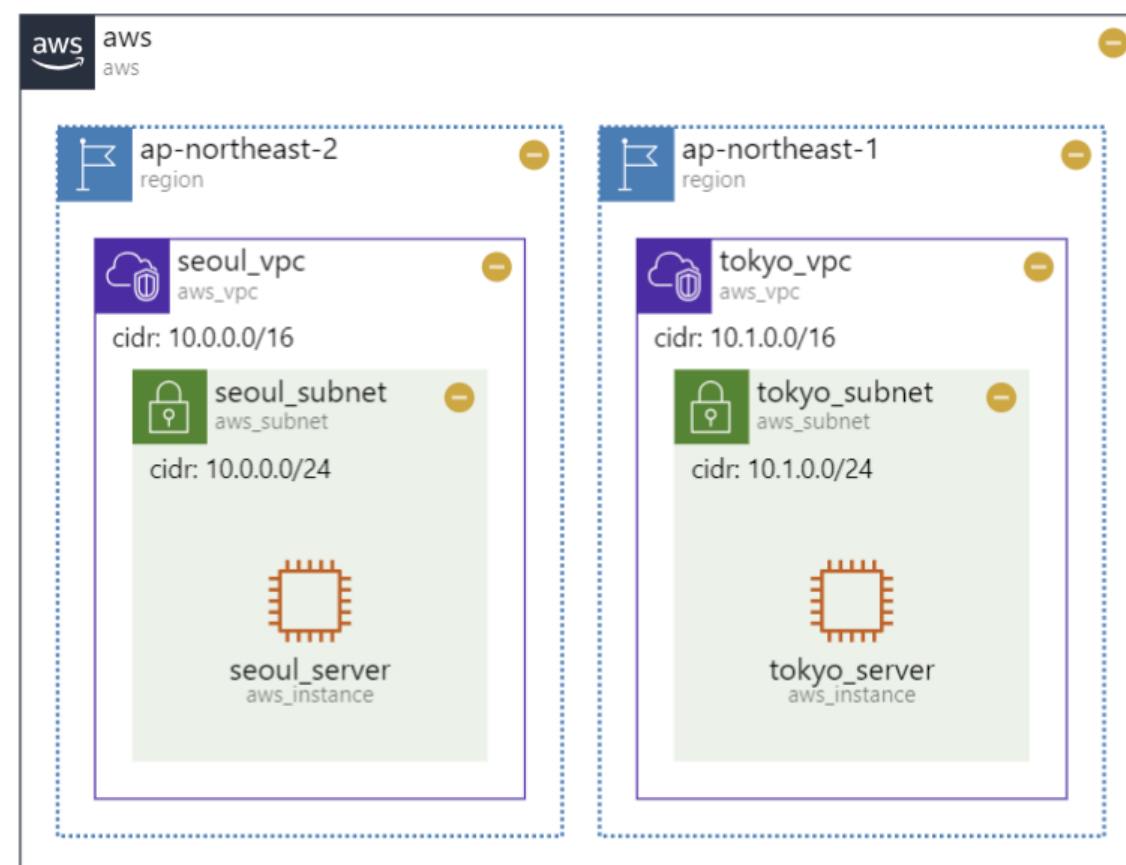
aws_2tier.tf

aws_region.tf

aws_s3.tf

ncloud_vpc.tf

multi_cloud.tf



Blocks

Nodes

Connection

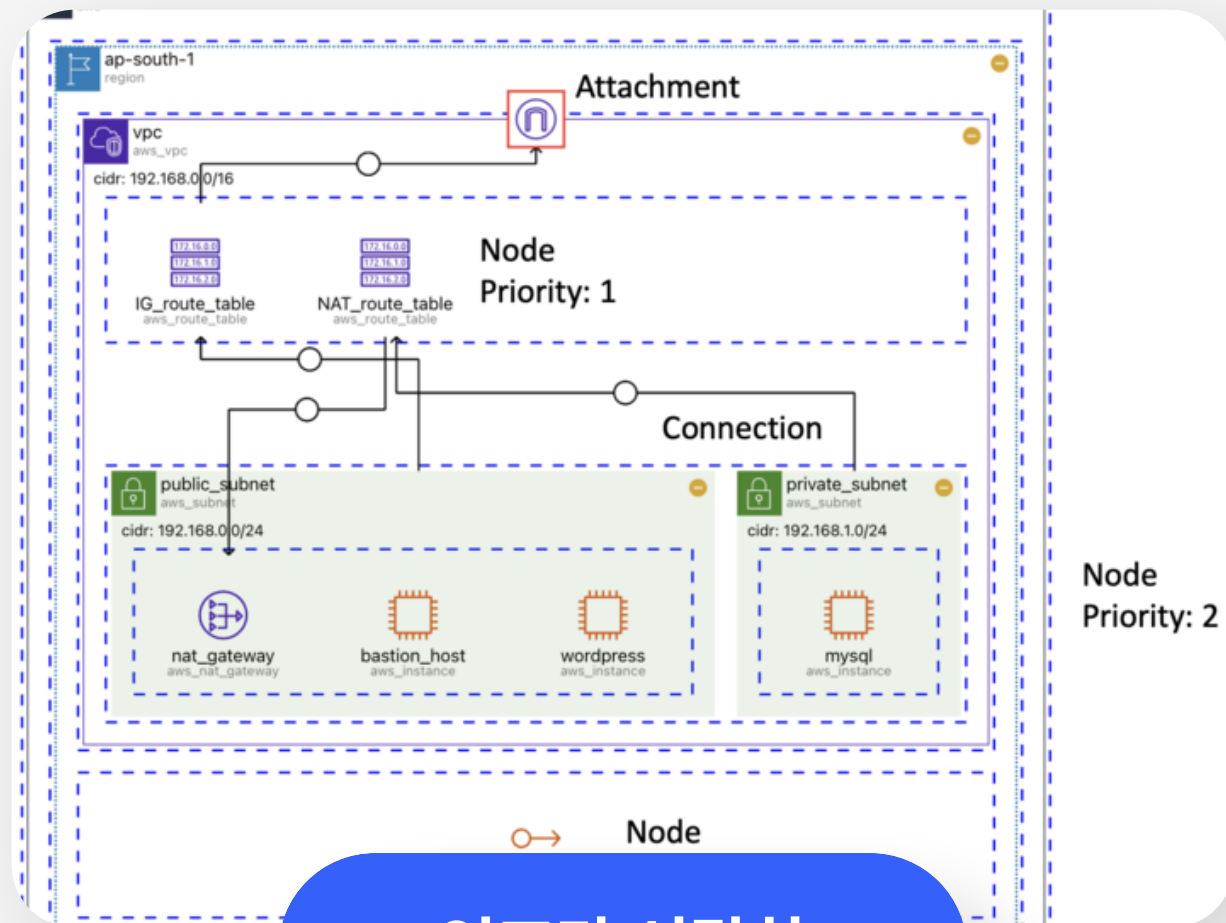
Policy

Tree View

- aws "aws" ✓
- region "ap-northeast-1" ✓
- aws_vpc "tokyo_vpc" ✓
- aws_subnet "tokyo_subnet" ✓
- aws_instance "tokyo_server" ✓
- region "ap-northeast-2" ✓
- aws_vpc "seoul_vpc" ✓
- aws_subnet "seoul_subnet" ✓
- aws_instance "seoul_server" ✓

