



멀티클라우드, 글로벌 스케일로 시작하다

클라우드바리스타 커뮤니티 제4차 컨퍼런스

[세션5] CB-Larva : 멀티클라우드 인프라 및 응용을 위한 네트워킹

김 윤 곤

CB-Larva 인큐베이터 리더

카페모카(Café Mocha) 한잔 어떠세요 ?

이번 세션은...

응용/도메인/기관 특화 SW



멀티클라우드 서비스 개방형 인터페이스

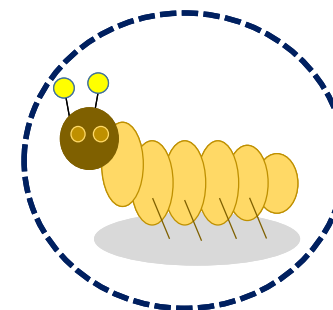
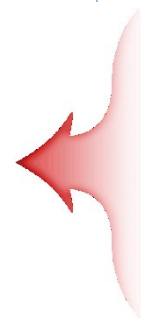
멀티클라우드 애플리케이션
통합관리 프레임워크

멀티클라우드 인프라 서비스
통합 관리 프레임워크

멀티클라우드 인프라 연동
프레임워크

멀티클라우드 통합
모니터링 프레임워크

멀티클라우드 서비스 공통 플랫폼



CB-Larva

목차

I

글로벌 스케일 네트워킹 개요

II

CLADNet: Cloud Adaptive Network

III

CLADNet 기술 시연: 글로벌 스케일 MCIS에 CLADNet 입혀 보기

IV

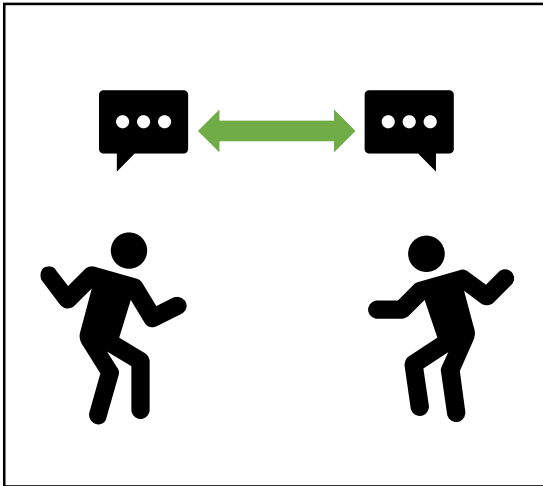
cb-storage, 첫 조각을 맞추다

IV

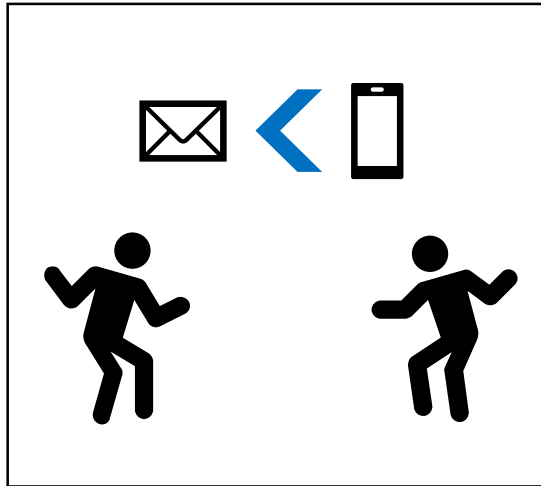
CB-Larva 향후 계획

[Ice-break] 글로벌 스케일 네트워킹에서 중요한 3가지?

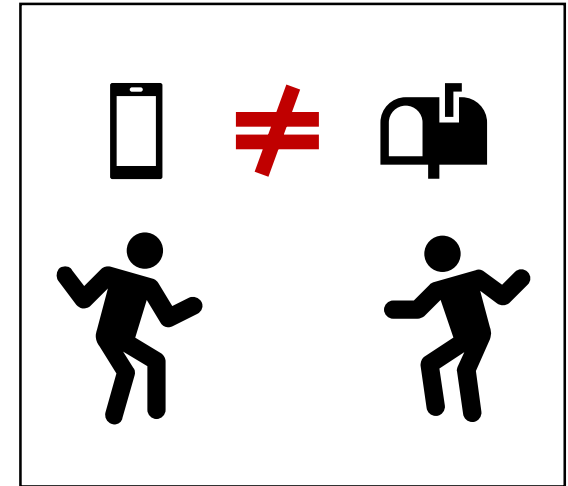
거리



전달속도



다름



글로벌 스케일 네트워킹에선 어떤 모습일까요?

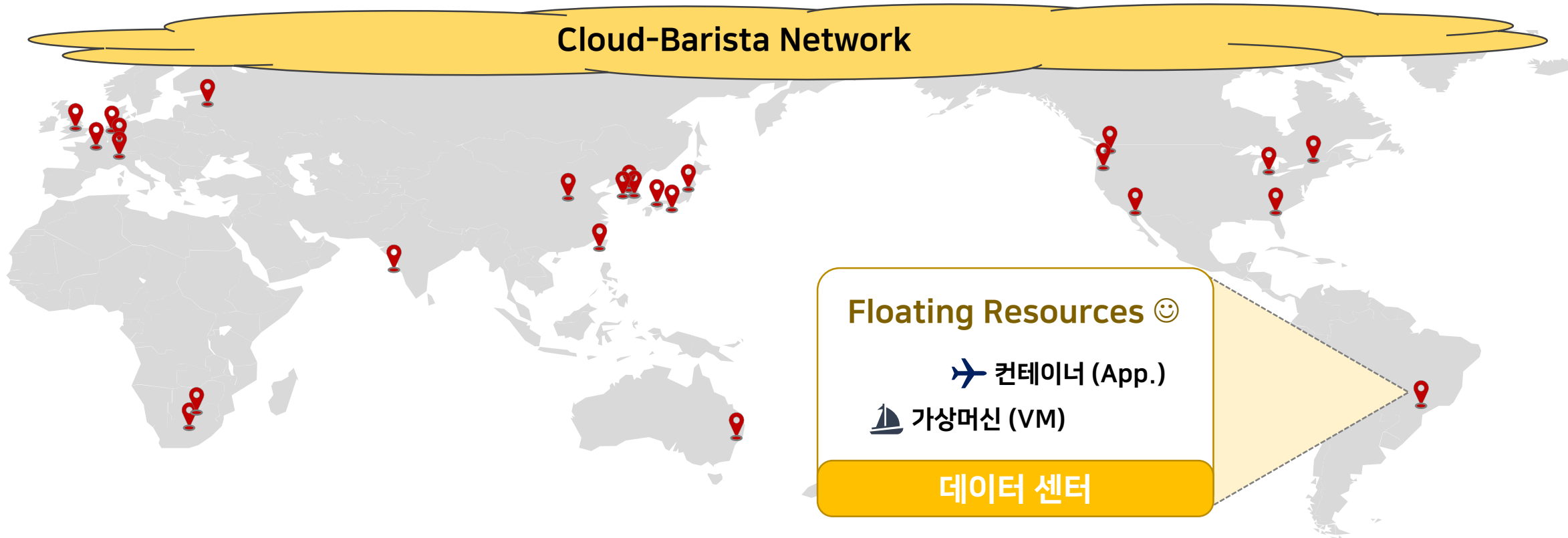
글로벌 스케일 네트워킹이란?!

* 멀티클라우드 인프라 및 응용의 통신을 위해 필수!!

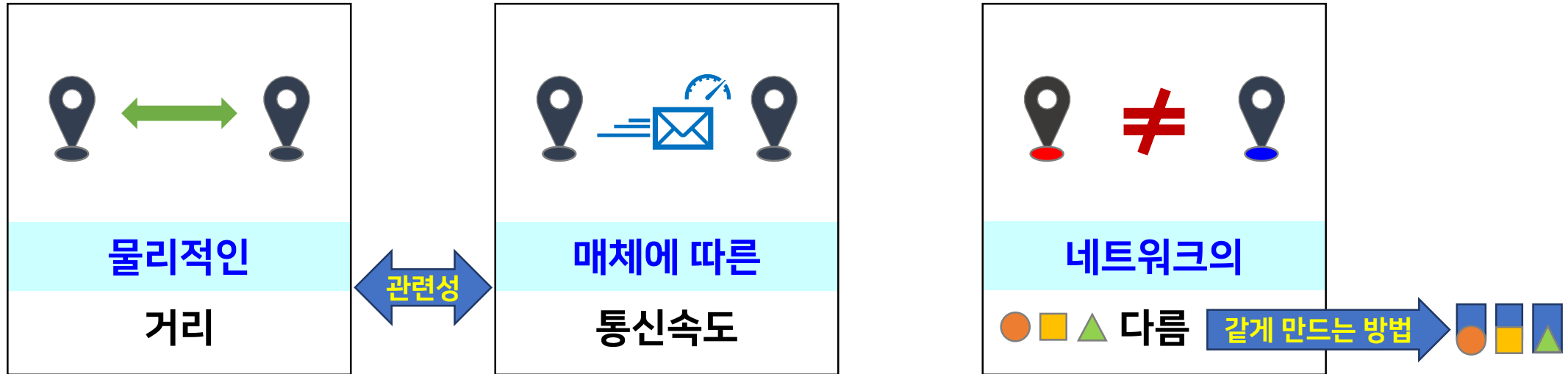
전세계 클라우드 인프라를 엮기 위해 클라우드 네트워크의 상이함과 가변성을 해결하는 효율적인 통신 방법

- “클라우드 네트워크”는 클라우드 서비스 사업자(CSP), 지역(Region), 구역(Zone)에 따라 **네트워크 구성이 상이함**
- 정책 및 상황에 따라 가상머신(VM)/컨테이너에 **할당되는 네트워크가 변경 될 수 있음(예, IP 변경)**
- 멀티클라우드 상의 가상머신(VM)/컨테이너 간 **거리 및 통신 속도가 매우 가변적임**

Cloud-Barista Network



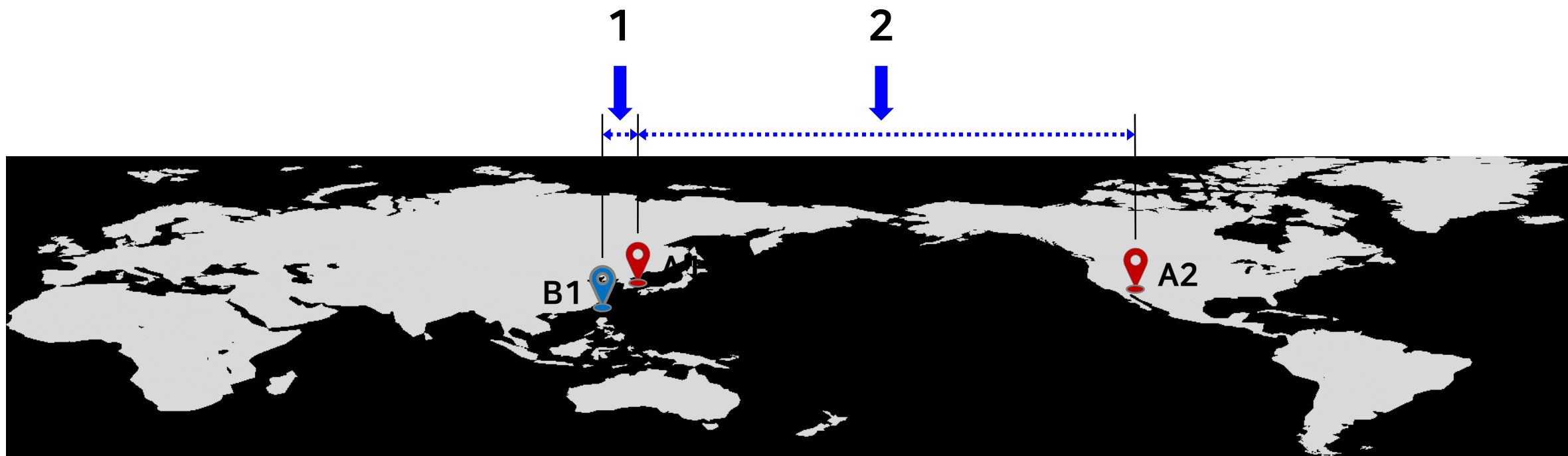
글로벌 스케일 네트워킹의 고려사항



[거리/속도] 물리적 거리 및 통신 속도

글로벌 스케일 네트워킹에서는 “물리적 거리”와 “매체의 통신 속도에 따른 상대적인 거리”를 함께 고려해야함!

