



멀티클라우드, “새로운 생태계를 향한 클라우드 비긴어게인”

클라우드바리스타 커뮤니티 제5차 컨퍼런스

[세션] CB-Tumblebug:

멀티클라우드 인프라 서비스 통합 관리

손 석 호

CB-Tumblebug 프레임워크 리더

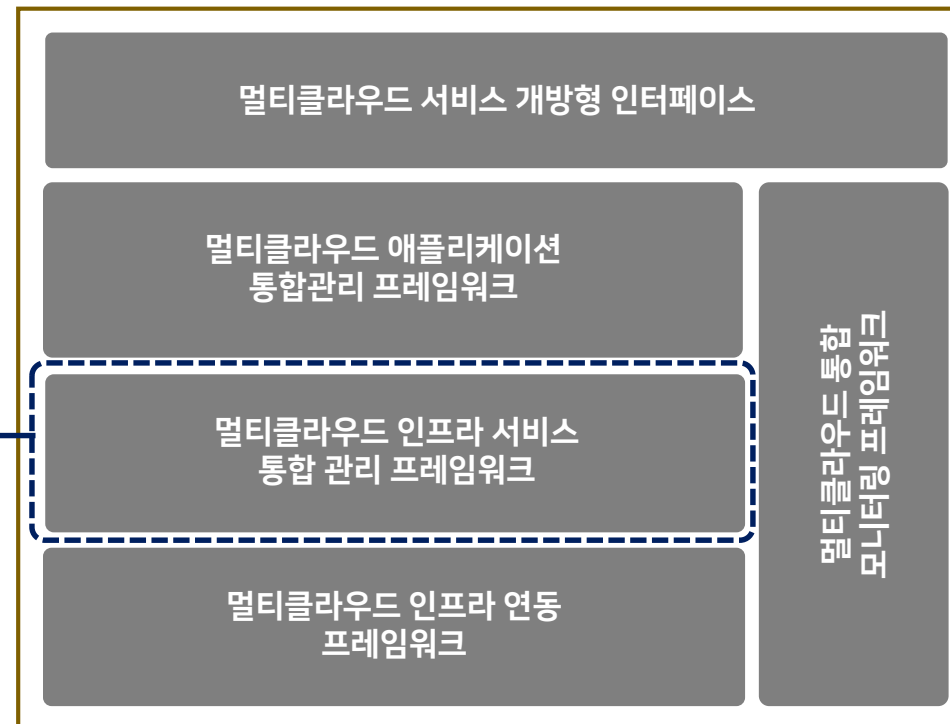
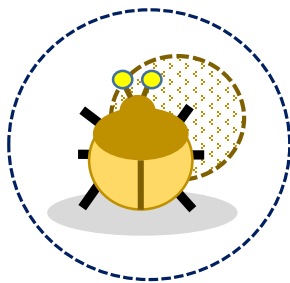
아포가토(Affogato) 한잔 어떠세요 ?

이번 세션은 ...

응용/도메인/기관 특화 SW



CB-Tumblebug



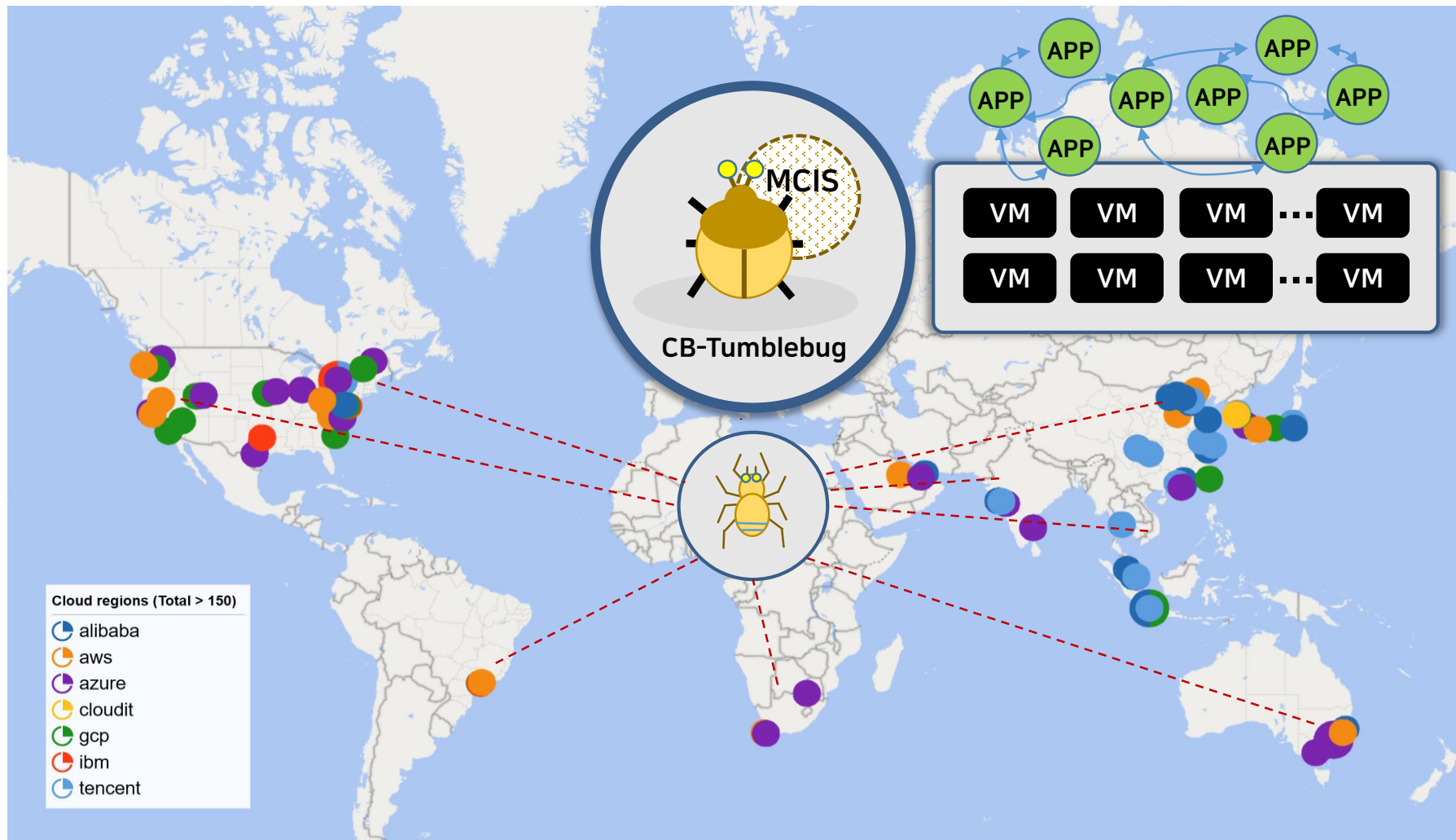
멀티클라우드 서비스 공통 플랫폼

I CB-Tumblebug 개요

II CB-Tumblebug 기능 및 구조 요약

III CB-Tumblebug 릴리스 주요 개선 사항

CB-Tumblebug: 멀티클라우드 인프라 서비스 통합 관리 필요성



멀티클라우드
애플리케이션

멀티클라우드 자원
통합 컴퓨팅 인프라

멀티클라우드 인프라 서비스 (MCIS)

컴퓨팅
자원

컴퓨팅
자원

컴퓨팅
자원

이중 멀티클라우드 연동

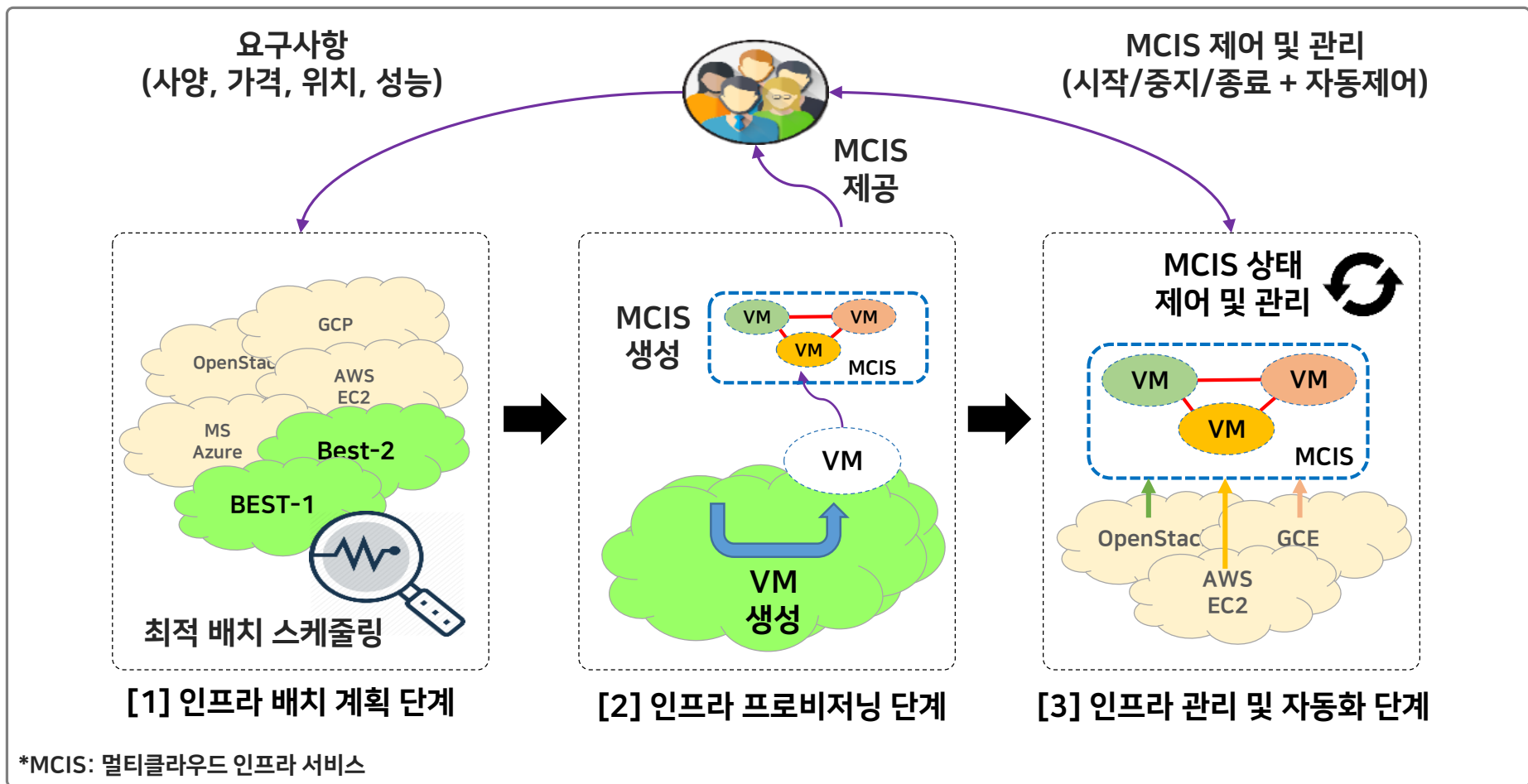


복잡한 멀티클라우드의 효과적인 활용을 위해, 자원을 유기적으로 통합 제공하는 멀티클라우드 인프라 서비스 통합 관리 필요

CB-Tumblebug: 멀티 클라우드 인프라 서비스 통합 관리 개념

기술 정의

사용자 요구사항에 따라 최적의 멀티 클라우드 인프라 서비스를 조합하여 프로비저닝하고,
통합 제어 및 관리를 통해 사용자의 컴퓨팅 인프라 운용을 지원하는 기술