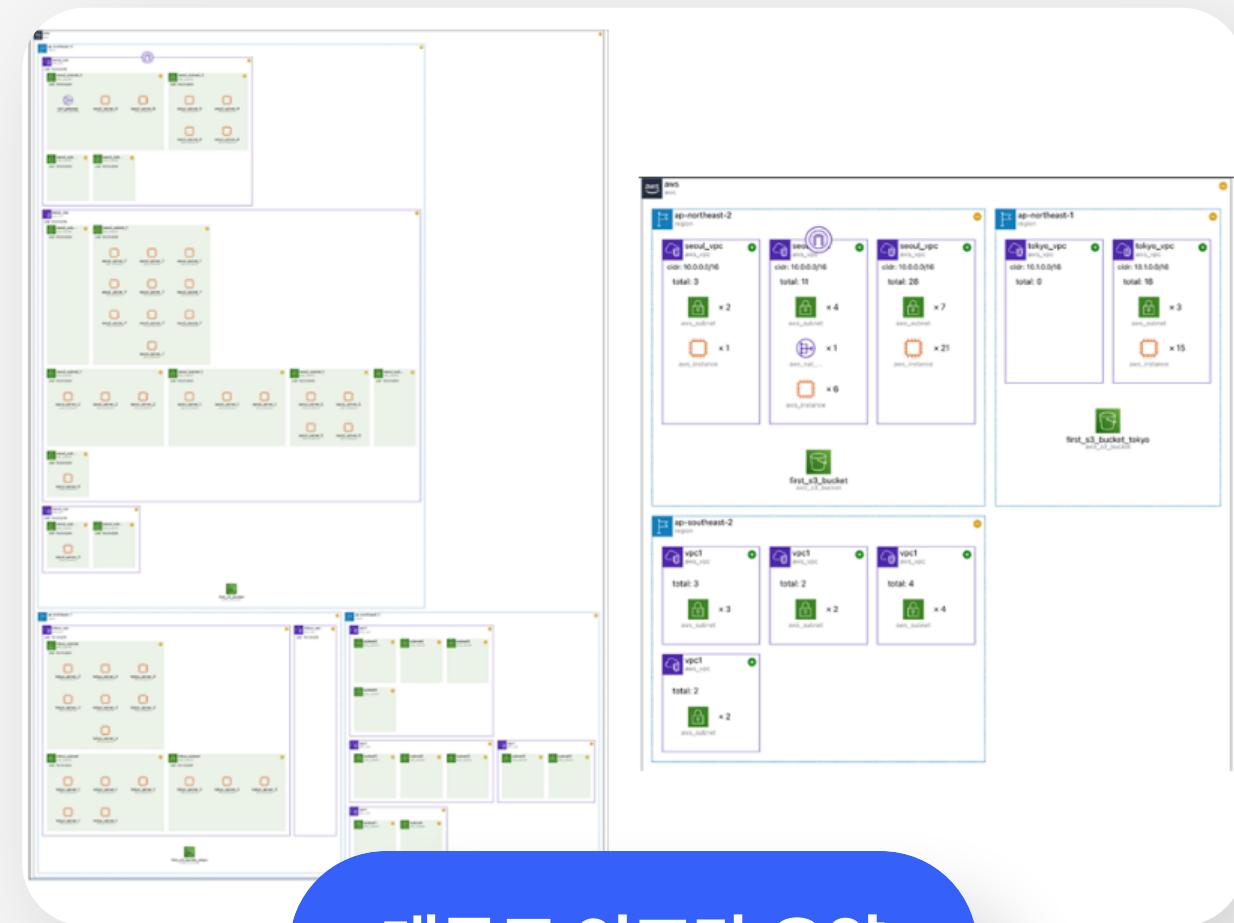
Function

프로젝트 제공 기능 : 인프라 시각화 및 요약



인프라 시각화

네트워크 흐름, 연결정보를 토대로 시각화를 처리함으로써
리소스 추가시 인프라 시각화 변동 및 영향 파악 가능



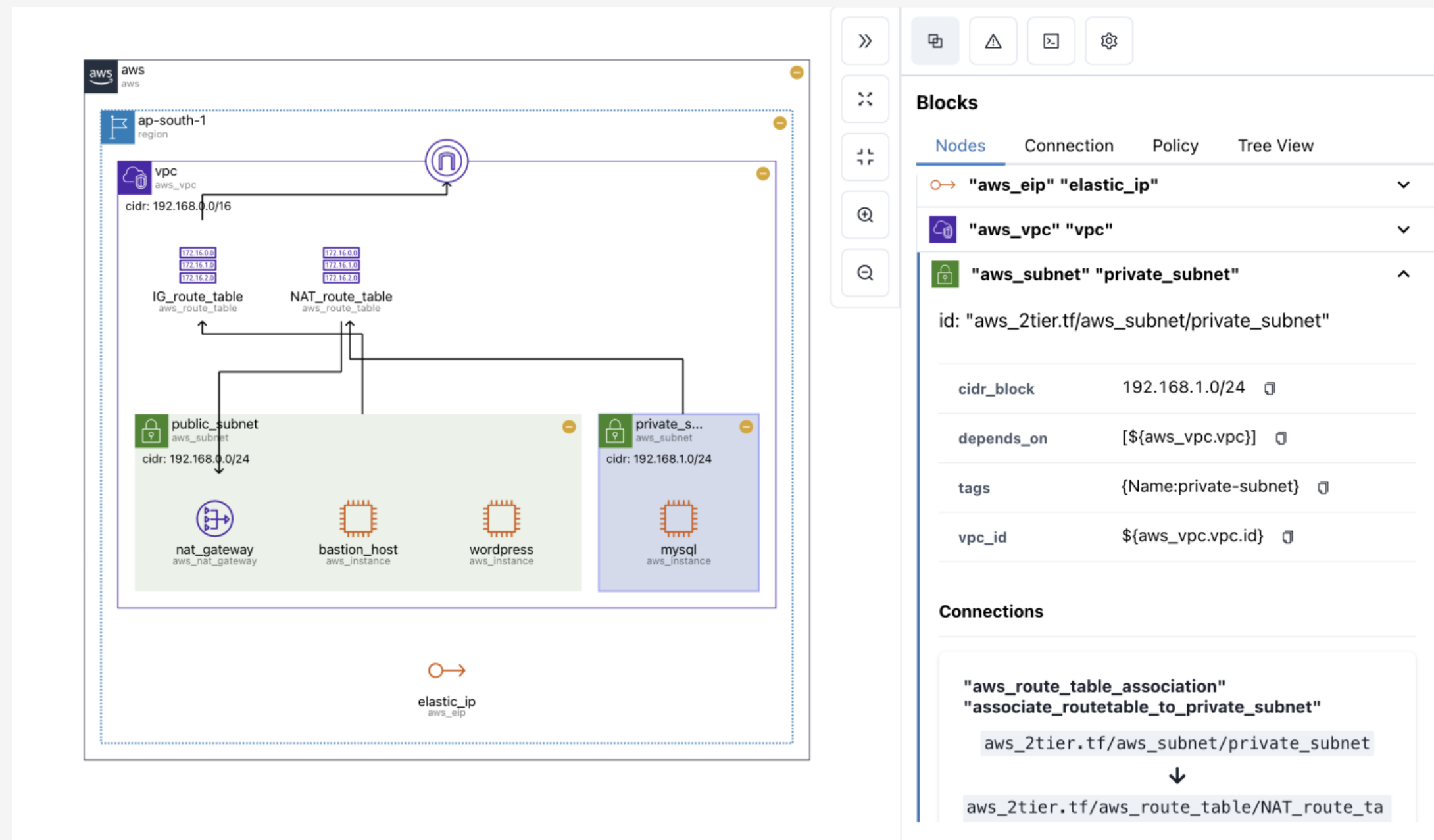
대규모 인프라 요약

현업의 대규모 인프라의 리소스가 난잡하게 흩어지지
않을 수 있도록 인프라 요약 기능 제공



Function

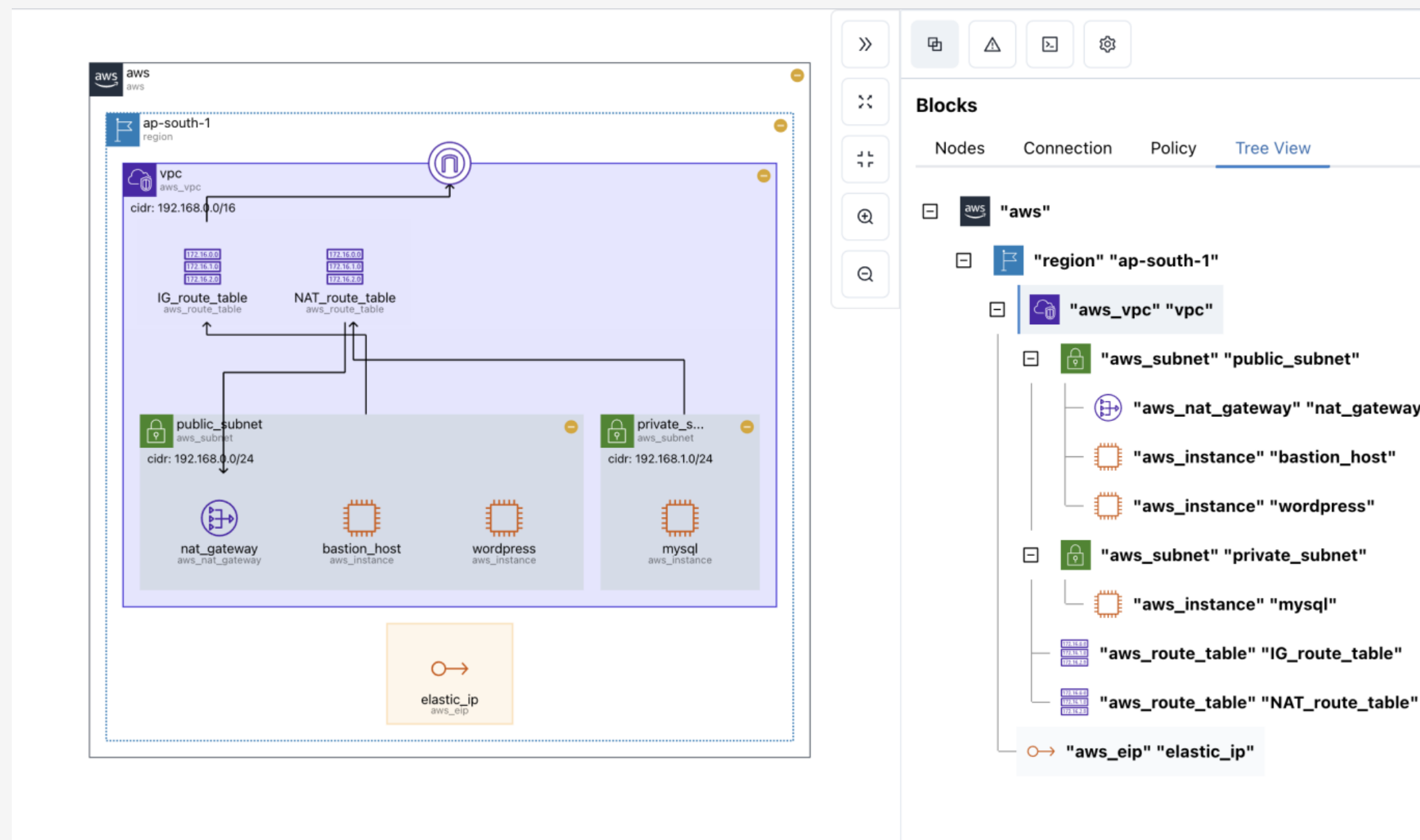
프로젝트 제공 기능 : 사이드바





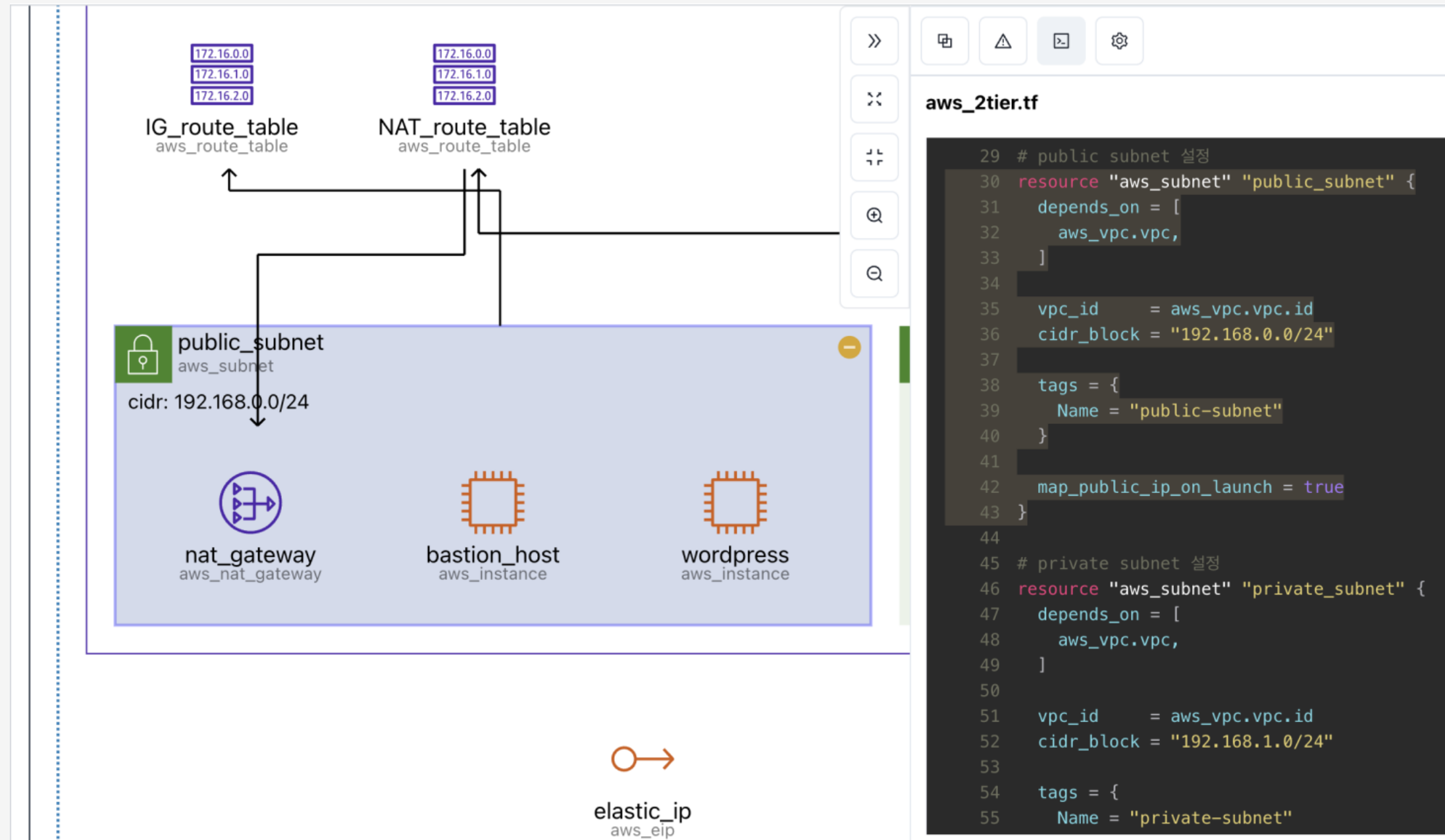
Function

프로젝트 제공 기능 : 사이드바



Function

프로젝트 제공 기능 : 사이드바





Function

프로젝트 제공 기능 : 패키지

v Search Packages	
@aww-vzc/container	
시각화 컴포넌트를 감싸 에러, 로딩등을 처리하는 컴포넌트입니다.	
Anonymous	v0.1.73 Published on 12/18/2023 9:00:30 PM · 8 days ago