- [그림 1과 2] 두 데이터 센터의 물리적인 거리가 증가하면 ▲ 통신속도(상대적인 거리)가 증가함 ▲
- 물리적인 거리가 증가하면 상대적인 거리가 항상 함께 증가하는가? NO! 통신 매체에 따라 달라질 수 있음
 - 예) A1-A2는 광 통신, A1-B1은 일반 통신하는 경우 상대적인 거리는 $2 < 1$ 일 수 있음 ← Under-study



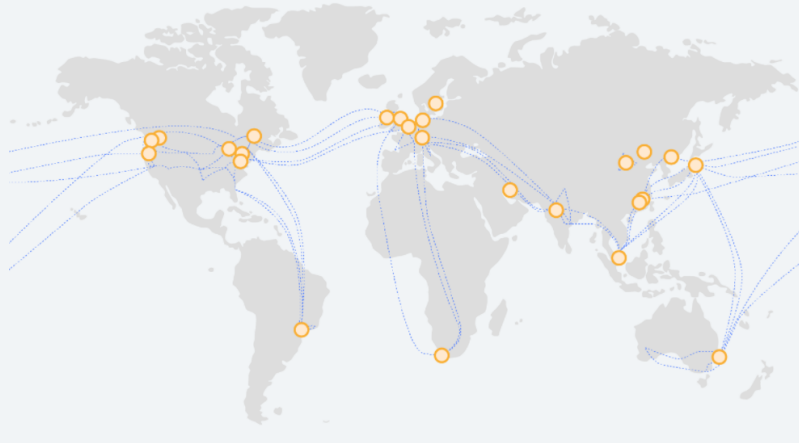
[참고] 글로벌 3사의 Fiber Network 현황

Global Network

Performance is a key driver of the design of the AWS global infrastructure. AWS has the largest global infrastructure footprint of any cloud provider, and this footprint is expanding continuously to help customers deliver better end-user experiences, rapidly expand operations to virtually any region or country, and to meet their data locality and sovereignty requirements. If a business wishes to serve its customers in Europe, the customer should be able to choose from a broad selection of regions or data centers in Europe such as Paris, London, Frankfurt or Dublin to deliver the best possible customer experience.

Global Infrastructure

Every data center, AZ, and AWS Region is interconnected via a purpose-built, highly available, and low-latency private global network infrastructure. The network is built on a global, fully redundant, parallel 100 GbE metro fiber network that is linked via trans-oceanic cables across the Atlantic, Pacific, and Indian Oceans, as well as the Mediterranean, Red Sea, and South China Seas.



Azure global network

Enable a wide range of enterprise and consumer services with a highly available, secure, and agile network

Global network Overview Features Documentation

- ✓ Join one of the largest networks in the world, backed by decades of continuous investment.
- ✓ Extend your reach across 60+ Azure regions with speed and scale to meet your needs.
- ✓ Trust the same resilient network infrastructure that supports Skype, Bing, and Microsoft Exchange.



- ✓ Fastest connectivity from your datacenter to the cloud at 100 Gbps through Azure ExpressRoute
- ✓ More than 170 global network POPs
- ✓ IP traffic stays entirely within our global network and never enters the public Internet
- ✓ 130,000 Miles of lit fiber optic and undersea cable systems

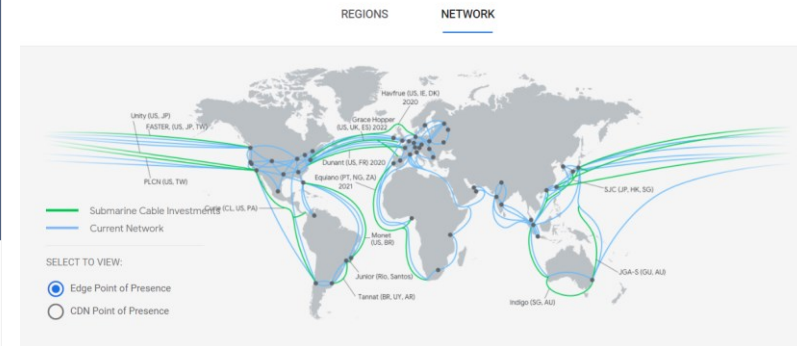
Premium Tier

Premium Tier delivers GCP traffic over Google's well-provisioned, low latency, highly reliable global network. This network consists of an extensive global private fiber network with over 100 points of presence (POPs) across the globe. By this measure, Google's network is the largest of any public cloud provider. See the Google Cloud network map.

GCP customers benefit from the global features within Global Load Balancing, another Premium Tier feature. You not only get the management simplicity of a single anycast IPv4 or IPv6 Virtual IP (VIP), but can also expand seamlessly across regions, and overflow or fail over to other regions.



Meet our network



출처:

AWS, Global Network (Accessed on 2021-05-21, https://aws.amazon.com/about-aws/global-infrastructure/global_network/?nc1=h_ls)

Azure, Azure global network (Accessed on 2021-05-21, <https://azure.microsoft.com/en-us/global-infrastructure/global-network/>)

Google Cloud, Network Service Tiers (Accessed on 2021-05-21, <https://cloud.google.com/network-tiers/>)

[성능비교 결과] 물리적 위치와 통신속도의 관계

성능비교 결과(CLADNet status test를 활용):

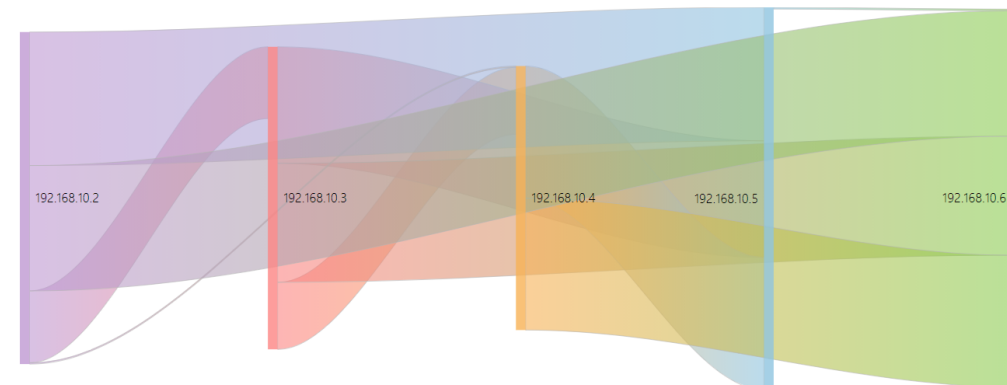
물리적 위치와 통신속도(Latency)간 비례 관계를 보임
(동일 클라우드 내라면 통신속도가 빠를 수 있다고 생각했는데... 😞)

- [3.55ms] AWS London (VM5) – GCP London (VM6)
- [2.78ms] AWS Sydney (VM2) – GCP Sydney (VM4)
- [140.48ms] GCP Changhua (VM3) – GCP Sydney (VM4)
- [150.57ms] GCP Changhua (VM3) – AWS Sydney (VM2)
- [244.04ms] GCP Changhua (VM3) – GCP London (VM5)
- [249.35ms] GCP Changhua (VM3) – AWS London (VM6)
- [277.10ms] GCP Sydney (VM3) – GCP London (VM5)
- [276.02ms] GCP Sydney (VM3) – AWS London (VM6)
- [279.64ms] AWS Sydney (VM2) – GCP London (VM5)
- [262.77ms] AWS Sydney (VM2) – AWS London (VM6)

CLADNet status table

Source	Destination	Latency (ms)
192.168.10.5 GCP, VM5, London	192.168.10.6 AWS, VM6, London	3.55
192.168.10.3 GCP, VM3, Changhua	192.168.10.4 GCP, VM4, Sydney	140.48
192.168.10.3 GCP, VM3, Changhua	192.168.10.5 GCP, VM5, London	244.04
192.168.10.3 GCP, VM3, Changhua	192.168.10.6 AWS, VM6, London	249.35
192.168.10.4 GCP, VM3, Sydney	192.168.10.5 GCP, VM5, London	277.10
192.168.10.4 GCP, VM3, Sydney	192.168.10.6 AWS, VM6, London	276.02
192.168.10.2 (aws) AWS, VM2, Sydney	192.168.10.3 GCP, VM3, Changhua	150.57
192.168.10.2 (aws) AWS, VM2, Sydney	192.168.10.4 GCP, VM4, Sydney	2.78
192.168.10.2 (aws) AWS, VM2, Sydney	192.168.10.5 GCP, VM5, London	279.64
192.168.10.2 (aws) AWS, VM2, Sydney	192.168.10.6 AWS, VM6, London	262.77

CLADNet status chart



[다름] CSP 네트워크의 다양성

* 각기 다른 방식으로 클라우드 네트워크 구축 (3차 컨퍼런스 참고)

각 CSP의 네트워크 관리 및 운용 영역/방침 상이

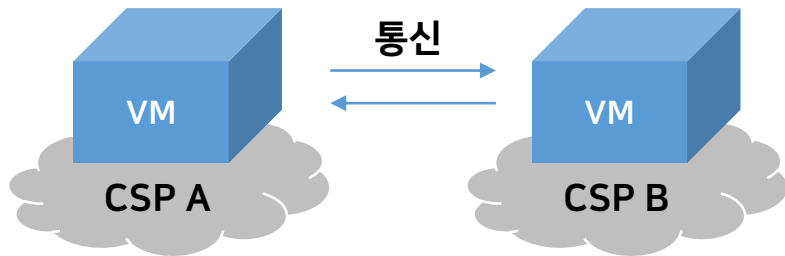
	AWS	Azure	GCP
Virtual network	Virtual Private Cloud (VPC)	Virtual Network (VNet)	Virtual Private Cloud (VPC)
Virtual network scope	Regional	Regional	Global
Network created by default	Yes	No	Yes (can be disabled)
Subnet scope	Zonal	Regional	Regional
Routing scope	Subnet	Subnet	VPC
Controlling in/outbound traffic	<ul style="list-style-type: none"> • Network Access Control Lists • Security Groups 	<ul style="list-style-type: none"> • Network Security Groups • Application Security Groups 	<ul style="list-style-type: none"> • Firewall
Private connections to services	VPC Endpoints	Service Endpoints	Private Google Access
Connections to other networks	<ul style="list-style-type: none"> • VPC peering • AWS Site-to-Site VPN • AWS Direct Connect 	<ul style="list-style-type: none"> • NET Peering • VPN Gateway • Express Route 	<ul style="list-style-type: none"> • VPC Network Peering • Cloud VPN • Cloud Interconnect
IPv6 support	Yes	Yes	No



Cloud-Barista는 서로 다른 클라우드 네트워크 상에 생성되는 가상머신/컨테이너를 관리, 운용함

커뮤니티 측면의 통신 관련 이슈와 요구사항

멀티클라우드 환경



통신 관련 실제 이슈

1. 멀티클라우드의 각 VM이 서로 **다른 대역의 네트워크(Subnet)**에서 구동됨
2. Public IP를 통한 통신을 해야 함에도 **VM 내부에서는 Private IP 만 알고 있음**
3. 네트워크 기술(예, VPN)을 활용하여 **동일 Subnet**을 구성함
4. 위 2, 3을 만족하면 서로 통신이 되지만 **Pod-Pod 통신에서 이슈 존재**

Cloud-Barista Network (cb-network) 요구사항

- ✓ **(다름)** 서로 다르고, 변화 가능한 네트워크상의 VM이 **하나의 네트워크 상에 존재하는 것 처럼 인식**해야 함
- ✓ **(위치 및 매체)** 통신 성능을 최대한으로 이끌어 내야함
- ✓ **(중요)** 글로벌 스케일 네트워크가 **Fault Tolerance** 해야함 (High availability (HA)보다 중요함)
- ✓ **(중요)** CSP 독립적인 네트워크를 구성 해야함 (Vendor lock-in 회피)
- ✓ **(중요)** VM의 **네트워크 변화에 동적으로 대응**할 수 있어야 함
- ✓ (기타) 여러 MCIS에 네트워크를 구분하여 할당 해야함
- ✓ (기타) 사용자(네트워크 전문가)를 위한 네트워크 할당 기능/인터페이스를 제공해야함



멀티클라우드 서비스 공통 플랫폼

CLADNet: Cloud Adaptive Network

- 설계 및 구현 -

카페모카(Café Mocha) 한잔 어떠세요 ?