성능 벤치마킹 기반 최적
멀티클라우드 인프라

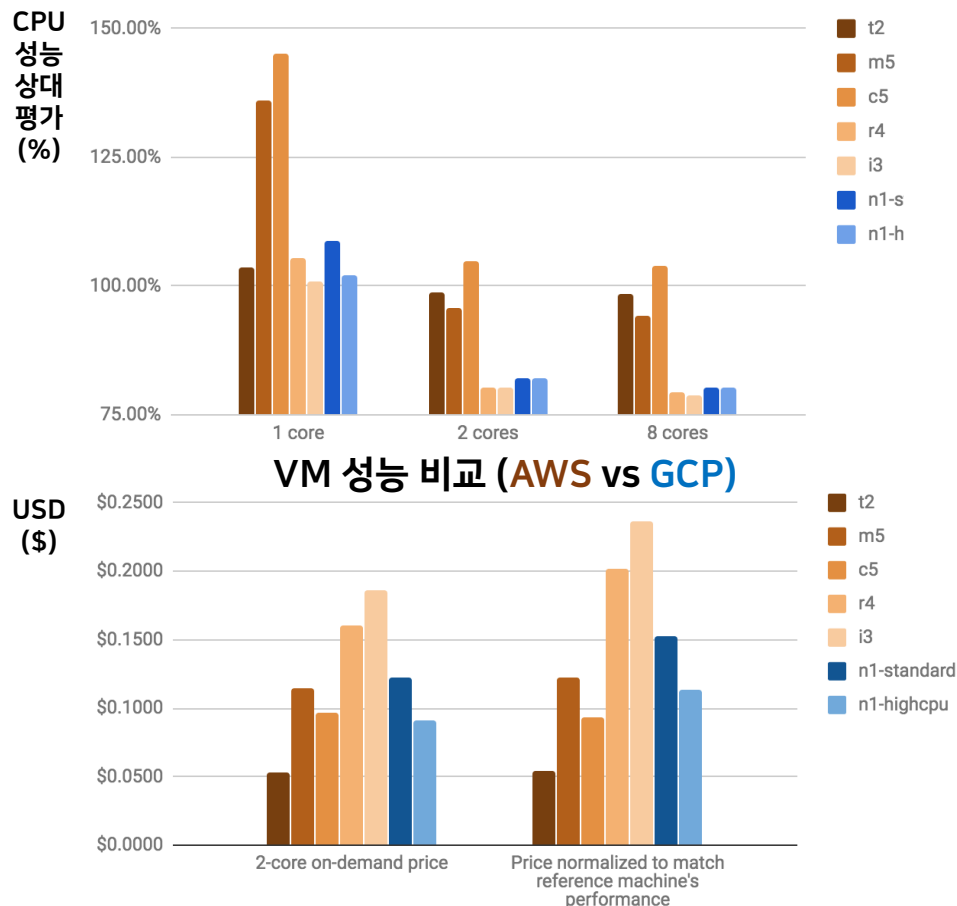
최적 클라우드 인프라
제공을 통한 효율성 증대

멀티클라우드 인프라
통합 운용 자동화

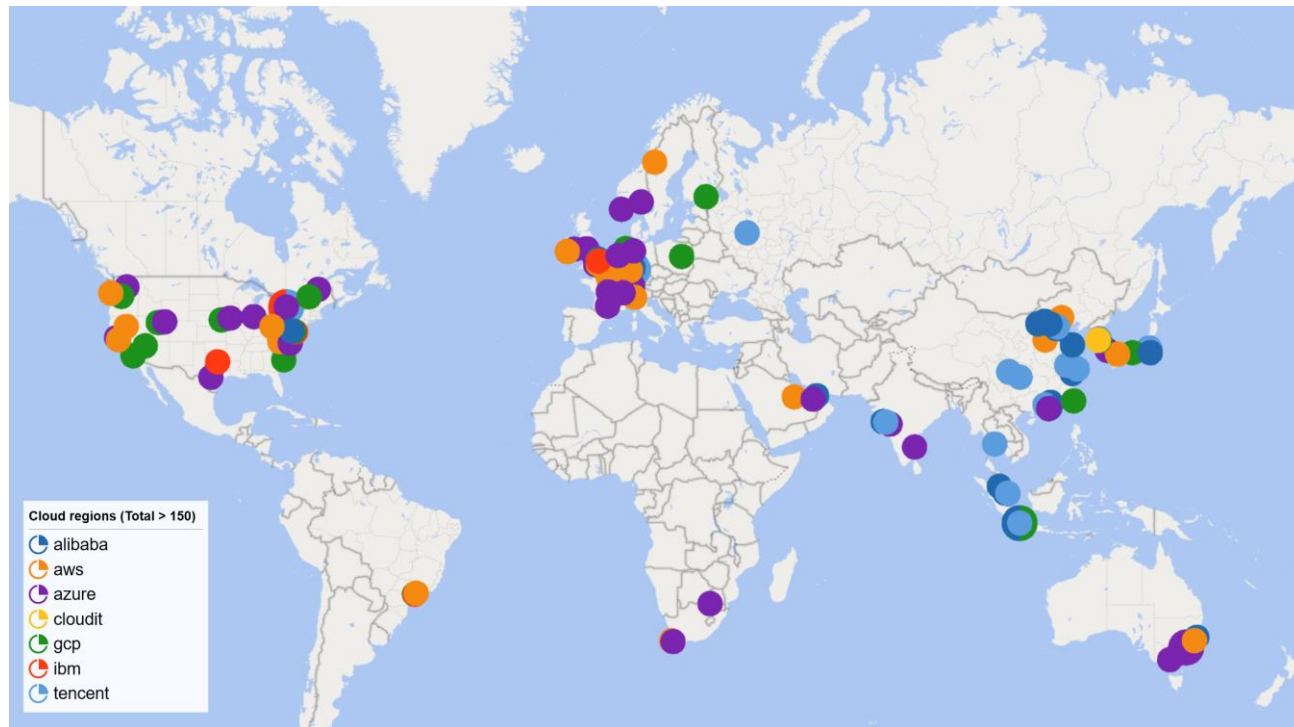
통합 제어, 정책 적용 등
관리 편의성 극대화

CB-Tumblebug 특징

(필요성) 1. 성능 벤치마킹 기반 최적 멀티 클라우드 인프라



Ref: [aws-vs-gcp-vs-on-premises-cpu-performance-comparison \(medium.com\)](https://medium.com/@aws-vs-gcp-vs-on-premises-cpu-performance-comparison)



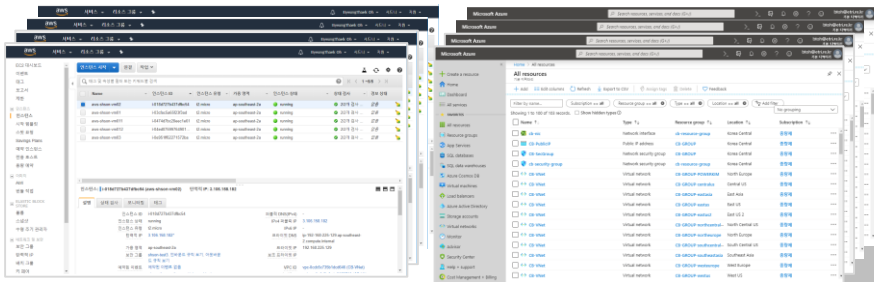
CSP의 클라우드 서비스 제공 리전(데이터센터) 현황

클라우드 서비스 성능.. 알고 보면 많이 달라요

클라우드에는 전세계에 퍼져 있습니다. 조금 더 가까이 다가가려면..

멀티 클라우드에서는 자원들의 성능 및 특성이 매우 다양하므로, 최적의 멀티 클라우드 인프라 서비스 제공 필요

(필요성) 2. 멀티 클라우드 인프라 통합 운용 자동화



수많은 VM을 각 클라우드 별로 개별 관리
(복잡성 증가)