@aww-vzc/errors	
Anonymous	
v0.1.20	Published on 12/18/2023 9:00:29 PM · 8 days ago
@aww-vzc/icons-data	
Anonymous	
v0.1.23	Published on 12/13/2023 3:48:52 AM · 13 days ago
@aww-vzc/locales	
Anonymous	
v0.1.56	Published on 12/18/2023 9:00:29 PM · 8 days ago
@aww-vzc/react	
코어로직과 함께 리액트 환경에서 필요한 컴포넌트, 훅, 컨텍스트를 제공합니다.	
Anonymous	v0.1.94 Published on 12/18/2023 9:00:30 PM · 8 days ago
@aww-vzc/react-common	
트리 구조 표현을 위한 컴포넌트입니다.	
Anonymous	v0.1.3 Published on 12/13/2023 4:26:30 AM · 13 days ago

오픈소스 개발

private npm 레지스트리에 배포
시각화 서비스가 다른 여러 서비스에서 사용 가능

세부 패키지 처리

레이아웃 계산 시뮬레이션, 시각화 데이터 UI 렌더러, 사이드바
기능을 세부 패키지로 처리하여 원하는 패키지만 사용 가능

사용성 용이

React 기반 컴포넌트 뿐만 아니라 바닐라 JS에서 또한 사용
가능하여 추후에는 웹어셈블리와 같은 고성능 포팅 가능

2.3 resource 블록 기초

6/8

여러 리소스에 대해서 한번 익혀보도록 하겠습니다.

VPC는 AWS의 리소스로, 논리적으로 격리된 사용자의 네트워크를 구축할 수 있는 서비스입니다.

보통 VPC를 구축하고, 내부에 subnet과 같은 리소스들을 통해 네트워크를 또 다시 분리하여 인프라를 구성할 수 있습니다.

aws의 경우 기본적으로 컴퓨팅 자원인 EC2를 이용하려면, VPC와 subnet이 구성되어야합니다.

해당 코드를 편집기 아래에 붙여넣은 뒤, 인프라 구조를 확인해 봅시다.

```
resource "aws_subnet" "seoul_subnet" {
  vpc_id      = aws_vpc.seoul_vpc.id
  cidr_block  = "10.0.0.0/24"
  availability_zone = "ap-northeast-2a"
}