[다름을 같음으로] Cloud Adaptive Network (CLADNet)

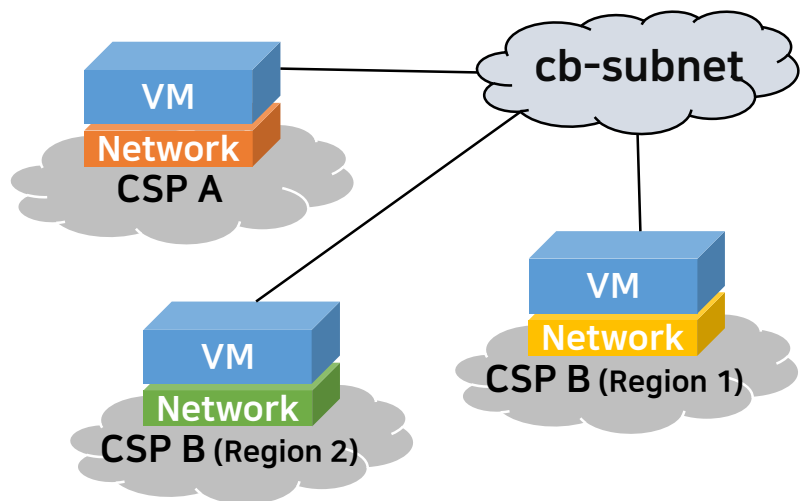
CLADNet: 멀티클라우드의 다양한 네트워크에 적응가능한 오버레이 네트워크로 논리적인 노드 그룹에 동일 네트워크를 제공함

예: Multi-Cloud Infra Service (MCIS)

[Before] 제3차 컨퍼런스(Espresso)

서로 다른 환경임에도 동일한 네트워크 처럼 활용할 수 있는
글로벌 네트워크 서비스 제공

Proof of concept (POC)



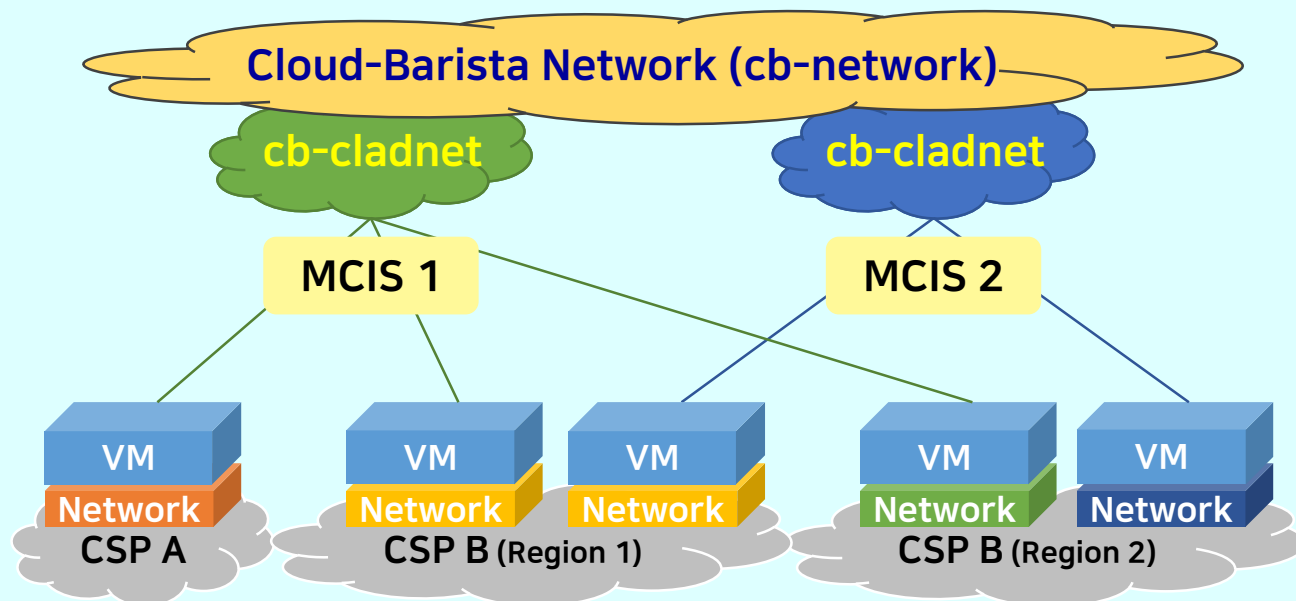
• cb-subnet

- 멀티클라우드를 위한 공통/동일 Subnet
- Cloud-Barista의 메인 네트워크를 개발 중(오해의 소지)

Subnet, 그 이상의 것을
제공하기 위하여 ☺

[After] 제4차 컨퍼런스(Cafe Mocha)

기존 클라우드 네트워크의 상이함과 변동성을 완화하여
전세계 클라우드 인프라를 엮는 글로벌 네트워킹 서비스 제공



• cb-cladnet

* clad: 차려입은, 장비한 (clothe의 과거분사)

- Cloud-Barista에서 자체 개발한 메인 네트워크를 의미함
- 여러 CSP의 상이한 네트워크에 적응할 수 있는 네트워크를 의미함

CLADNet를 위한 지향점, 구조 및 기능

지향점

Adaptive: 여러 사업자의 상이한 Cloud Network에 적응 가능한 네트워크

Fault tolerant: 사업자 및 리전 이슈에 대비하는 글로벌 장애 허용 네트워크

Lightweight: 호스트(VM) 자원 사용을 최소화하는 경량한 네트워크

Handy: 사용자가 쉽고 빠르게 사용할 수 있는 네트워크

구조

Event-driven 아키텍처: 분산 Key-Value store 기반의 Event-driven 아키텍처로 데이터의 변경, 생성, 삭제(CUD)시 발생하는 서비스의 의미있는 변화를 바탕으로 효율적인 워크플로우 수행

→ Microservice Architecture를 향해 나아가는 중

메쉬(Mesh)형 토폴로지: 오버레이 네트워크를 메쉬형 토폴로지로 구성하여 중개 노드의 위치에 따른 성능 차이를 최소화함

→ IPSec 등 타 프로토콜 적용을 위해 Pluggable Interface로 구조 개선 예정

기능

Watch 기반의 CLADNet 정보 동적 관리 및 업데이트

TUN 기반의 가상 네트워크 인터페이스 할당 기능

네트워킹 규칙(Rule) 기반의 페이로드 En/decapsulation 기능

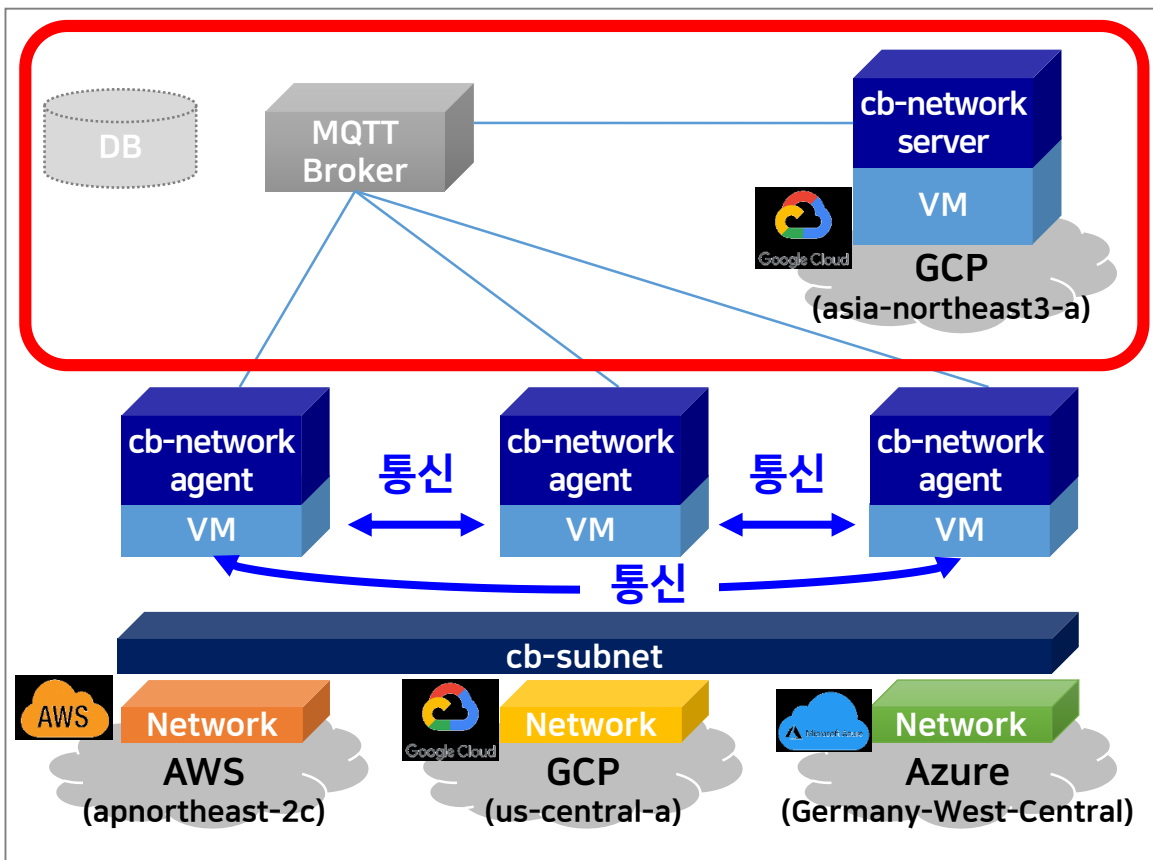
사용자 요구 기반의 CLADNet 할당 기능

Compare-and-swap (CAS) 기반의 호스트 내/외부 네트워크 정보 업데이트 기능

CLADNet 상태 확인 기능

기존 cb-network 시스템의 보완 포인트

제3차 컨퍼런스의 형상: MQTT기반 cb-network 시스템 (POC)



보완 포인트:

- cb-network 시스템내 Single Point of Failure (SPOF) 존재
- MQTT broker의 위치에 따른 Networking rule 동기화 성능 차이
- Networking Rule 저장 및 관리 방안 부재
- 다중 cb-network agent의 요청 처리를 위한 동시성 처리 부재

보완 목표:

글로벌 스케일 네트워킹을 위해
위치 및 거리를 고려한 cb-network 시스템 운용 방안 마련

보완 방안: 1) Event-driven 아키텍처로 구조 개선, 2) 네트워크 제어 방식 대폭 개선

- 분산 Key-value store 구축 → SPOF 해결, 네트워크 정보 관리
- cb-network server 및 agent 재기동 방식 추가 → SPOF 해결
- Client Load Balancer 적용 → 부하 분산 및 동기화 성능 차이 완화
- 분산 락(lock) / Compare-and-swap (CAS) → 동시성 처리, 경량성 지원
- Event-driven 네트워크 제어 → 네트워크 정보 동적 업데이트

[위치, 속도, 다름 고려] cb-network 시스템 구축

멀티클라우드를 위한 “최적의 cb-network 시스템” 구축 방안은?