- 수많은 VM 상태 한번에 확인 필요
- 수많은 VM 통합/자동 제어 필요

멀티 클라우드 애플리케이션

V
M

V
M

V
M

V
M

V
M

V
M

V
M

V
M

V
M

V
M

V
M

V
M

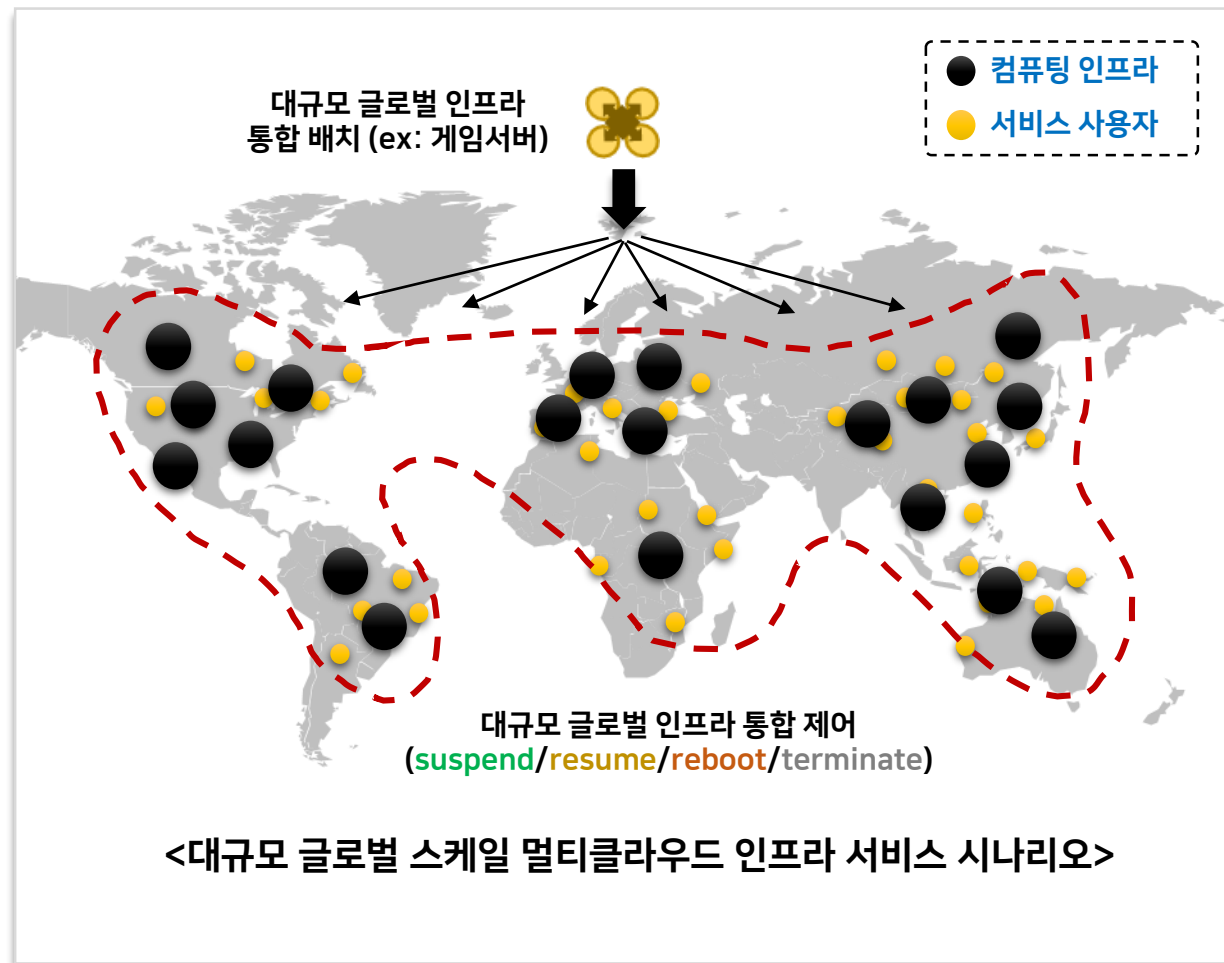
....

V
M

V
M

V
M

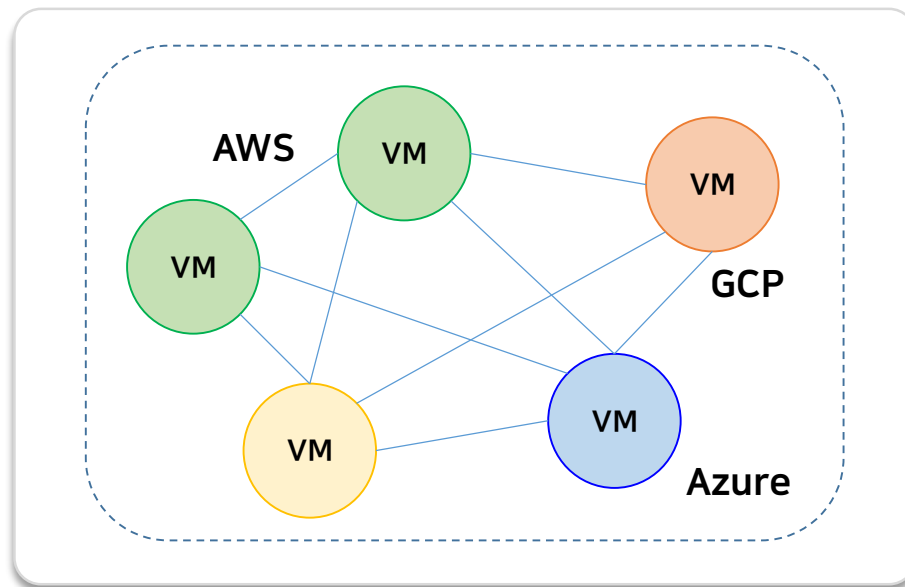
CB-Tumblebug 활용 사례



CB-Tumblebug 주요 개념 및 용어

• 멀티 클라우드 인프라 서비스 (MCIS)

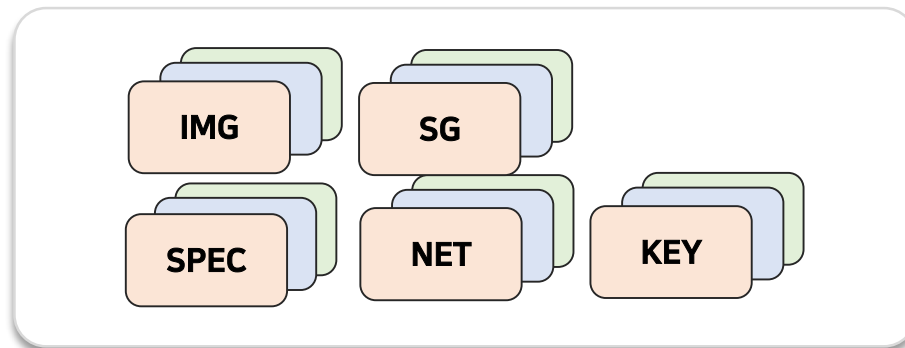
- 지역적으로 격리된 **다수의 클라우드 환경**에서 단일 목적(응용서비스, 애플리케이션 등)을 위해 **하나 이상의 클라우드 인프라 서비스**(가상머신 등)를 **조합 및 상호 연계한 컴퓨팅 인프라 그룹**
- 용도 : 멀티 클라우드 인프라의 통합 제어 및 관리



[MCIS 예시]

• 멀티 클라우드 인프라 리소스 (MCIR)

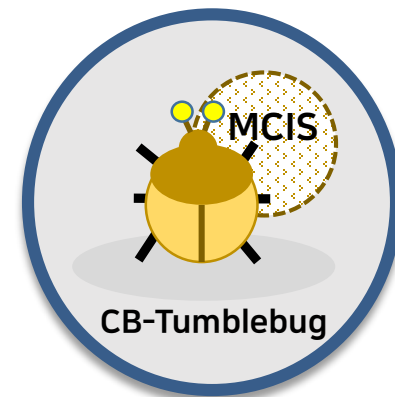
- 다수의 클라우드 환경에서 컴퓨팅 인프라 생성을 위해 관리하는 모든 리소스 (예: vNet, Image, AccessKey, ...)
- 용도 : MCIS 생성 및 설정을 위해 사용



[MCIR 예시]

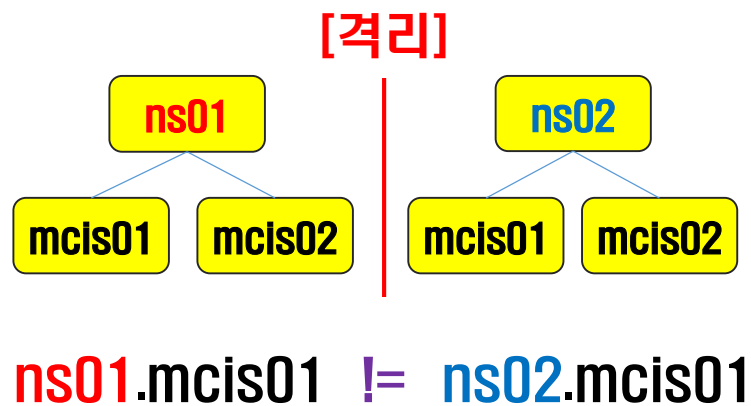
CB-Tumblebug 주요 기능 요약

- NS(네임스페이스) 및 MCIR 오브젝트 관리
 - 다양한 클라우드 인프라 자원을 오브젝트로 관리하는 기능
- MCIS 프로비저닝 및 특화 구성
 - 다양한 클라우드 자원을 활용하여 MCIS를 생성하고 특화하는 기능
- MCIS 라이프사이클 관리
 - MCIS 라이프사이클 상태를 종합적으로 관리, 통합 제어하는 기능
- MCIS 최적 구성 및 배치 스케줄링
 - 클라우드 평가를 통해 MCIS를 최적 구성하고 스케줄링 하는 기능
- MCIS 자동 제어
 - MCIS를 진단하고 결과에 따라 자동 제어하는 운용 자동화 기능