resource "aws_instance" "seoul_server" {
  subnet_id    = aws_subnet.seoul_subnet.id
  ami          = "ami-0c9c942bd7bf113a2"
  instance_type = "t2.micro"
}
```

Seoul region의 `aws_vpc` 에 `aws_subnet` 과 `aws_instance` 가 생성된 것을 확인해 볼 수 있습니다.

`aws_subnet` 의 경우 어떤 vpc 내부의 subnet을 생성할 것인지 결정하는 `vpc_id` 라는 인수와, cidr을 설정할 수 있는 `cidr_block` 을 인수로 가지고 있습니다.

`aws_instance` 의 경우 이전 페이지에서 확인했던 `ami` , `instance_type` 이외에도 `subnet_id` 라는 인수가 생겼습니다.

해당 인수는 어떤 subnet 대역에 인스턴스를 생성할지 설정하는 인수입니다.

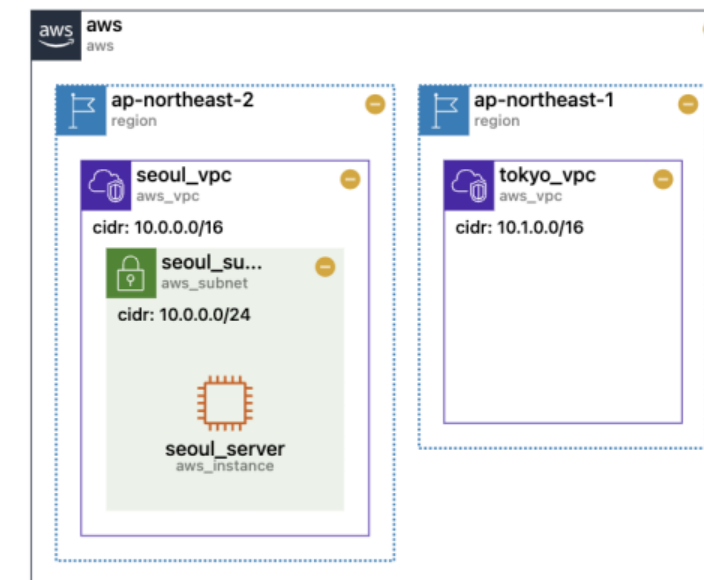
`resource` 블록은 이와 같이 서로의 값을 참조하며 구성사항을 설계할 수도 있습니다.