고민: 멀티클라우드 상에 **cb-network 시스템의 컨트롤 플레인**을 어떻게 배치해야 할까?



...

최적의 cb-network 시스템 구축이란?

- ✓ 네트워킹 성능 최대화(Networking performance maximization)
- ✓ 결함 허용 네트워크(Fault tolerant network)

[모델링 스터디] 멀티클라우드 상에 컨트롤 플레인 배치 방안 필요

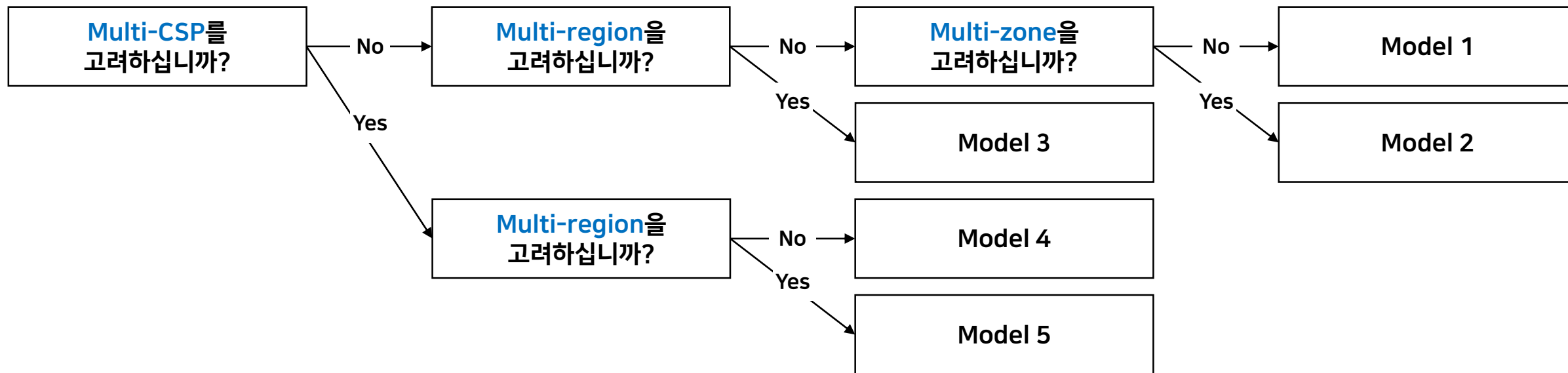
일반화

주의!! Cloud-Barista 플랫폼 Itself를 위한 컨트롤 플레인 아님

사용자가 글로벌 스케일 인프라에서 자체 서비스를 제공하고자 할 때 해당 “서비스를 위한 컨트롤 플레인 배치 방안”이 필요함

- 가정) 서비스 제공을 위한 컨트롤 플레인은 클러스터 형태로 구성됨
- 결정 방법) 결정 트리(Decision Tree)를 통해 “컨트롤 플레인 배치 모델”을 도출함

클러스터(Multi-host) 구성 시



컨트롤 플레인 배치 모델(cont'd)

• Model 1 – Single-zone cluster

- 컨트롤 플레인의 Hosts를 하나의 zone에 배치
- 해당 Zone에 이슈가 있을 시 서비스 장애 발생
- 테스트 용도로 활용

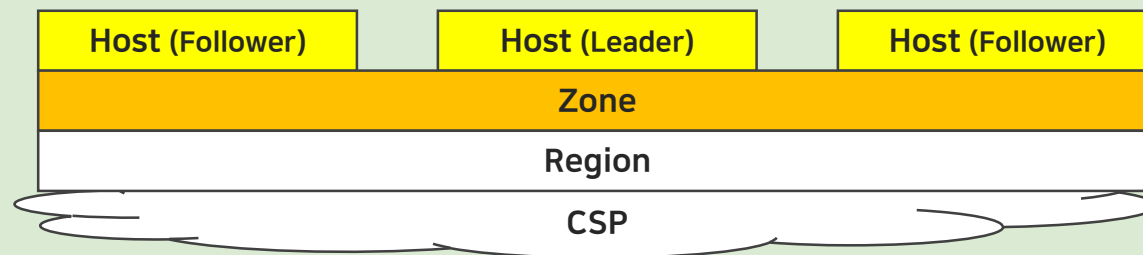


• Model 2 – Multi-zone cluster

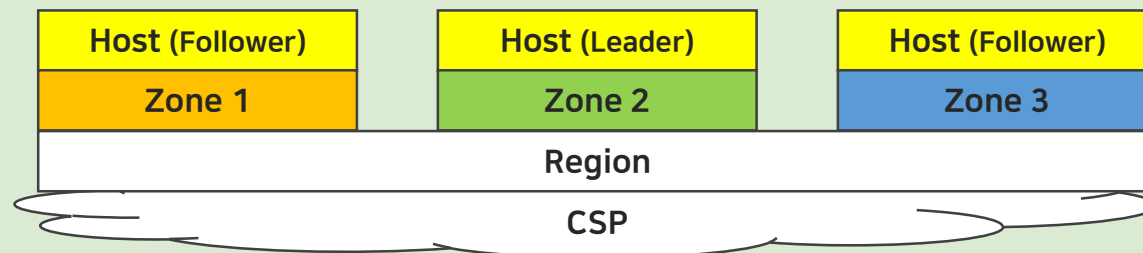
- 컨트롤 플레인의 Hosts를 하나의 Region, 여러 Zone에 배치
- 해당 Region에 이슈가 있을 시 서비스 장애 발생
- 지역 서비스 시 활용 및 별도 대응방안 마련 필요



컨트롤 플레인 클러스터 - Model 1



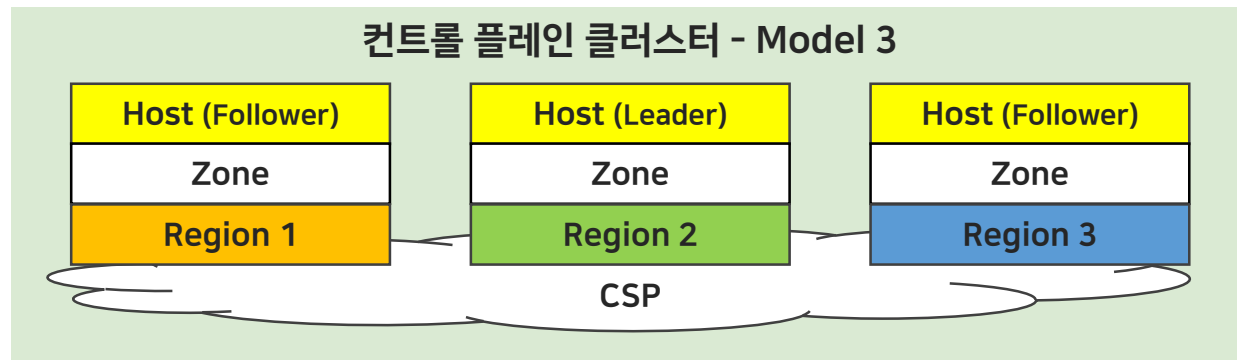
컨트롤 플레인 클러스터 - Model 2



컨트롤 플레인 배치 모델(cont'd)

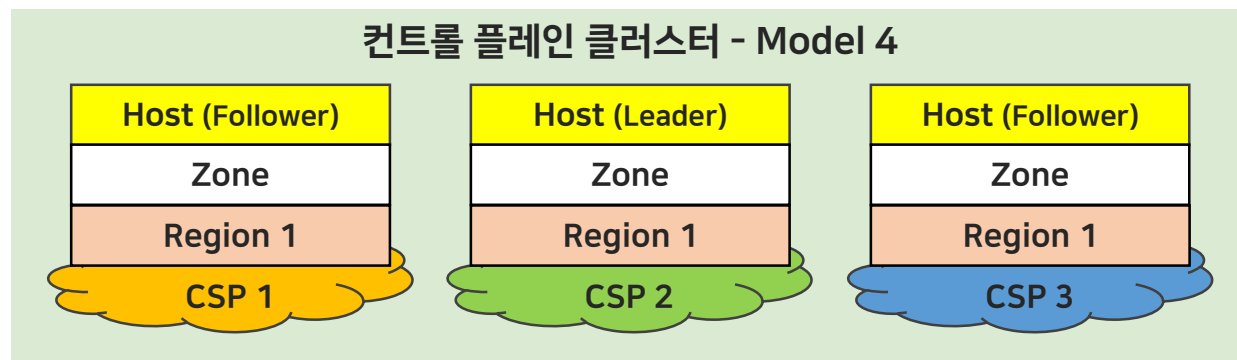
• Model 3 – Multi-region cluster in a single CSP

- 컨트롤 플레인의 Hosts를 하나의 CSP, 여러 Region에 배치
- Region이 달라지면 Zone이 달라짐
- 지역적인 이슈 발생시 Fault tolerance 함
- CSP 수준의 문제(예, 데이터센터 전소) 발생시 장애로 이어질 수 있음



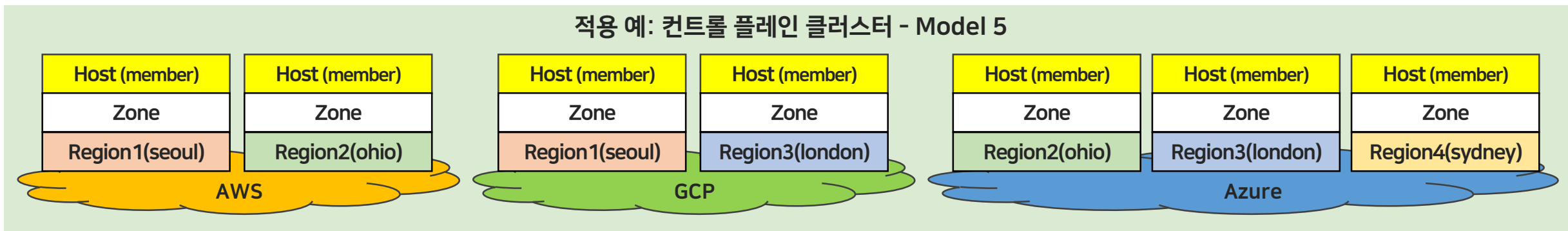
• Model 4 – Multi-CSP cluster in a single region

- 컨트롤 플레인의 Hosts를 여러 개의 CSP에 배치
- 논리적으로 동일 Region
- CSP 수준의 이슈 발생시 Fault tolerance 함
- 지역적인 문제 (예, 지진으로 기반시설 손실) 발생시 장애로 이어질 수 있음



컨트롤 플레인 배치 모델

- Model 5 – Multi-CSP cluster in multi-region
 - 컨트롤 플레인의 Hosts를 여러 CSP의 여러 Region에 배치