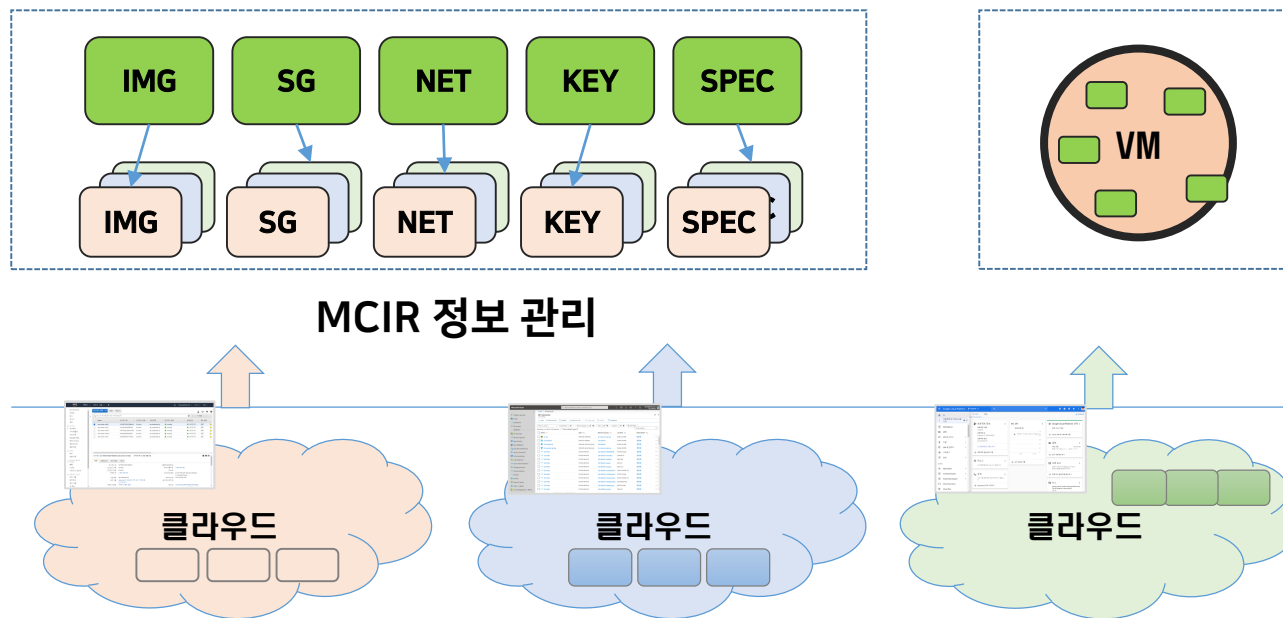


NS 및 MCIR 관리 기능

네임스페이스 관리

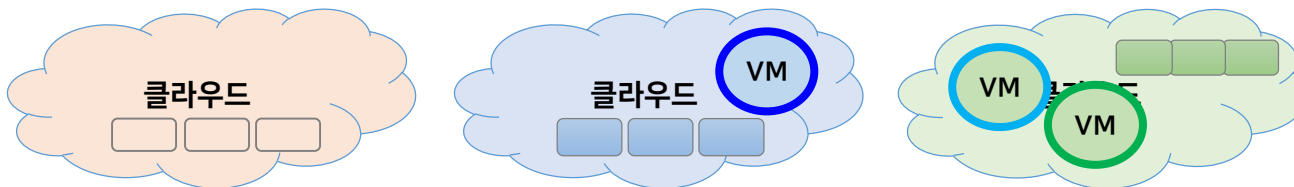
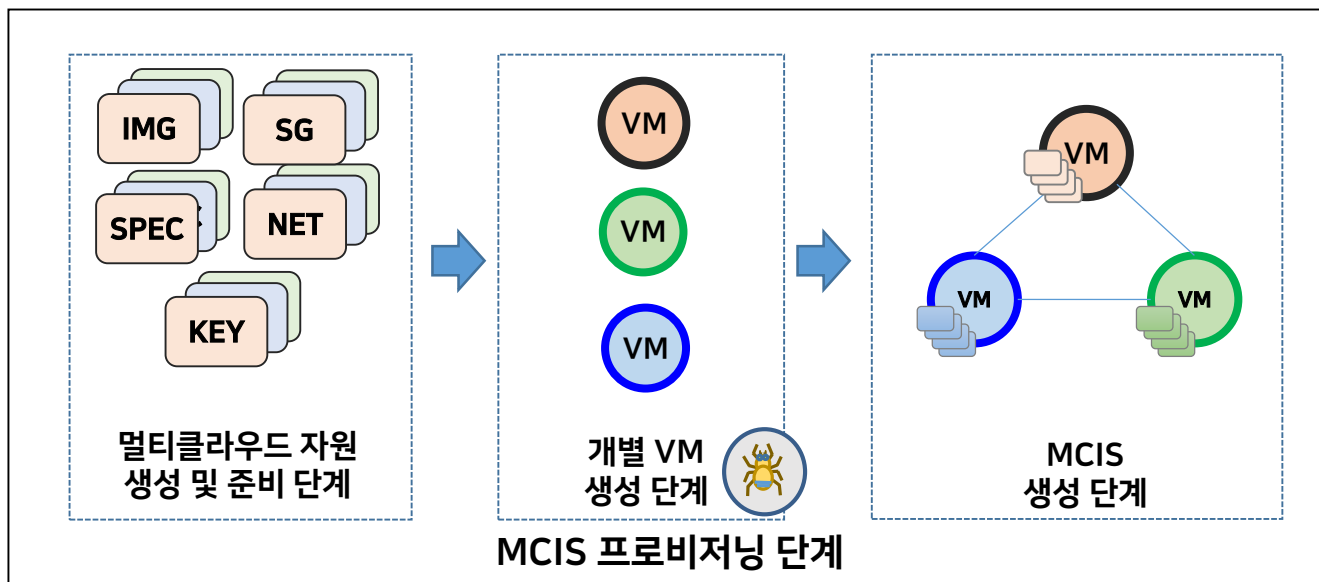


멀티 클라우드 인프라 자원 (MCIR) 관리

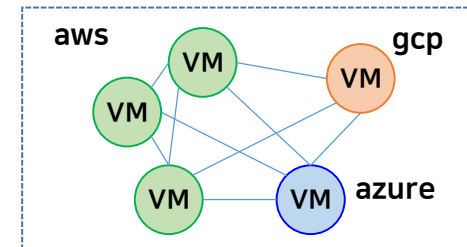


MCIS 프로비저닝 및 특화 구성

- 멀티클라우드 자원을 활용하여 MCIS를 생성하고 특화하는 기능

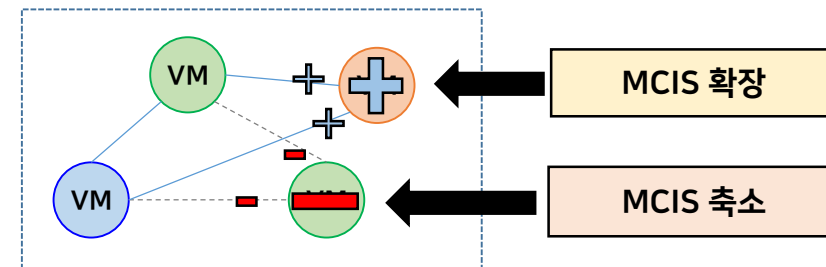


[MCIS 구성 예시]

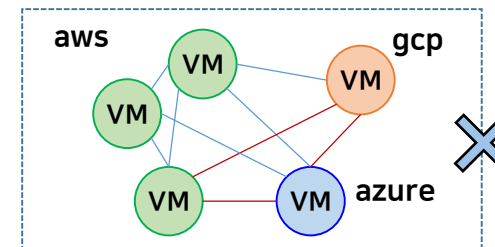


다양한 클라우드 자원을
통합 제어하기 위한
논리적 그룹 객체

[MCIS 구성 변경 예시]



[MCIS 특화 구성 예시]



VPN

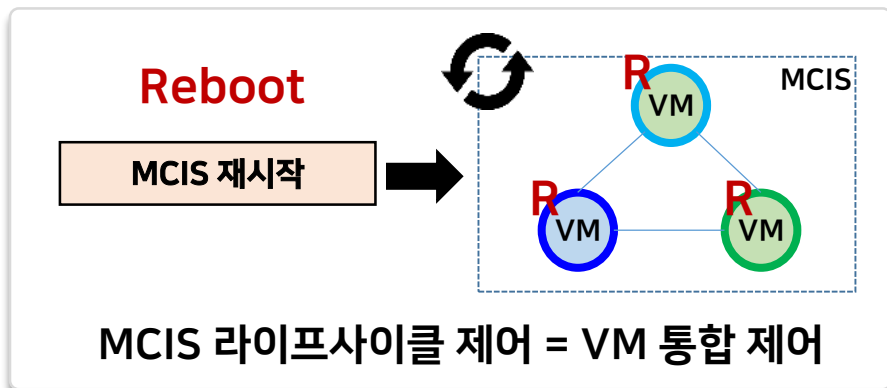
Load-Balancing

...

특화 기능

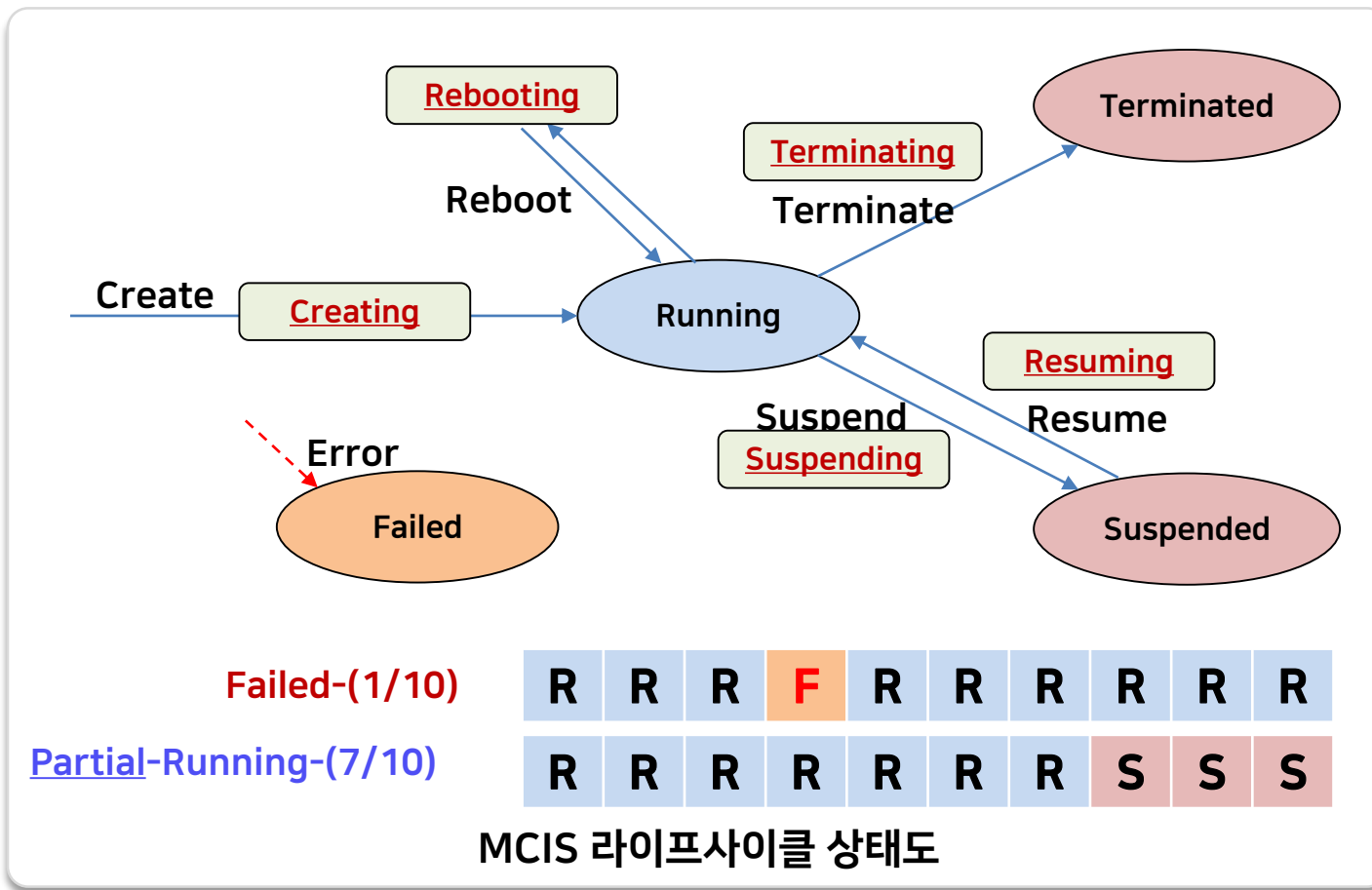
MCIS 라이프사이클 관리

- MCIS의 라이프사이클 통합 제어하고 상태를 쉽게 파악할 수 있는 기능
 - MCIS의 세부 요소들의 대표적인 상태를 한눈에 표시 (ex: Partial State)