Previous

Next

```
1 provider "aws" {
2   access_key = "your_access_key"
3   secret_key = "your_secret_key"
4   region     = "ap-northeast-2"
5 }
6
7 provider "aws" {
8   region     = "ap-northeast-1"
9   alias      = "tokyo"
10 }
11
12 resource "aws_vpc" "seoul_vpc" {
13   cidr_block = "10.0.0.0/16"
14 }
15
16 resource "aws_vpc" "tokyo_vpc" {
17   provider = aws.tokyo
18   cidr_block = "10.1.0.0/16"
19 }
20
```

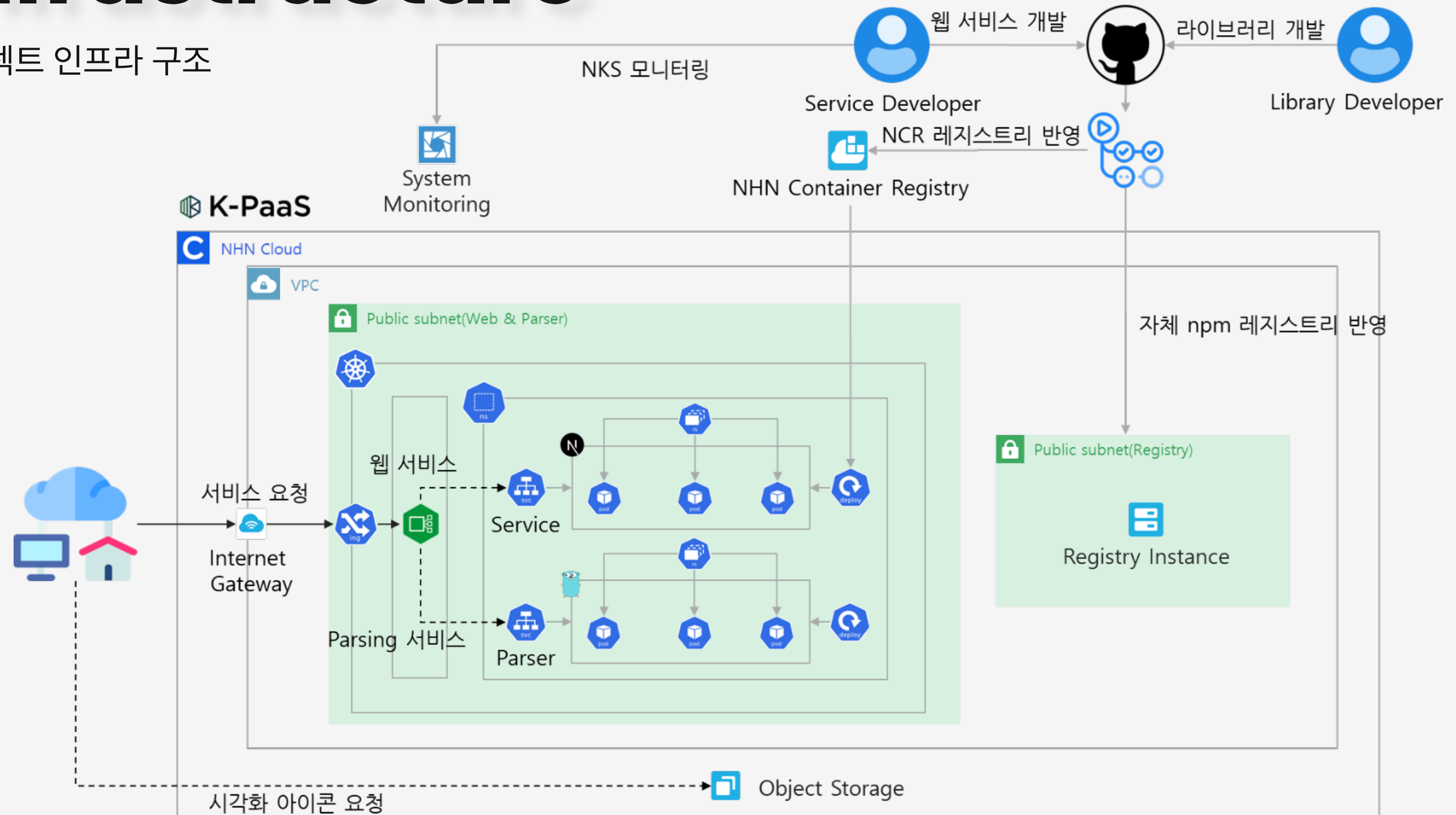
Visualize



laCDOG를 활용한 교육 플랫폼

Infrastructure

프로젝트 인프라 구조





Benefit[●]