➡ Model5는 글로벌 스케일 관점에서 Fault Tolerance(FT) 및 지역적(Region) 성능 품질을 고려한 모델

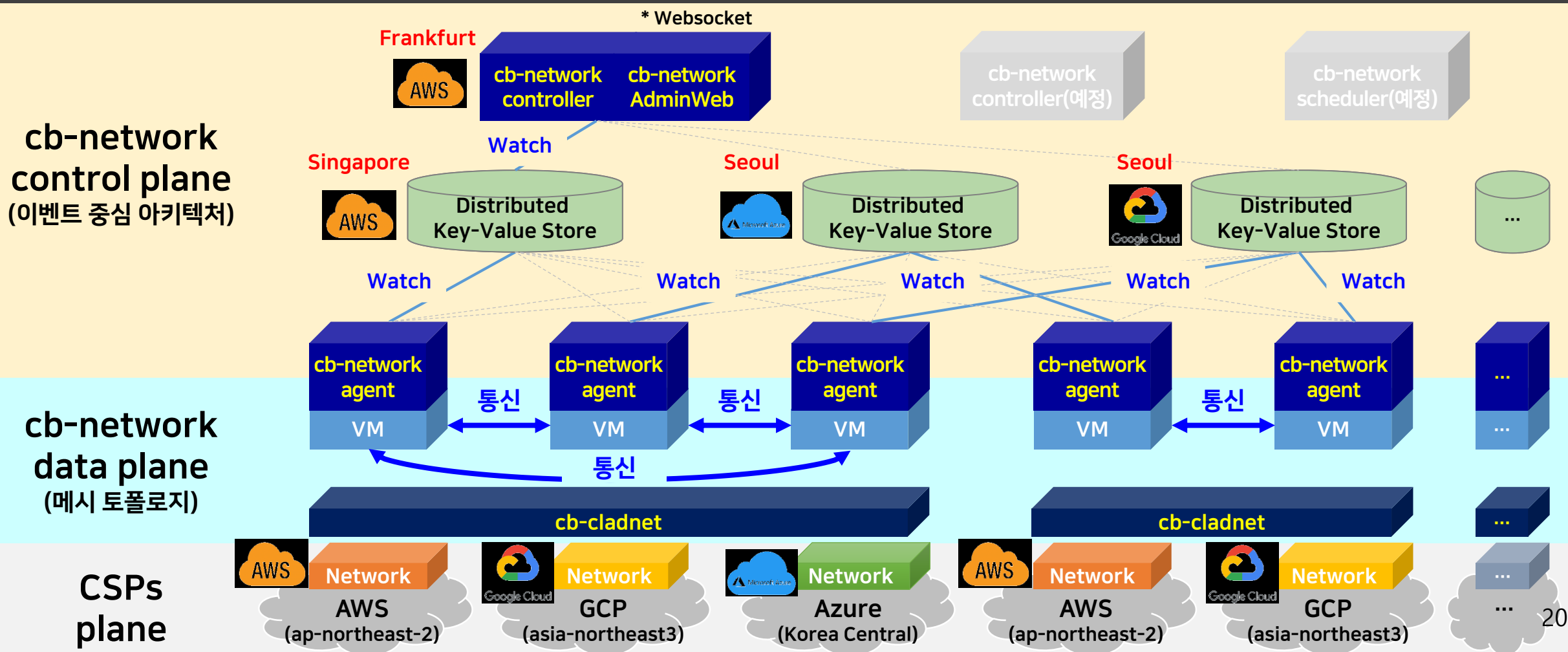
예1 - 지역적 성능) Cloud A의 Seoul region이 다운 되어도 클라우드 B의 Seoul region에서 데이터 이용 가능

예2 - 장애 허용) 천재지변으로 인해 Region A의 모든 CSP에 장애가 발생시 Region B에서 데이터 이용 가능

➡ 글로벌 스케일로 시작하는 멀티클라우드를 기반으로
우리의 서비스를 세계 곳곳으로 보내는 그날까지...



Event-driven Architecture, Mesh Topology, MCIS마다 CLADNet 할당

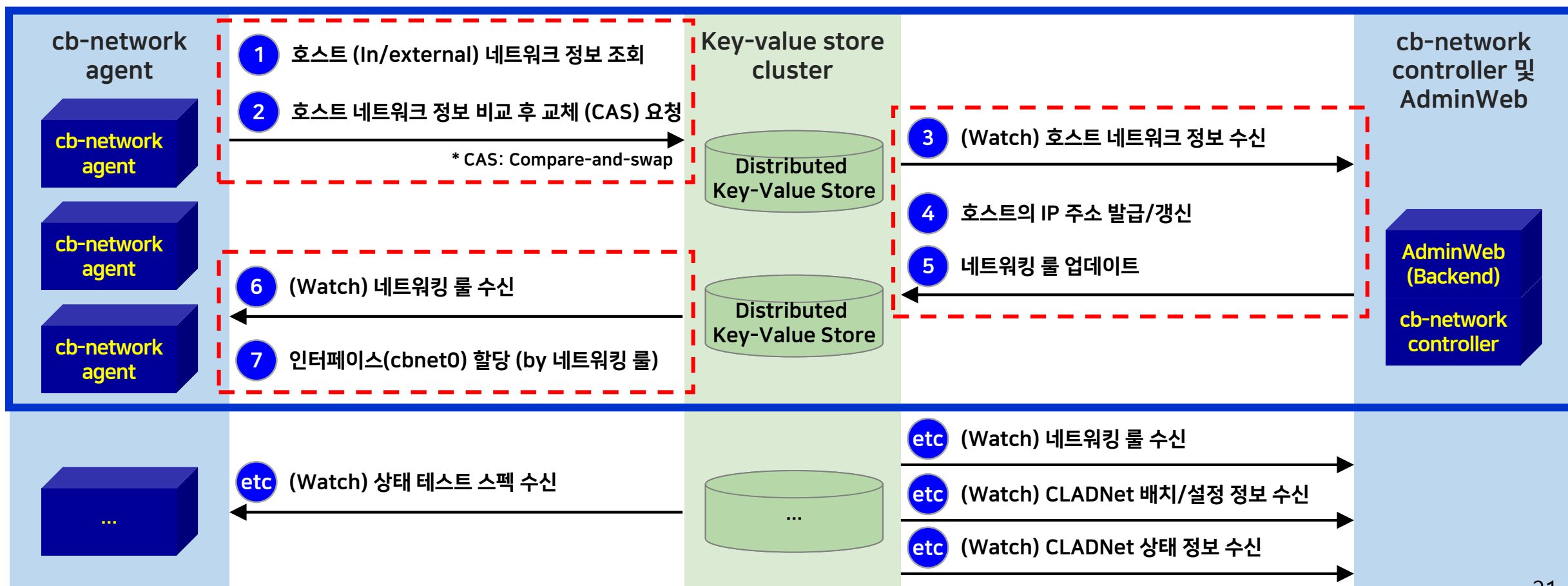


[워크플로우] Agent의 CLADNet 참가/연결 + 기타

사전작업:

1. cb-network 시스템의 Control Plane이 작동하고 있는 상태
2. 사용자가 CLADNet CIDR block을 생성하고, CLADNet ID를 발급 받아 agent에 할당한 상태

대표 워크플로우

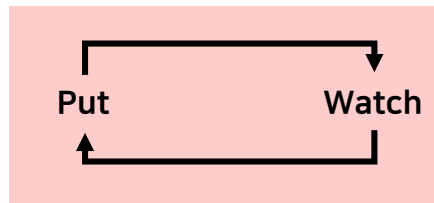


cb-network 시스템의 데이터 구조

목적: Watch 기능을 최대한 활용하기 위한 데이터 구조 디자인

- (부적합) 데이터 구조 v0.1: 큰 단위 → 작은 단위 → Watch대상
- 사유:
 - WildCard를 적용하여 Watch 하기에 적절하지 않음
 - Watch한 데이터를 업데이트 하기 때문에 Looping 이슈 발생

cladnet/{group-id}/{vm-id}/host-network-information
cladnet/{group-id}/networking-rule



기존 사례 분석: K8s의 etcd 데이터 구조 참고

```
1 $ for k in $(cat etcd-kv.json | jq '.kvs[].key' | cut -d '"' -f2); do echo $k | base64 --decode
2
3 /registry/apiregistration.k8s.io/apiservices/v1.
4 /registry/apiregistration.k8s.io/apiservices/v1.apps
5 /registry/apiregistration.k8s.io/apiservices/v1.authentication.k8s.io
6 /registry/apiregistration.k8s.io/apiservices/v1.authorization.k8s.io
7 /registry/apiregistration.k8s.io/apiservices/v1.autoscaling
8 /registry/apiregistration.k8s.io/apiservices/v1.batch
9 /registry/apiregistration.k8s.io/apiservices/v1.coordination.k8s.io
10 /registry/apiregistration.k8s.io/apiservices/v1.networking.k8s.io
11 /registry/apiregistration.k8s.io/apiservices/v1.rbac.authorization.k8s.io
12 /registry/apiregistration.k8s.io/apiservices/v1.scheduling.k8s.io
13 /registry/apiregistration.k8s.io/apiservices/v1.storage.k8s.io
14 /registry/apiregistration.k8s.io/apiservices/v1beta1.admissionregistration.k8s.io
15 /registry/apiregistration.k8s.io/apiservices/v1beta1.apixtensions.k8s.io
16 /registry/apiregistration.k8s.io/apiservices/v1beta1.apps
```

- (적합) 데이터 구조 v0.2: Watch대상 → 큰 단위 → 작은 단위

cloud-adaptive-network/configuration-information/{group-id}
cloud-adaptive-network/host-network-information/{group-id}/{hostid}
cloud-adaptive-network/networking-rule/{group-id}
cloud-adaptive-network/status/test-specification/{group-id}
cloud-adaptive-network/status/information/{group-id}/{host-id}

출처:

Luc Juggery, A Closer Look at Etcd: The Brain of a Kubernetes Cluster, 2019. (Accessed on 2021-03-22, <https://betterprogramming.pub/a-closer-look-at-etcd-the-brain-of-a-kubernetes-cluster-788c8ea759a5>)

AdminWeb 개선 (기존 cb-network 대시보드)

통신 프로토콜 및 매커니즘 개선: 네트워크 정보의 Event-driven 업데이트를 위해 **Websocket 적용** (기존 MQTT 통신)

cb-network 구성 정보 (Multi-CLADNet 지원)



The AdminWeb of Cloud-Barista Network

See AdminWeb

The configuration of Cloud-Barista Network (cb-network)