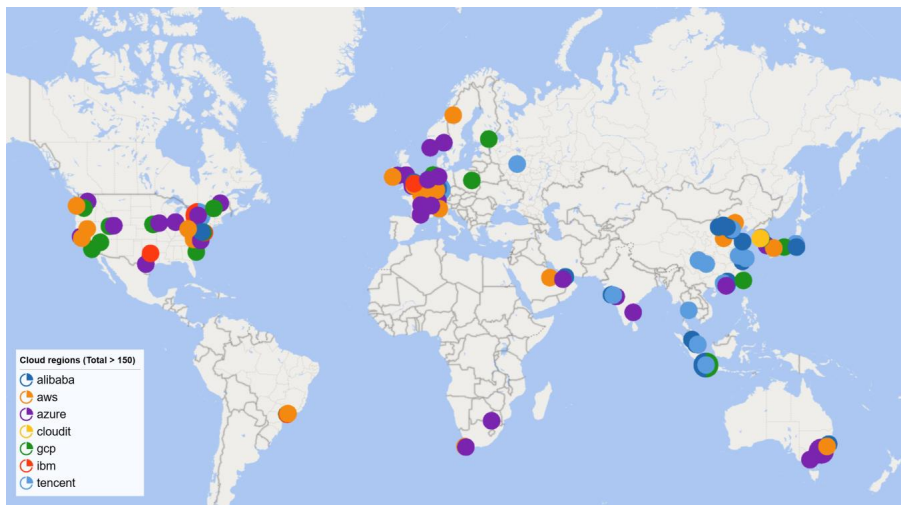
Cloud-Barista	Creating	Running	Suspending	Suspended	Resuming	Rebooting	Terminating	Terminated	Failed
Alibaba	Pending	Running	Stopping	Stopped	Resuming (자체생성상태)	Rebooting (자체생성상태)	Terminating (자체생성상태) - 삭제, suspending	Deleted	상태 정보를 얻을 수 없을 때 오류에 의한 비정상 상태
AWS	일정 보류로 (일정 Succeeded) (Pending 상태 확인 후 30 분 후 다시 확인)	running	stopping	stopped	Resuming (자체생성상태)	Rebooting	shutting-down	terminated	상태 정보를 얻을 수 없을 때 오류에 의한 비정상 상태
GCP	Provisioning staging	running	stopping	terminated	Resuming (자체생성 상태)	Rebooting (자체생성 (OP: Stop&Start) - restart)	Terminating (자체생성상태) (OP: Delete)	(예외) NotExist	상태 정보를 얻을 수 없을 때 오류에 의한 비정상 상태
Azure	starting /-	running /succeeded	stopping /-	stopped /succeeded	Resuming (자체생성상태) -> Creating...	Rebooting (자체생성상태) (OP: Stop&Start)	deallocating /-	(예외) NotExist	- /failed
OpenStack	BUILD	ACTIVE	Suspending (자체생성상태)	SHUTOFF	Resuming (자체생성상태)	REBOOT	Terminating (자체생성상태) (OP: Delete)	(예외) NotExist	Error
Cloudit	CREATING	RUNNING	STOPPING	STOPPED	STARTING	REBOOTING	DESTROYING	(예외) NotExist	FAILED

MCIS 라이프사이클 상태 검증 및 보장

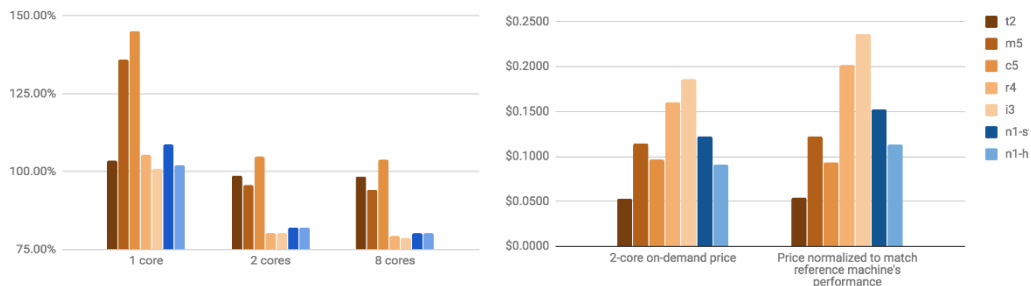


MCIS 최적 구성 및 배치 스케줄링

- 다양한 정보 수집을 통해 MCIS를 최적으로 구성하고 스케줄링
 - 멀티클라우드에는 다양한 클라우드를 활용하므로 최적 선정이 매우 중요



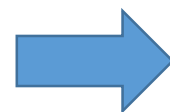
클라우드 리전 위치 (약 150개 리전)



VM 성능 비교 (AWS vs GCP)

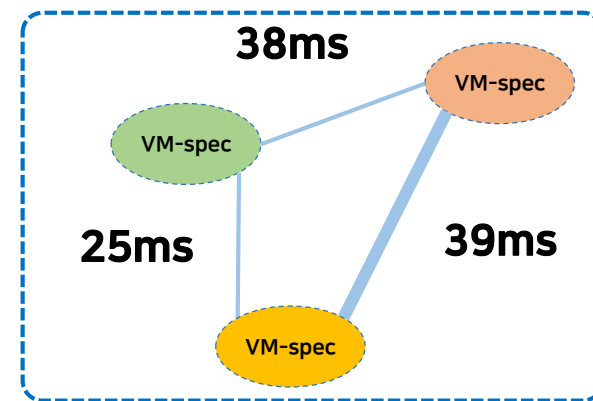
VM 가격 비교 (AWS vs GCP)

Source: <https://medium.com/infrastructure-adventures/aws-vs-gcp-vs-on-premises-cpu-performance-comparison>



MCIS 통합 최적 배치

- VM 간 응답속도 기반 배치



예) VM 간 응답시간 < 40ms

MCIS 개별 VM의 최적 배치

- VM 스펙 기반 배치
- VM 가격 기반 배치
- VM 위치 기반 배치
- VM 성능 기반 배치 (벤치마킹)
- VM 복합 조건 기반 배치

사용자 정책 기반 MCIS 자동 제어

- 사용자 정책을 기반으로 MCIS의 상태를 진단하고, 결과에 따라 MCIS를 자동 제어하는 기능

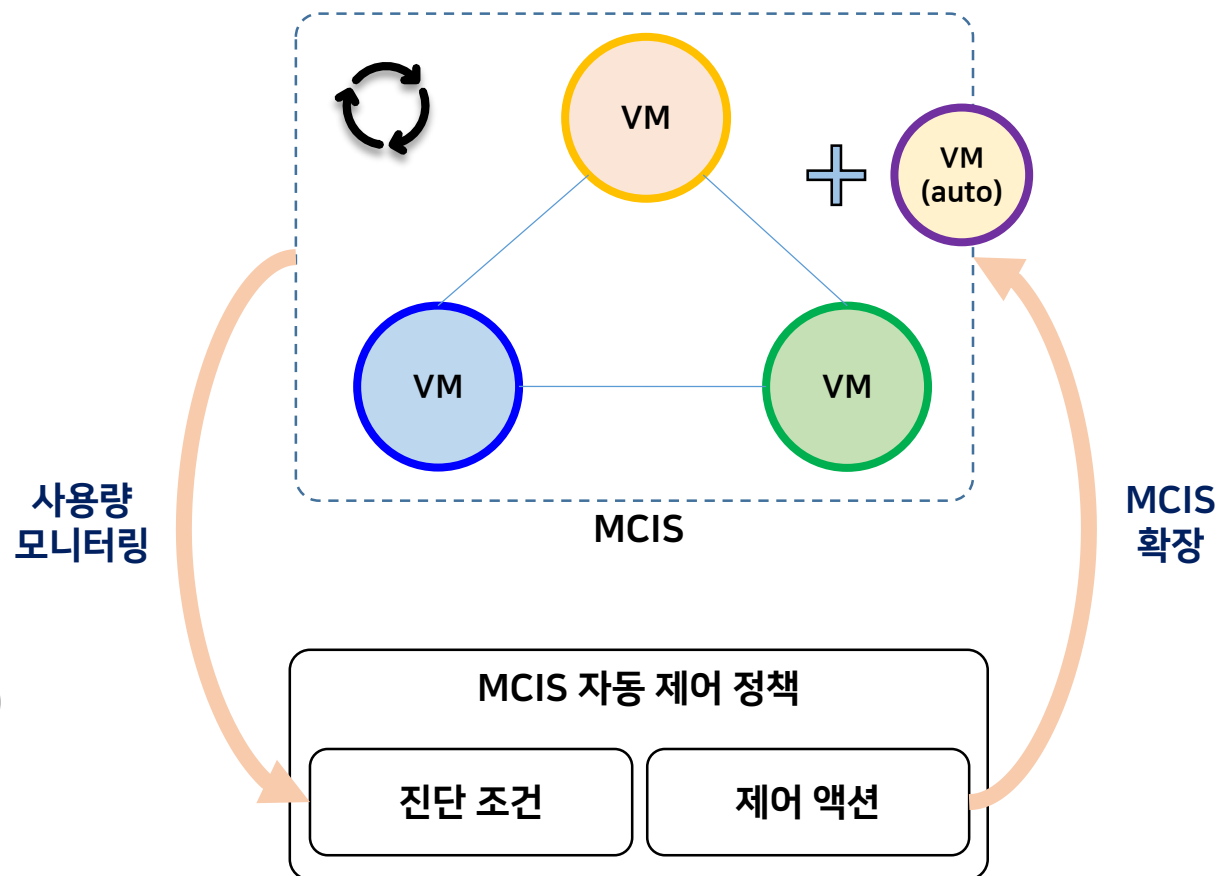
- 대규모 자원의 수동 처리 한계 극복

- MCIS 진단 조건

- MCIS 자원 사용량 진단
- MCIS 라이프사이클 진단
- MCIS 성능 진단
- ...