프로젝트 기대효과

범용적 사용

IaC에 익숙하지 않은 초보자, IaC를 자주 사용하는 현업자 모두 도움을 주는 것이 가능
초보자의 경우 learning cost를 줄여 인프라에 대한 영향을 빠르게 확인하는 것이 가능



복잡한 구조 시각화

방대한 양의 코드를 분석하며 인프라를 확인하는 것이 아닌, 시각화 모듈을 통해 자원 확인이 가능
거대하고 복잡한 인프라의 변경사항, 서로의 작업이 인프라에 미치는 영향을 빠르게 파악 가능

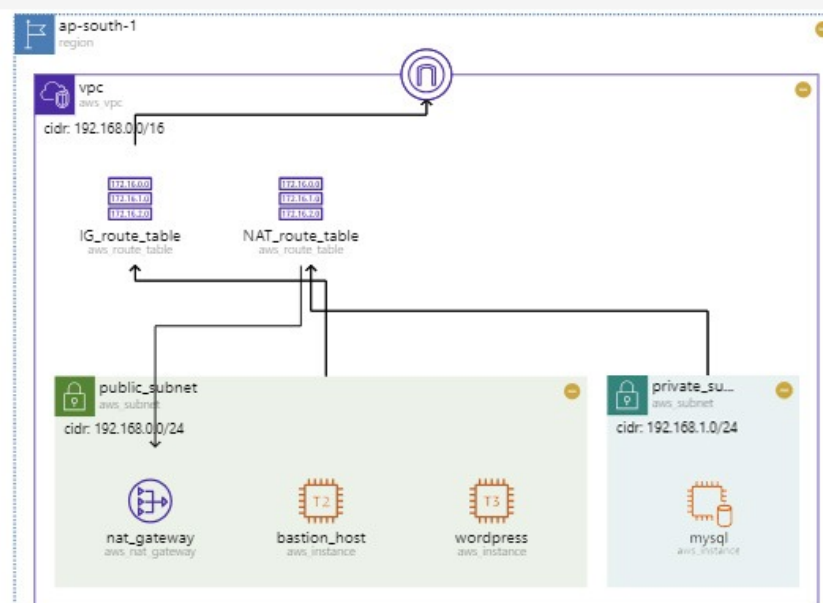


패키지화

단순히 웹서비스로 끝나는 것이 아닌 타 서비스에서도 IaC를 쉽게 사용할 수 있도록 패키지화하여 클라우드 네이티브 생태계에 기여

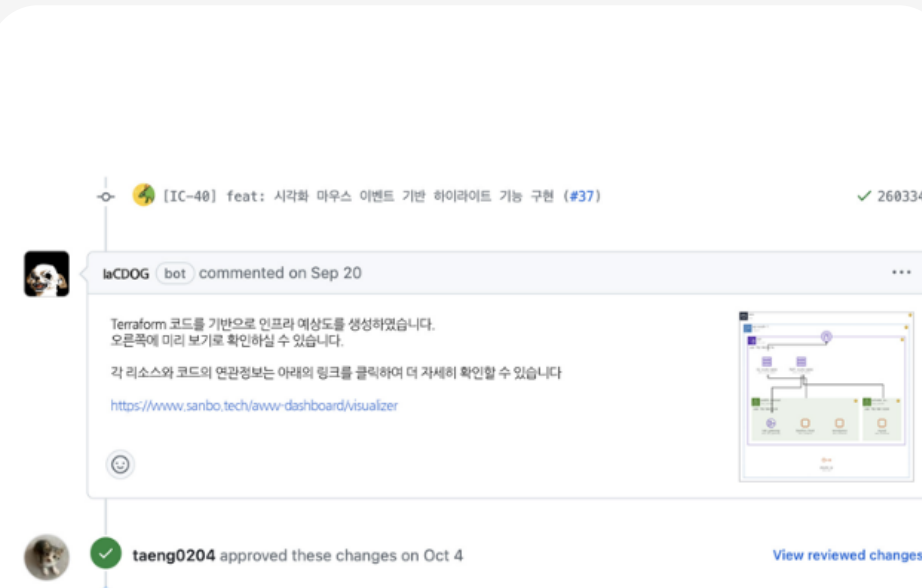
Benefit

프로젝트 기대효과



서비스 고도화

복잡하게 얹히거나 겹치는 커넥션의 개선,
리소스 아이콘 및 public private에
맞추어 고도화 진행



서비스 확장

다양한 서비스에서 사용할 수 있도록 진행한
패키지화를 활용해 여러 서비스에 적용



커뮤니티 형성

각자 IaC를 사용하면서 익힌 클라우드와
노하우 등을 플랫폼으로 쉽게 공유하고 피드백

클라우드 네이티브를 위한 한걸음, IaCDOG

Thank you!