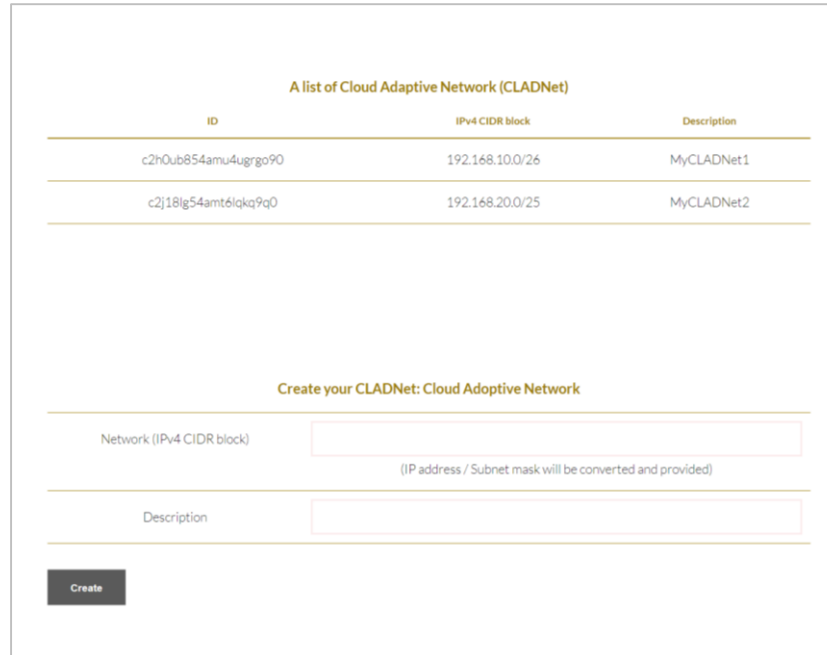
Host ID	Host IP CIDR Block	Host IP Address	Public IP
gcp-eu-west2-kimyk01-0	192.168.10.2/26	192.168.10.2	35.246.30.12
gcp-australia-se1-kimyk01-0	192.168.10.3/26	192.168.10.3	34.116.84.82
aws-ap-southeast-2-kimyk01-lead-0	192.168.10.4/26	192.168.10.4	52.63.143.20

The cb-network configuration



CLADNet c2h0ub854amu4ugrgo90

CLADNet CIDR block 생성 기능 (CSP의 웹 콘솔/포털 벤치마킹)



A list of Cloud Adaptive Network (CLADNet)

ID	IPv4 CIDR block	Description
c2h0ub854amu4ugrgo90	192.168.10.0/26	MyCLADNet1
c2j18lg54amt6lqkq9q0	192.168.20.0/25	MyCLADNet2

Create your CLADNet: Cloud Adoptive Network

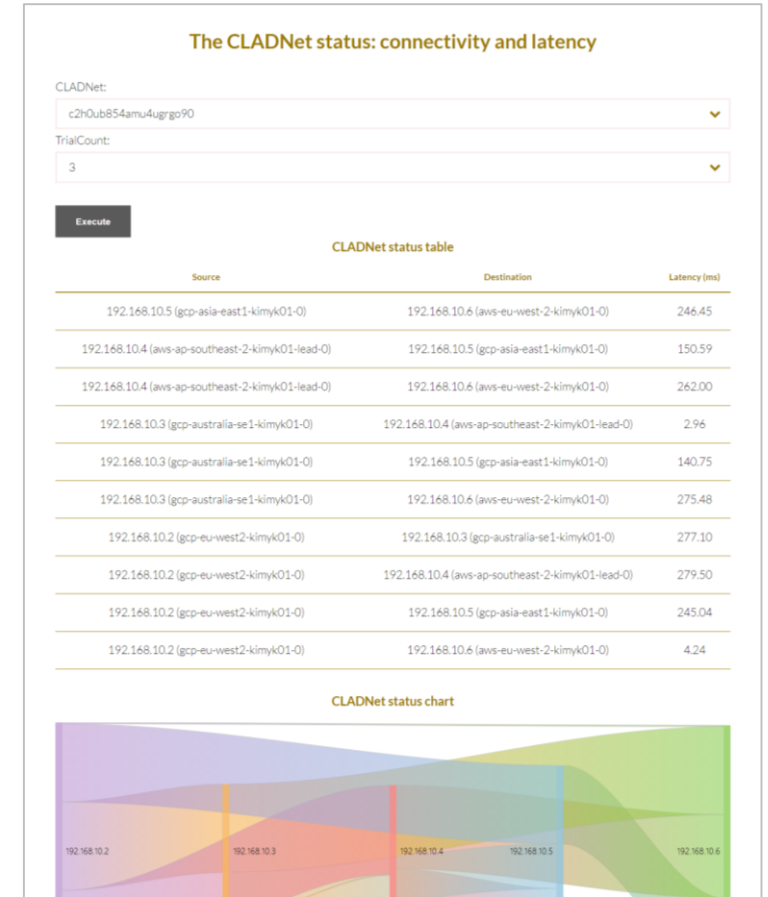
Network (IPv4 CIDR block)

(IP address / Subnet mask will be converted and provided)

Description

Create

CLADNet 상태 확인 기능 (추후 모니터링 기능에 활용)



The CLADNet status: connectivity and latency

CLADNet:

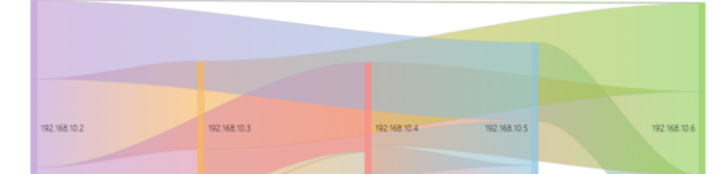
TrialCount:

Execute

CLADNet status table

Source	Destination	Latency (ms)
192.168.10.5 (gcp-asia-east1-kimyk01-0)	192.168.10.6 (aws-eu-west-2-kimyk01-0)	246.45
192.168.10.4 (aws-ap-southeast-2-kimyk01-lead-0)	192.168.10.5 (gcp-asia-east1-kimyk01-0)	150.59
192.168.10.4 (aws-ap-southeast-2-kimyk01-lead-0)	192.168.10.6 (aws-eu-west-2-kimyk01-0)	262.00
192.168.10.3 (gcp-australia-se1-kimyk01-0)	192.168.10.4 (aws-ap-southeast-2-kimyk01-lead-0)	2.96
192.168.10.3 (gcp-australia-se1-kimyk01-0)	192.168.10.5 (gcp-asia-east1-kimyk01-0)	140.75
192.168.10.3 (gcp-australia-se1-kimyk01-0)	192.168.10.6 (aws-eu-west-2-kimyk01-0)	275.48
192.168.10.2 (gcp-eu-west2-kimyk01-0)	192.168.10.3 (gcp-australia-se1-kimyk01-0)	277.10
192.168.10.2 (gcp-eu-west2-kimyk01-0)	192.168.10.4 (aws-ap-southeast-2-kimyk01-lead-0)	279.50
192.168.10.2 (gcp-eu-west2-kimyk01-0)	192.168.10.5 (gcp-asia-east1-kimyk01-0)	245.04
192.168.10.2 (gcp-eu-west2-kimyk01-0)	192.168.10.6 (aws-eu-west-2-kimyk01-0)	4.24

CLADNet status chart





멀티클라우드 서비스 공통 플랫폼

CLADNet 기술 시연

- 글로벌 스케일 MCIS에 CLADNet 입혀 보기 -

* MCIS: Multi-Cloud Infra Service

* CLADNet: Cloud Adaptive Network

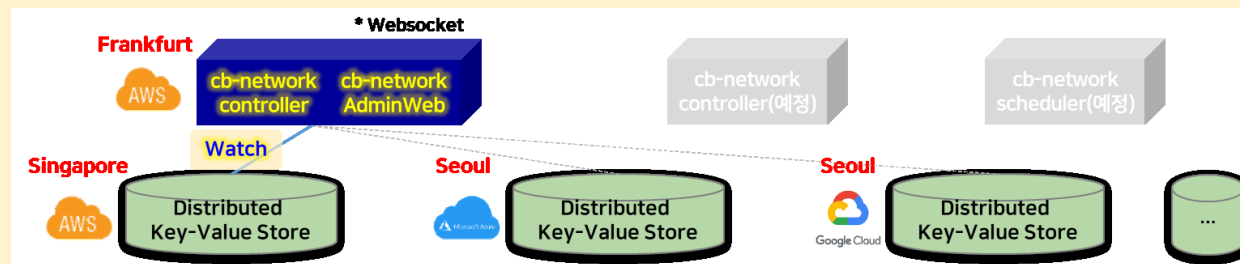
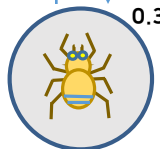
카페모카(Café Mocha) 한잔 어떠세요 ?

I'm a user

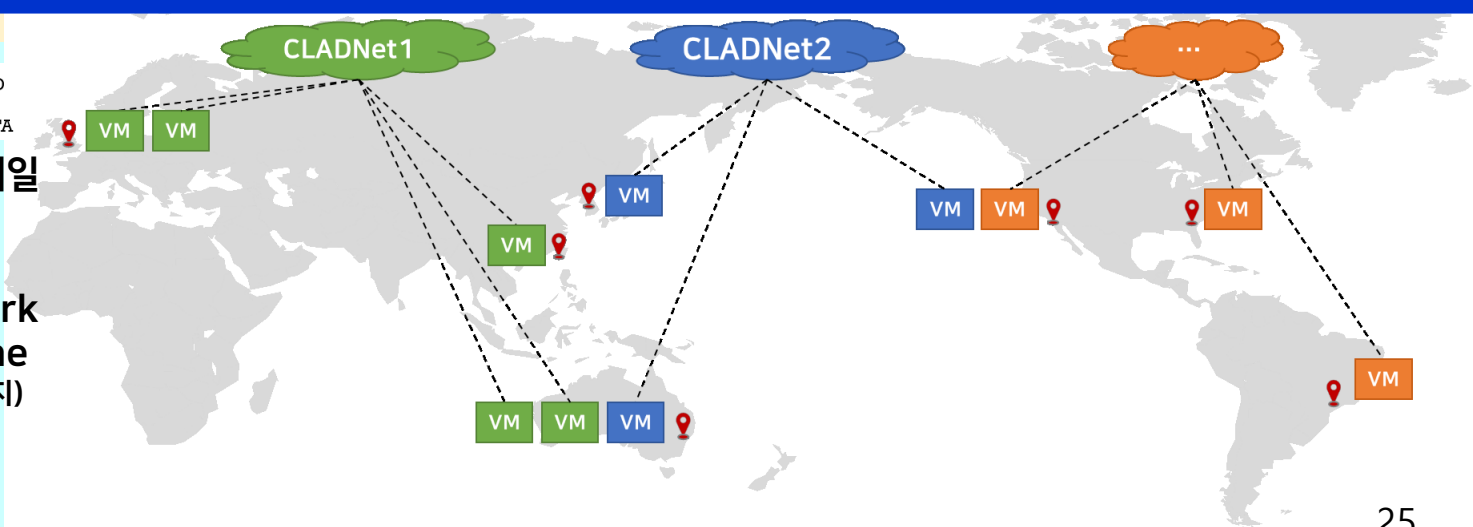
- ### 0.3.6



0.3.8



Cloud-Barista Network (cb-network)



[시나리오] 글로벌 스케일 MCIS에 CLADNet 입혀 보기

영상 시연: 2개의 MCIS 생성 후 각각 CLADNet이 구성됨을 시연함 (시나리오는 아래 참고)

0. 준비작업

1. Distributed key-value store cluster 구동
2. cb-network controller 구동
3. 컨테이너 기반 CB-Spider 구동
4. 소스 기반 CB-Tumblebug 구동

1. CLADNet 설정

1. AdminWeb 접속
2. CLADNet을 위한 Network CIDR 설정 요청
3. CLADNet ID 획득

2. MCIS 생성

1. CB-Tumblebug에 MCIS 생성 요청
2. 생성된 MCIS 상태 확인

3. cb-network agent 배포 시연

1. MCIS에 cb-network agent 배포
2. CB-Tumblebug log를 통해 배포 진행 상황 확인
3. AdminWeb에서 MCIS에 CLADNet 구성 확인

4. CLADNet 상태 확인 시연

1. AdminWeb에서 CLADNet에 상태 확인 요청
2. 각 cb-network agent 에서 연결성 및 성능 테스트 수행
3. AdminWeb에서 결과 값 확인

5. CLADNet을 활용한 WeaveScope 설치 후, 구성 상태 시연

1. WeaveScope 설치 스크립트에 CLADNet 정보 입력
2. MCIS에 WeaveScope 설치 수행
3. WeaveScope 대시보드에서 네트워크 구성 상태(Mesh) 확인
4. MCIS에 cb-network agent 종료 요청
5. WeaveScope 대시보드에서 네트워크 구성 상태(없음) 확인