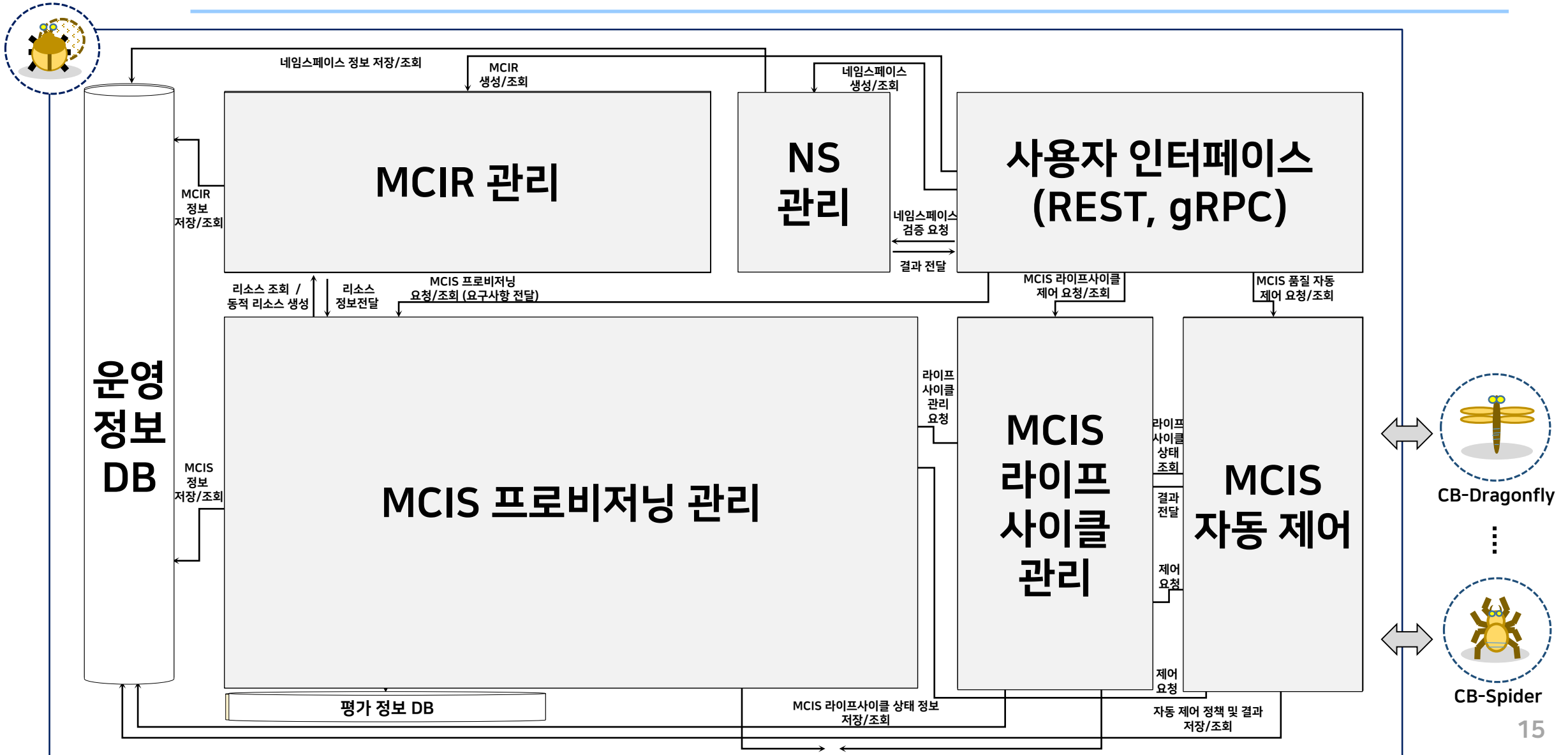
- MCIS 제어 액션

- MCIS 규모 제어 (Scale In/Out)
- MCIS 라이프사이클 제어 (Suspend/Resume)
- MCIS 자원 교체 (Replace)
- ...



CB-Tumblebug 시스템 구조

향후 주요 개선 포인트
(help wanted!)



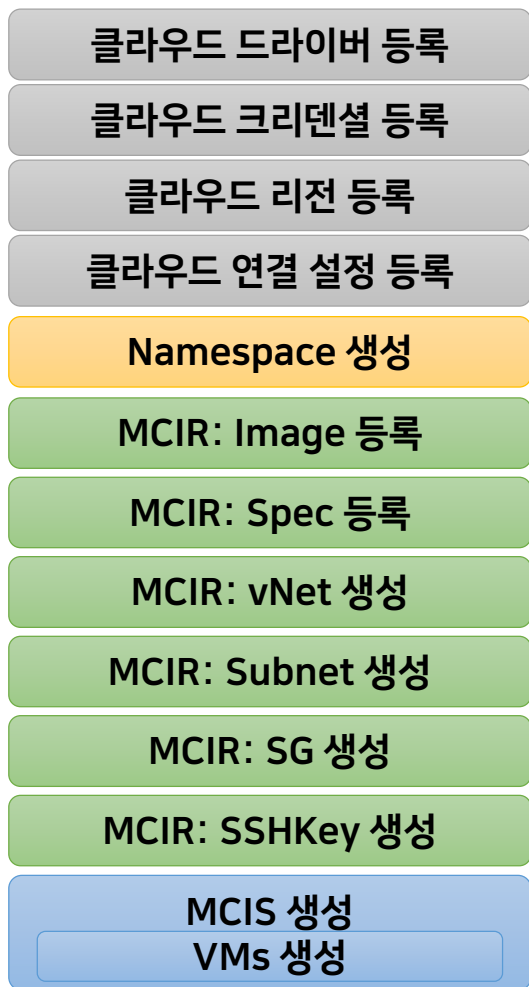
Affogato 주요 개선 포인트

동적 MCIS 프로비저닝

MCIS 최적 배치

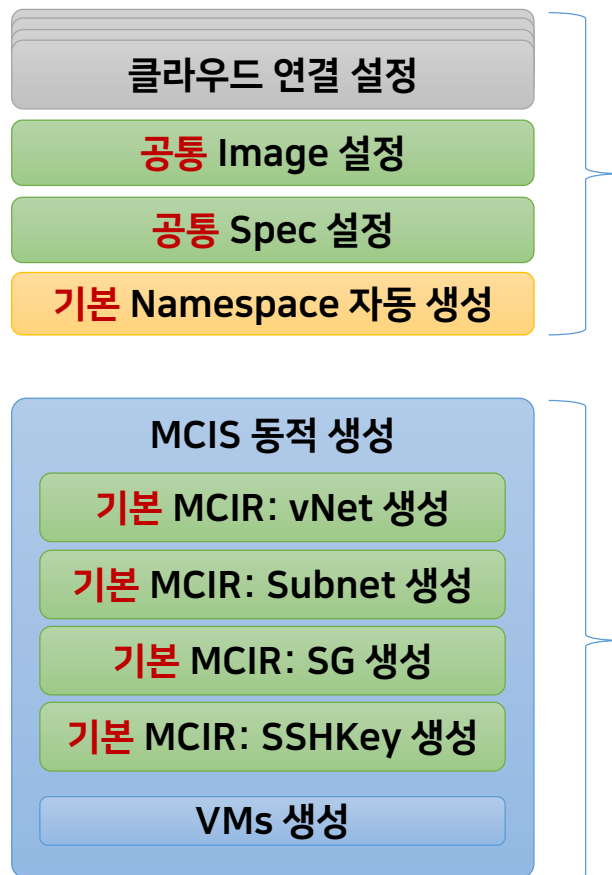
This time, we will not visit the details
CB-TB hopes to focus on "**How to use**"

MCIS 동적 프로비저닝 (프로비저닝 단계 간소화 및 자동화)



(기본) MCIS 프로비저닝 과정

호출 API: 12개



(동적) MCIS 프로비저닝 과정

호출 API: 1개

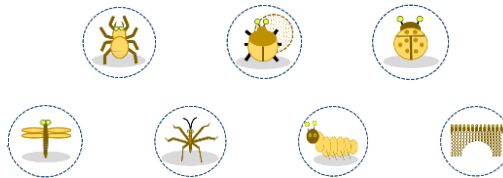
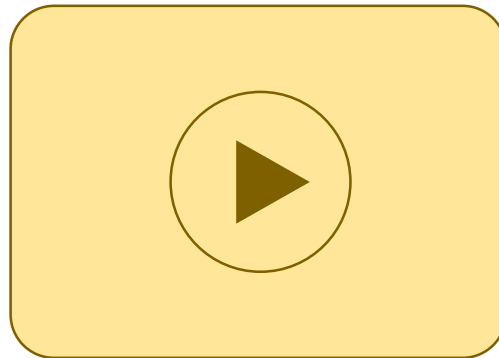
CB-TB 시스템 환경 설정 (script)

동작 검증된 image, spec 활용
[cb-tumblebug/assets/cloudimage.csv](https://github.com/cb-tumblebug/assets/cloudimage.csv)
[cb-tumblebug/assets/cloudspec.csv](https://github.com/cb-tumblebug/assets/cloudspec.csv)

MCIS 생성 과정에서
자동으로 기본 MCIR 생성

MCIS 동적 프로비저닝 시연

동영상 시연이 준비되어 있습니다.



세션을 놓치셨다면? 걱정하지 마세요! Cloud-Barista 커뮤니티 YouTube 채널에서 확인하실 수 있습니다 ^^

 YouTube : Cloud-Barista 커뮤니티(<https://cloud-barista.github.io/youtube>)

Affogato 주요 개선 포인트

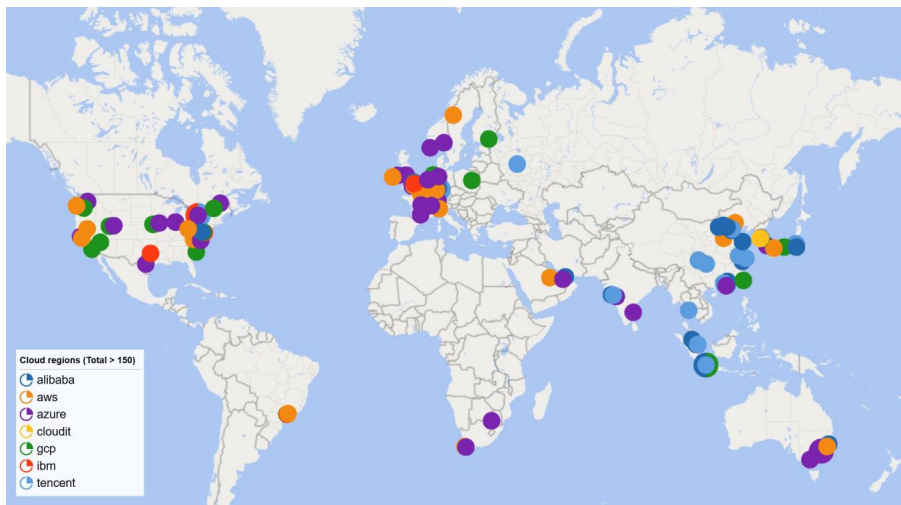
동적 MCIS 프로비저닝

MCIS 최적 배치

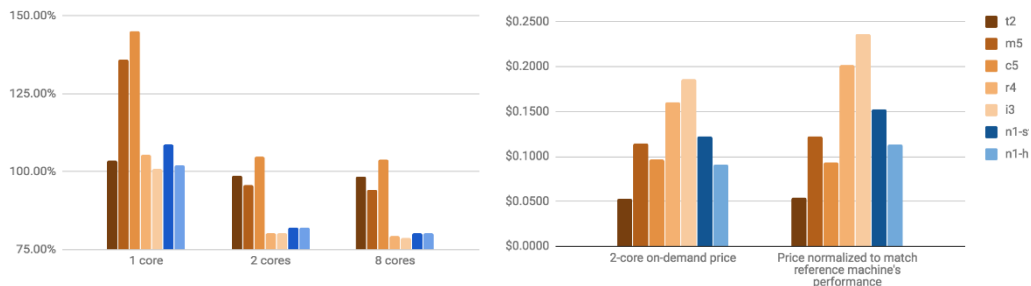
This time, we will not visit the details
CB-TB hopes to focus on "**How to use**"