* CB-Tumblebug API를 활용하여 MCIS를 운용함



멀티클라우드 서비스 공통 플랫폼

cb-storage, 첫 조각을 맞추다

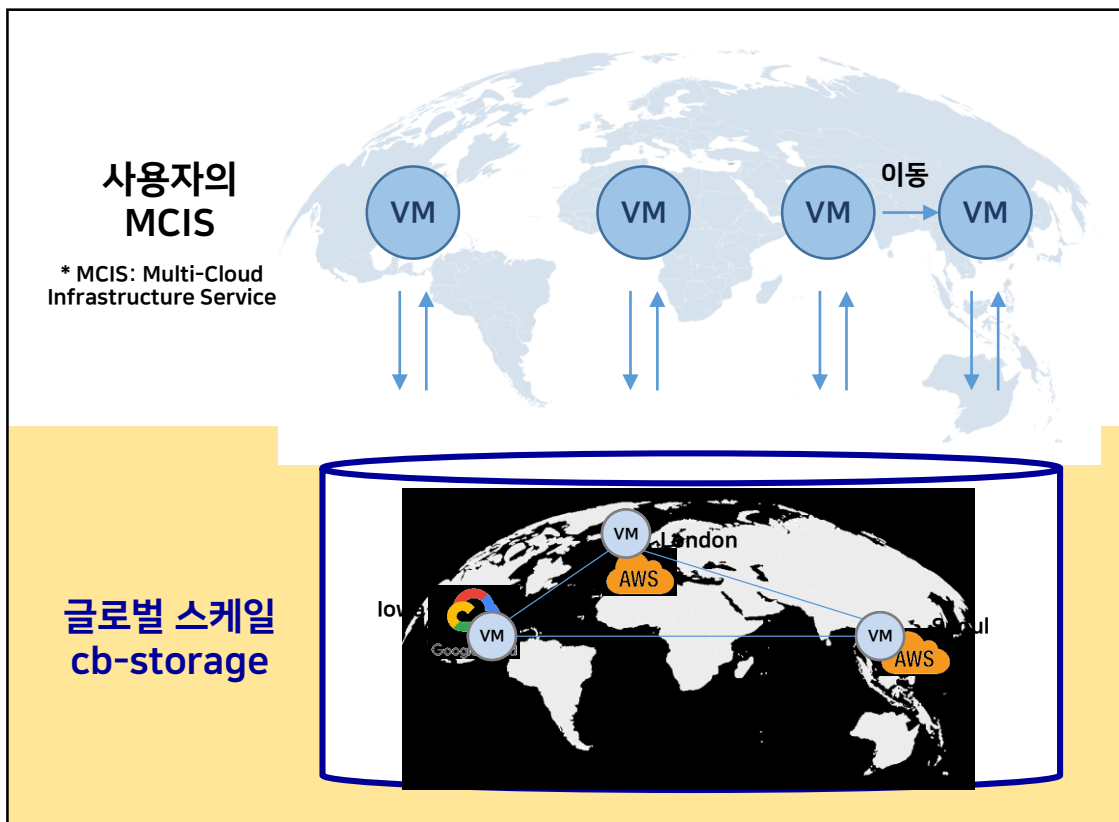
카페모카(Café Mocha) 한잔 어떠세요 ?

멀티클라우드에서 데이터 관리는?

이슈 해결 방안: 멀티클라우드 환경에서 글로벌 스케일 스토리지 제공

※ 멀티클라우드 환경에서 데이터 관리는? 서비스 이동 시 데이터는?

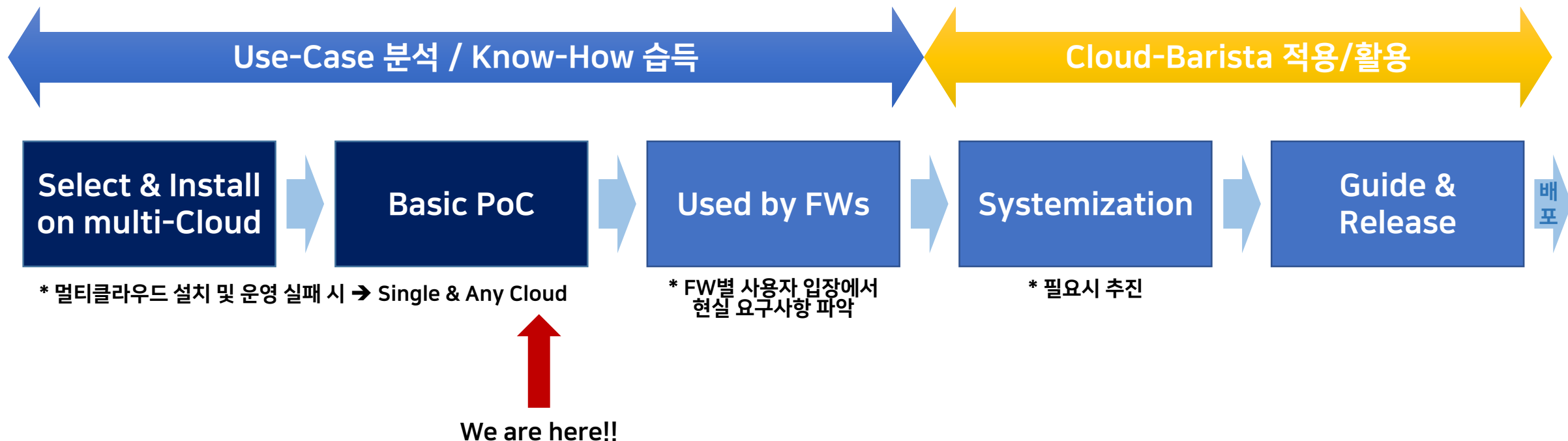
Cloud-Barista Storage (cb-storage) 개념도



사용자 요구사항

- ✓ **데이터 별도 관리:**
멀티클라우드 환경에서 컴퓨팅과 저장 관리를 분리 운영할 수 있는 스토리지를 제공해야 한다.
- ✓ **클라우드 서비스 독립적:**
특정 클라우드 서비스에 의존성이 없는 스토리지 시스템이어야 한다.
- ✓ **데이터 보존:**
사용자의 멀티클라우드 서비스가 처리하는 데이터를 영구 저장 및 관리할 수 있어야 한다.
- ✓ **사용자 편의:**
사용자의 멀티클라우드 서비스 별로 독립 활용할 수 있는 블록 서비스를 제공할 수 있어야 한다.
- ✓ **데이터 공유:**
데이터 인가된 멀티클라우드 서비스 간에 상호 정보를 공유할 수 있어야 한다.²⁸

cb-storage 연구 개발 계획



cb-storage 연구 개발 진행상황

다측면 분석 수행



기술 선정 조건 도출

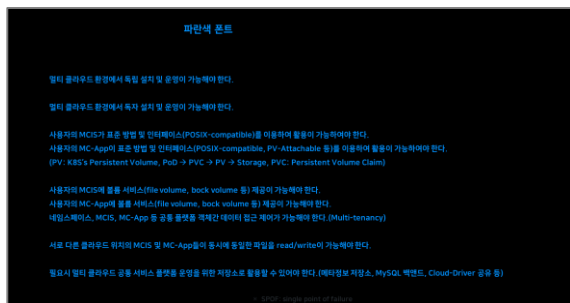


커뮤니티 논의

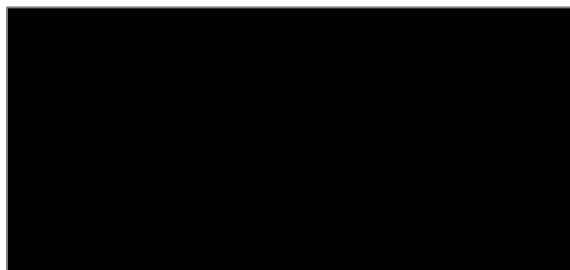


설치/시험을 통한 스토리지 선정

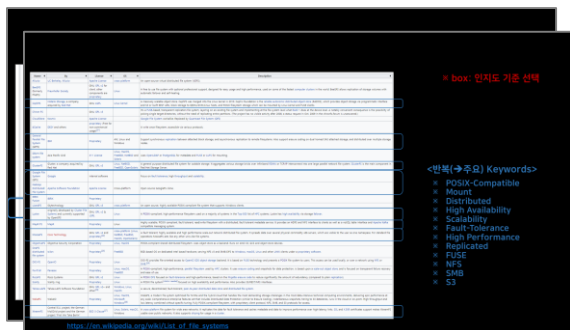
[사용자/시스템 요구 분석]



[사용자 활용 패턴 분석]



[스토리지 기능/특징 키워드 분석]



오래 갈 공개SW, 인지도 등

Global Accessible

Mountable

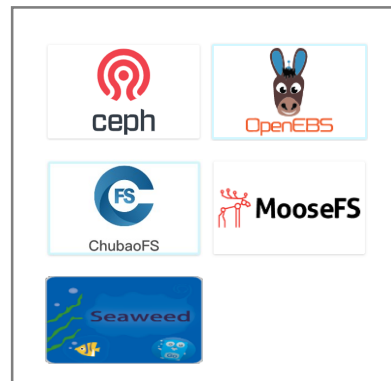
POSIX-compatible

Volume support

Multi-tenancy

그외 덕목: 최신기술, Replica, Scalable, Available, Avoid lock-in, Windows Client, ...

[후보군: 5종]



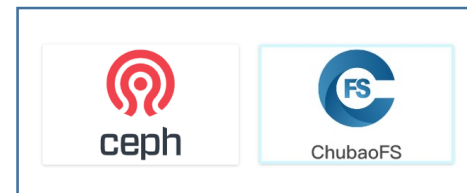
※ 설치 및 운영: 상대적으로 빠름/간단(추정)

[공통 스토리지군: 3종]



※ 설치 및 운영: 상대적으로 오래걸림/복잡(추정)

[FS 선정 및 설치/시험]



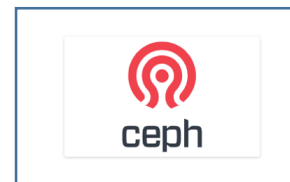
* Ceph 및 ChubaoFS 설치 및 시험

* Ubuntu 기반 ChubaoFS 설치 실패

최종안

[최종안: Ceph]

* 현 시점의 최종안
(변경 될 수 있음)



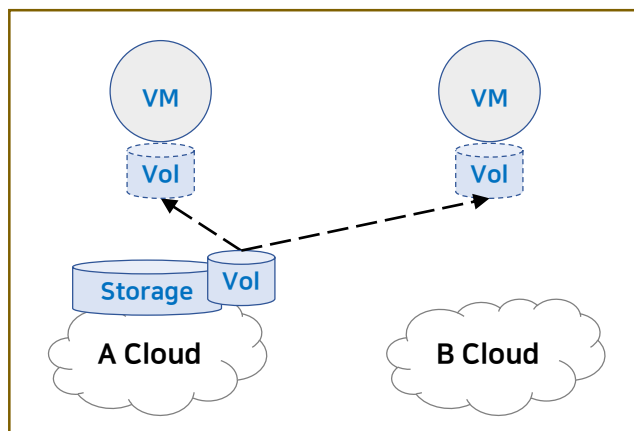
* 빠른 선적용으로 노하우 축적
* 노하우 축적 후 재평가
* 필요시 후보군 중 재선정
(POSIX 준수 → 교체 비용 적음)