MCIS 최적 구성 및 배치 스케줄링

- 다양한 정보 수집을 통해 MCIS를 최적으로 구성하고 스케줄링
 - 멀티클라우드는 다양한 클라우드를 활용하므로 최적 선정이 매우 중요



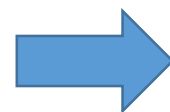
클라우드 리전 위치 (약 150개 리전)



VM 성능 비교 (AWS vs GCP)

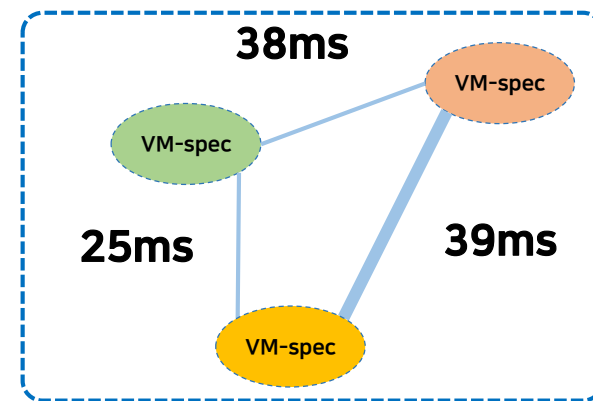
VM 가격 비교 (AWS vs GCP)

Source: <https://medium.com/infrastructure-adventures/aws-vs-gcp-vs-on-premises-cpu-performance-comparison>



MCIS 통합 최적 배치

- VM 간 응답속도 기반 배치



예) VM 간 응답시간 < 40ms

MCIS 개별 VM의 최적 배치

- VM 스펙 기반 배치
- VM 가격 기반 배치
- VM 위치 기반 배치
- VM 성능 기반 배치 (벤치마킹)
- VM 복합 조건 기반 배치

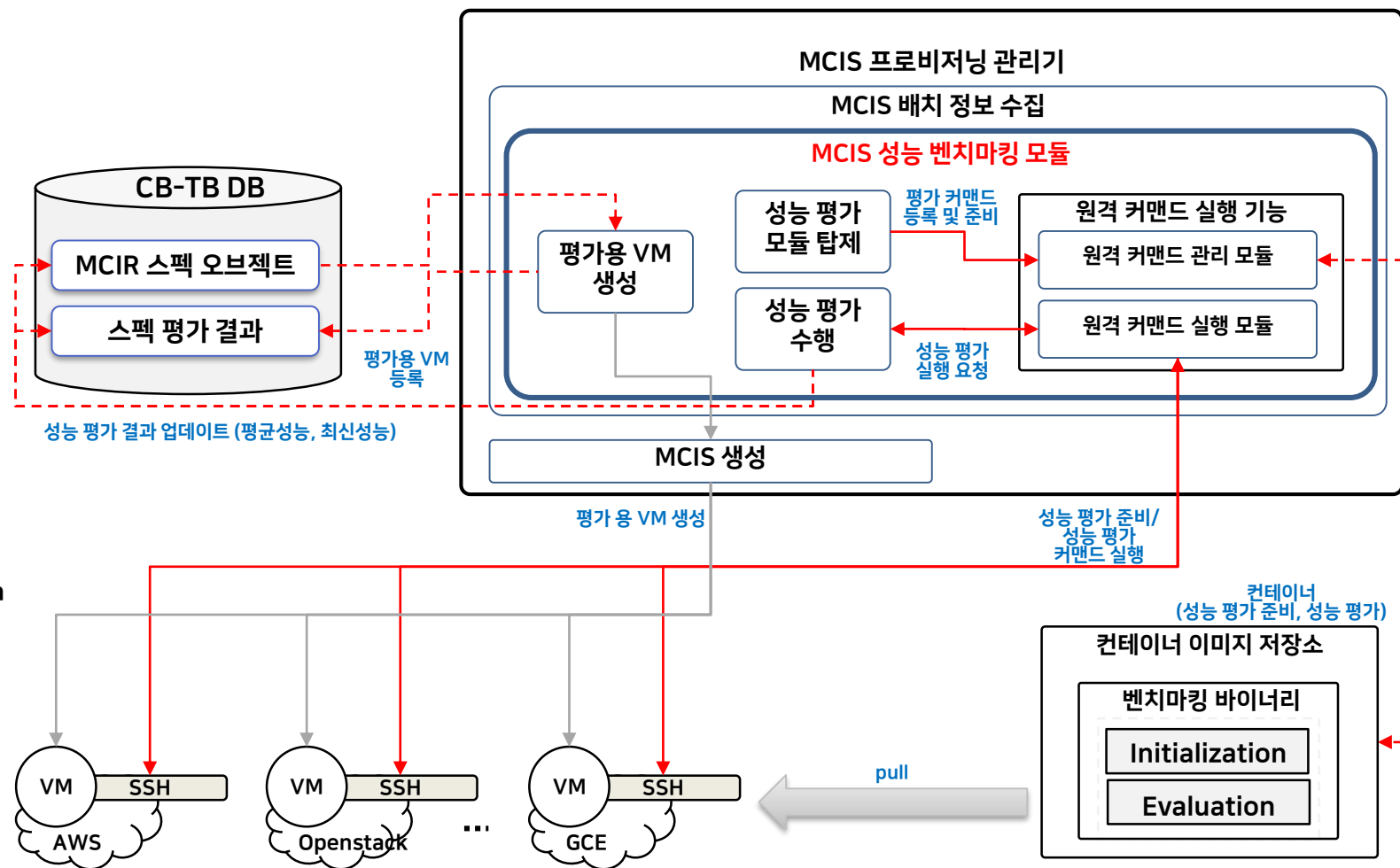
최적 MCIS 배치를 위한 동적 성능 벤치마킹 방식

에이전트 기반의 동적 성능 벤치마킹

1. 평가용 MCIS 생성
2. 주기적으로 에이전트에 평가 요청
3. 각 에이전트는 성능 평가를 수행하고 결과를 CB-TB에게 통보
4. 평가 결과를 테이블 형태로 수집

동적 성능 평가 지표

- CPU 계산속도 (싱글/멀티 코어)
 - Memory 처리속도 (읽기/쓰기)
 - FileIO 처리속도 평가 (읽기/쓰기)
 - DB 처리속도 평가 (읽기/쓰기)
 - 목적지까지 네트워크 지연 시간
 - VM간 상호 네트워크 지연 시간
- Sysbench
- Ping

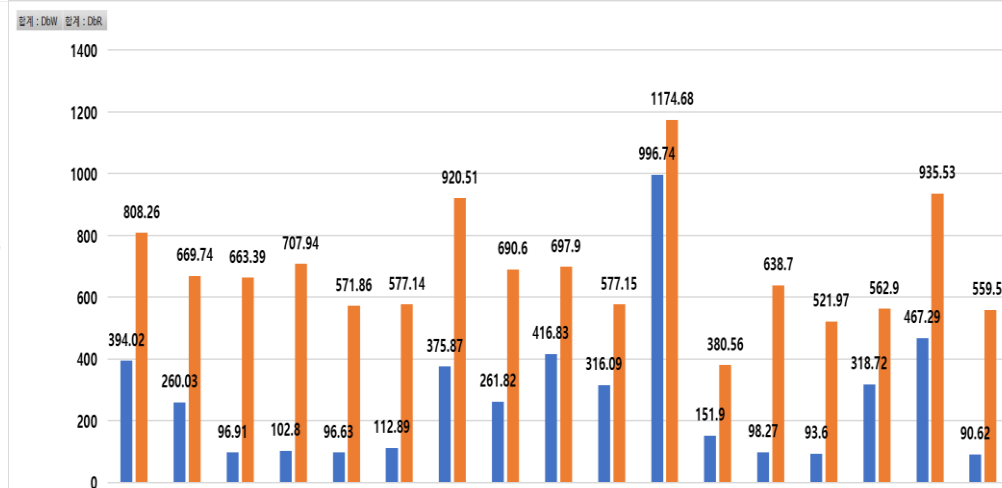
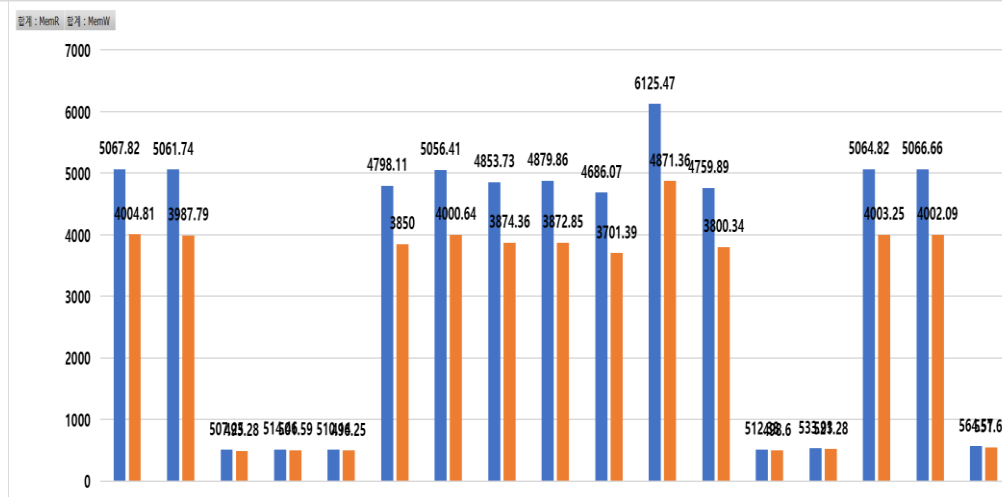
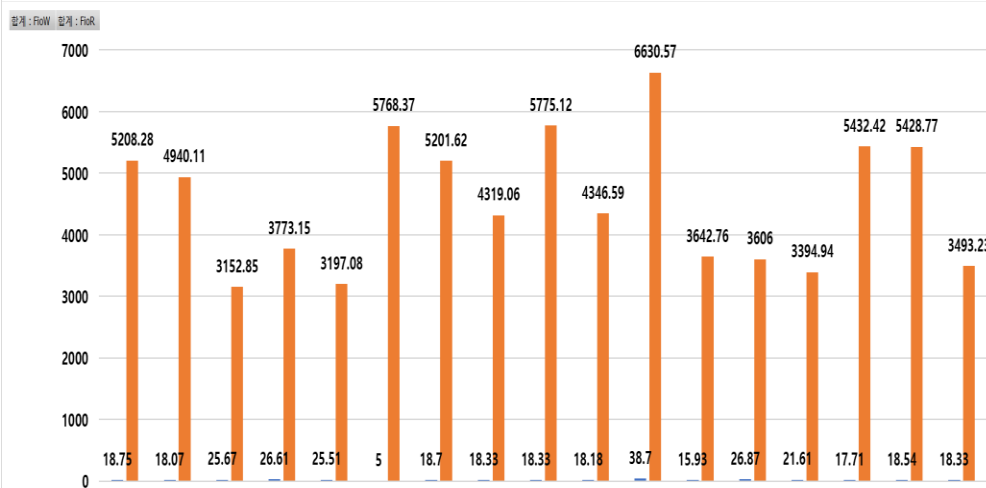
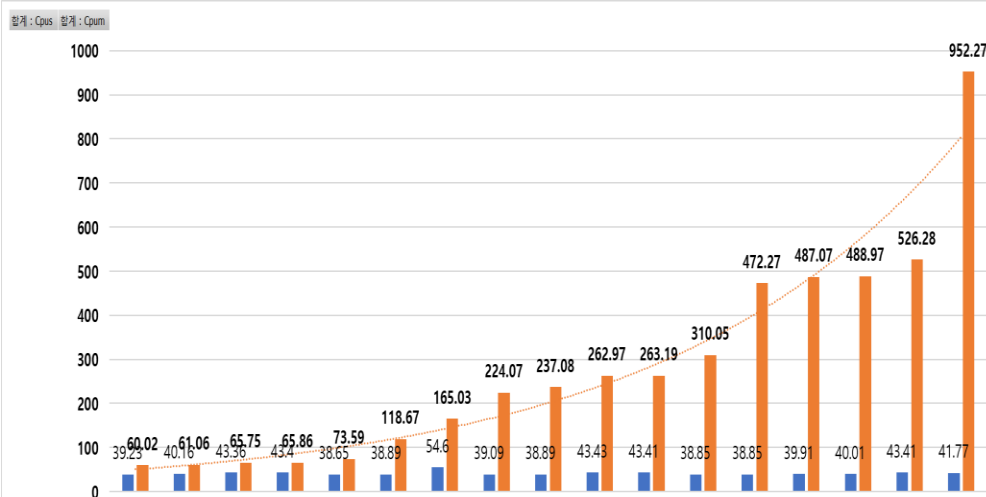