<기술 개발 범위>

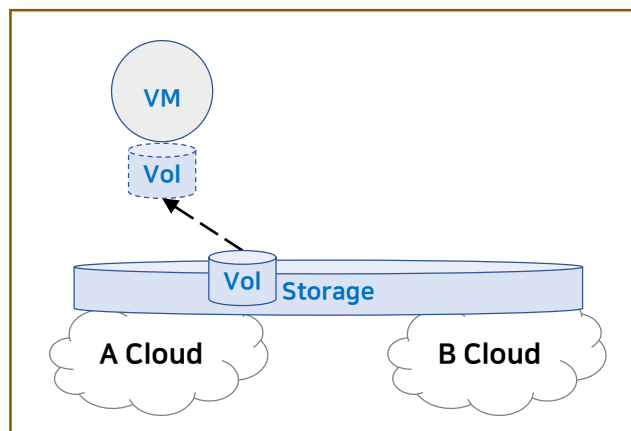
* One of Storages로 활용 가이드 제공
* 꼭, 필요한 경우에만 보수적 기술 개발 적용
* 손을 대면 → 교체 비용 증가

cb-storage POC 수행 결과

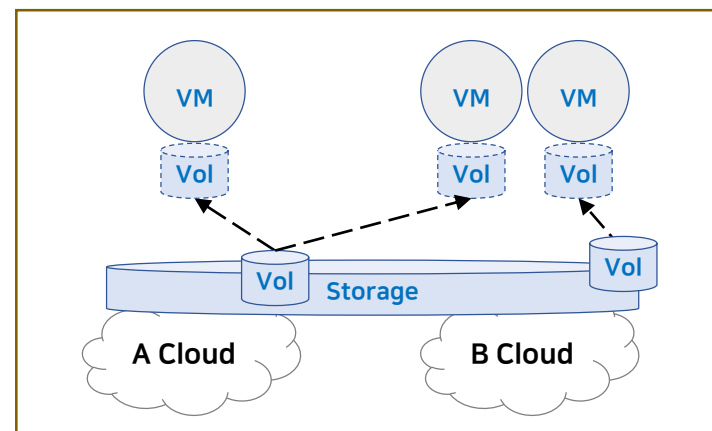
- * Test Case-I: 타 Cloud에서의 Mount가 가능한가 ? **가능함(별도 작업 필요)**
- * Test Case-II: 타 Cloud의 OSD(storage node) 연동이 가능한가? **OSD 등록 안됨 (방안 모색 중)**



<Test Case-I>



<Test Case-II>



<Goal>



세부 내용 및 결과는 적절한 시기에 공개 예정



멀티클라우드 서비스 공통 플랫폼

CB-Larva 향후 계획

카페모카(Café Mocha) 한잔 어떠세요 ?

글로벌 스케일 네트워킹: 요구사항 → CLADNet을 통해 해결 중 / 글로벌 스케일 스토리지: 첫 발을 내딛음

글로벌 스케일 네트워킹의
고려사항 및 요구사항



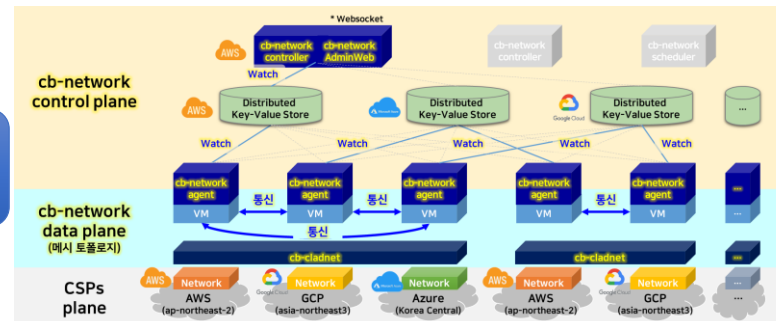
요구사항을 만족하기 위한
cb-network 시스템 구조, 기능, 데이터

글로벌 스케일 데이터 보존을 위한
cb-storage

(cb-network 상에서 운용된다면? ☺)



시스템
구조



기능

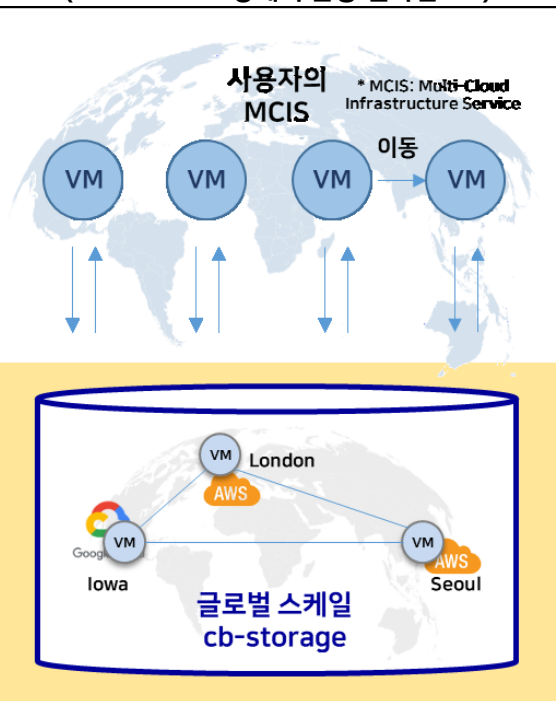


데이터
구조

cloud-adaptive-network/configuration-information/{group-id}
cloud-adaptive-network/host-network-information/{group-id}/{hostid}
cloud-adaptive-network/networking-rule/{group-id}
cloud-adaptive-network/status/test-specification/{group-id}
cloud-adaptive-network/status/information/{group-id}/{host-id}

Cloud-Barista Network (cb-network) 요구사항

- ✓ (다름) 서로 다르고, 변화 가능한 네트워크상의 VM이 하나의 네트워크 상에 존재하는 것 처럼 인식해야 함
- ✓ (위치 및 매체) 통신 성능을 최대한으로 이끌어 내야함
- ✓ (중요) 글로벌 스케일 네트워킹이 Fault Tolerance 해야함 (High availability (HA)보다 중요함)
- ✓ (중요) CSP 독립적인 네트워크를 구성 해야함 (Vendor lock-in 회피)
- ✓ (중요) VM의 네트워크 변화에 동적으로 대응할 수 있어야 함
- ✓ (기타) 여러 MCIS에 네트워크를 구분하여 할당 해야함
- ✓ (기타) 사용자(네트워크 전문가)를 위한 네트워크 할당 기능/인터페이스를 제공해야함



세부 연구 개발 현황 및 향후 계획

CLADNet 라이프사이클 연구 개발

CLADNet 운용 및 배포를 위한 고도화 (Usecase 적용, CB 플랫폼과 통합, 등을 위해)

cb-storage 연구 개발

완료

진행

예정

주요 업무	상세 업무	수행 내용	결과물 공개 수준	대상 버전
멀티클라우드 네트워킹 기술 (cb-network) 인큐베이팅	[설계] cb-cladnet 시나리오와 요구사항 정리 및 개선 (cladnet: 클라우드 적용형 네트워크(전 cb-subnet))	- TUN 기반 cb-subnet POC 분석 - 사용자 및 시스템 요구사항 정리 - 기관 및 내부의 추가 시나리오 명세 및 분석 (Logging, Tunneling TCP Packets, and etc.) - 사용자 및 시스템 요구사항 개선 (Multiple cb-cladnet, Name service)	개발문서/TM	-
	[분석 및 개발] cladnet 생성/구축 기술 개발	- 기존 cb-subnet과 VM 네트워크 충돌여부 검증 - 멀티 VM에 적용 가능한 네트워크 생성/구축 기술 개발	POC	-
	[POC/고도화] 네트워킹 정보 동기화 기술 개발	- 분산 Key-value store 스터디 및 적용 방안 도출 - 분산 Key-value store 기반 Networking rule 동기화 기술 (+ 네트워킹 정보 동기화 기술로 확대) POC	POC, TM	Café Mocha
	[POC/고도화] Tunneling 기능 개선/기술 교체	- TUN 기반 Tunneling의 이슈 분석 - Tunneling 기술 분석 및 적합한 기술 선정 (다중화 가능 기술 여부 반영, e.g., MPLS) - Tunneling 기술 POC	POC, TM	Affogato
	[cb-cladnet의 Usecase] 이중 CSP의 VM에 Application Runtime 글로벌 설치 및 운용	- CB Platform을 통한 Application Runtime(e.g., K8s) 설치 데모를 위한 테스트베드 구성 - cb-cladnet을 기반으로 Application Runtime 설치 수행 및 이슈 도출 - 이슈 해결 및 설치 반복 수행	설치 및 운용 DEMO (미확정)	Affogato
	[설계 및 개발] cb-cladnet 라이프사이클 제어 메커니즘 설계 및 개발	- cb-cladnet 라이프사이클(CRUD) 정의 (cb-subnet POC 기반으로) - cb-cladnet 라이프사이클에 대해 서비스 제공 필요 여부 검토(=API화 필요성 검토) - cb-cladnet 라이프사이클 제어 메커니즘 설계 - cb-cladnet 라이프사이클 제어 메커니즘 개발	대상버전릴리스	Affogato
	[검증] cb-cladnet 시험 및 안정화	- Precondition: CB-Spider 및 CB-Tumblebug 설치 및 구동 - cb-cladnet 라이프사이클 제어 메커니즘 시험 - cb-cladnet 라이프사이클 제어 메커니즘 안정화 수행 - 성숙도 확보 후, 기존 기술(예, WireGuard)과의 성능 벤치마킹 수행	대상버전릴리스	Affogato
	[설계 및 개발] 다중 cb-cladnet 기술 설계 및 개발	- MCIS 라이프사이클 분석 및 cb-cladnet와 연관성 도출 - 다중 MCIS를 위한 다중 cb-cladnet 기술 개발	개발문서/TM 대상버전릴리스	Affogato
	[검증] 다중MCIS를 위한 다중 cb-cladnet 시험 및 안정화	- 다중 MCIS 생성을 통한 다중 다중 cb-cladnet 시험 및 안정화	대상버전릴리스	Affogato
	[개발] 다중 cb-cladnet 대시보드	- 기존 cb-subnet 대시보드의 cb-cladnet AdminWeb화	대상버전릴리스	Café Mocha Affogato
멀티클라우드 데이터 관리 기술 (cb-storage) 인큐베이팅	[POC] cb-storage 설계 및 개발	- 유사 시스템 기술 분석 (대상: Alluxio, Quantcast, IPFS, MiniIO, SeaweedFS, HDFS) - 내/외부의 cb-storage 사용 시나리오 도출 및 요구사항 정리 - cb-storage 개발 및 검증 - (향후 계획) cb-storage 규격 정의 및 설계	(대상버전릴리스)	(Affogato)

Topics

Multi-cloud network

Multi-cloud storage

Event-driven architecture

Micro-service architecture

Golang

Web frontend

Distributed key-value store

Distributed storage



[참고] CB-Larva 저장소: <https://github.com/cloud-barista/cb-larva>

[참고] Cloud-Barista's Coffeehouse 저장소: <https://github.com/cloud-barista/cb-coffeehouse>

[참고] Cloud-Barista Community의 공개SW 활동 비전 및 영상: <https://youtu.be/J0wmFLMxc1w>

감사합니다.

<https://github.com/cloud-barista>
<https://cloud-barista.github.io>

(김 윤 곤 / contact-to-cloud-barista@googlegroups.com)

“멀티클라우드, 글로벌 스케일로 시작하다”

클라우드바리스타들의 네번째 이야기

Cloud-Barista Community the 4th Conference