동적 성능 평가 결과 (예: CPU, Memory, FIO, DB 성능)

멀티 클라우드 VM 타입 (지역구분)

gcp-asia-east1-e2-standard-8
gcp-asia-east1-e2-highcpu-2
aws-us-east-1-m4.xlarge
aws-us-east-1-f1.2xlarge
aws-us-east-1-m4.xlarge
azure-koreacentral-Standard_D11_v2
gcp-asia-east1-e2-highcpu-8
gcp-asia-east1-e2-highmem-16
gcp-asia-east1-n1-standard-16
gcp-europe-west3-e2-highcpu-2
aws-ap-northeast-2-c5.xlarge
gcp-asia-east1-e2-highmem-2
aws-us-east-1-t2.2xlarge
aws-us-east-1-i2.2xlarge
gcp-asia-east1-e2-medium
gcp-asia-east1-e2-highcpu-16
aws-us-east-1-c3.8xlarge

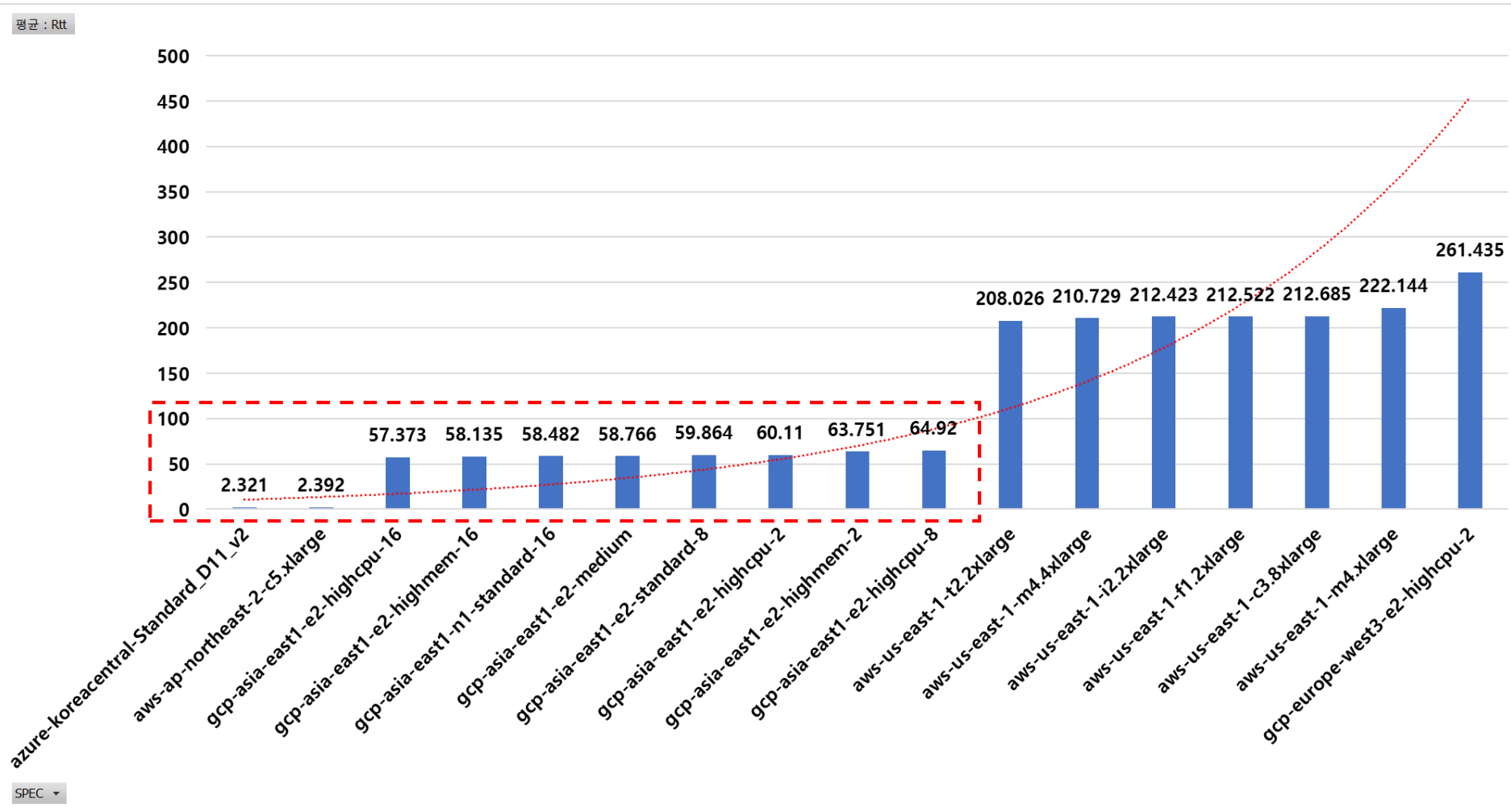
- CPU
- Memory
- FIO
- DB



X축을 CPU의 결과를 기준으로 나열하였으나, 높은 CPU 성능이 다른 지표에 대한 높은 성능을 보장하지 않음
(사용자 상황에 맞는 사양 선택 필요 !!)

동적 성능 평가 결과 (예: 한국 지역 기준 응답 속도)

행 레이블	평균 : Rtt
azure-koreacentral-Standard_D11_v2	2.321
aws-ap-northeast-2-c5.xlarge	2.392
gcp-asia-east1-e2-highcpu-16	57.373
gcp-asia-east1-e2-highmem-16	58.135
gcp-asia-east1-n1-standard-16	58.482
gcp-asia-east1-e2-medium	58.766
gcp-asia-east1-e2-standard-8	59.864
gcp-asia-east1-e2-highcpu-2	60.11
gcp-asia-east1-e2-highmem-2	63.751
gcp-asia-east1-e2-highcpu-8	64.92
aws-us-east-1-t2.2xlarge	208.026
aws-us-east-1-m4.4xlarge	210.729
aws-us-east-1-i2.2xlarge	212.423
aws-us-east-1-f1.2xlarge	212.522
aws-us-east-1-c3.8xlarge	212.685
aws-us-east-1-m4.xlarge	222.144
gcp-europe-west3-e2-highcpu-2	261.435
총합계	119.18



- 특정 IP를 기준으로 응답속도 평가

MCIS 최적 배치 요청 방법

- MCIS 최적 배치 요청 처리 방식
 - Filtering (필터링) & Prioritizing (우선순위 정렬) 조합
 - 필수 조건으로 필터링
 - 우선순위로 정렬

- Affogato 릴리스 지원 파라미터
 - Filtering
 - vCPU 수, Memory 용량
 - 가격
 - Prioritizing
 - 가격
 - 위치: 지리적 근접성
 - 성능: CPU, Memory, FileIO, DB

VM Spec List

gcp-asia-east1-e2-standard-8
 gcp-asia-east1-e2-highcpu-2
 aws-us-east-1-m4.4xlarge
 aws-us-east-1-f1.2xlarge
 aws-us-east-1-m4.xlarge

1) Filtering

azure-koreacentral-Standard_D11_v2	5
gcp-asia-east1-e2-highcpu-8	6
gcp-asia-east2-e2-highmem-16	1
gcp-asia-east1-n1-standard-16	7
gcp-asia-east3-e2-highcpu-16	4
aws-ap-northeast-2-c5.xlarge	2
aws-us-east-1-i2.2xlarge	3

2) Prioritizing

aws-us-east-1-t2.2xlarge
 gcp-asia-east1-e2-highmem-2
 gcp-asia-east1-e2-medium
 gcp-europe-west3-e2-highcpu-2
 ...

MCIS 최적 배치 요청 예시

Filtering

정책 1

Metric: CPU /
조건: ≥ 8 core

정책 2

Metric: Mem /
조건: ≥ 1 GB

정책 3

Metric: Cost /
조건: $\leq \$0.73$

Prioritizing

정책 1

Metric: 위치

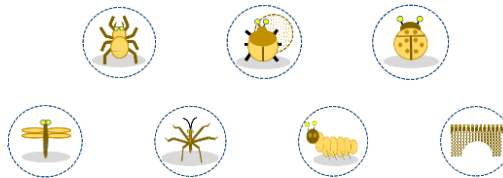
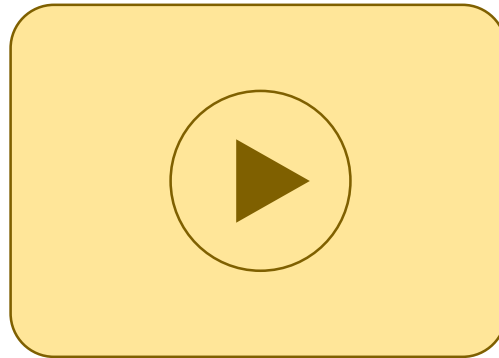
Key: 지리적 근접성
Value: 위도/경도

정책 2

...

CB-MapUI 활용 MCIS 최적 배치 시연

동영상 시연이 준비되어 있습니다.



세션을 놓치셨다면? 걱정하지 마세요! Cloud-Barista 커뮤니티 YouTube 채널에서 확인하실 수 있습니다 ^^

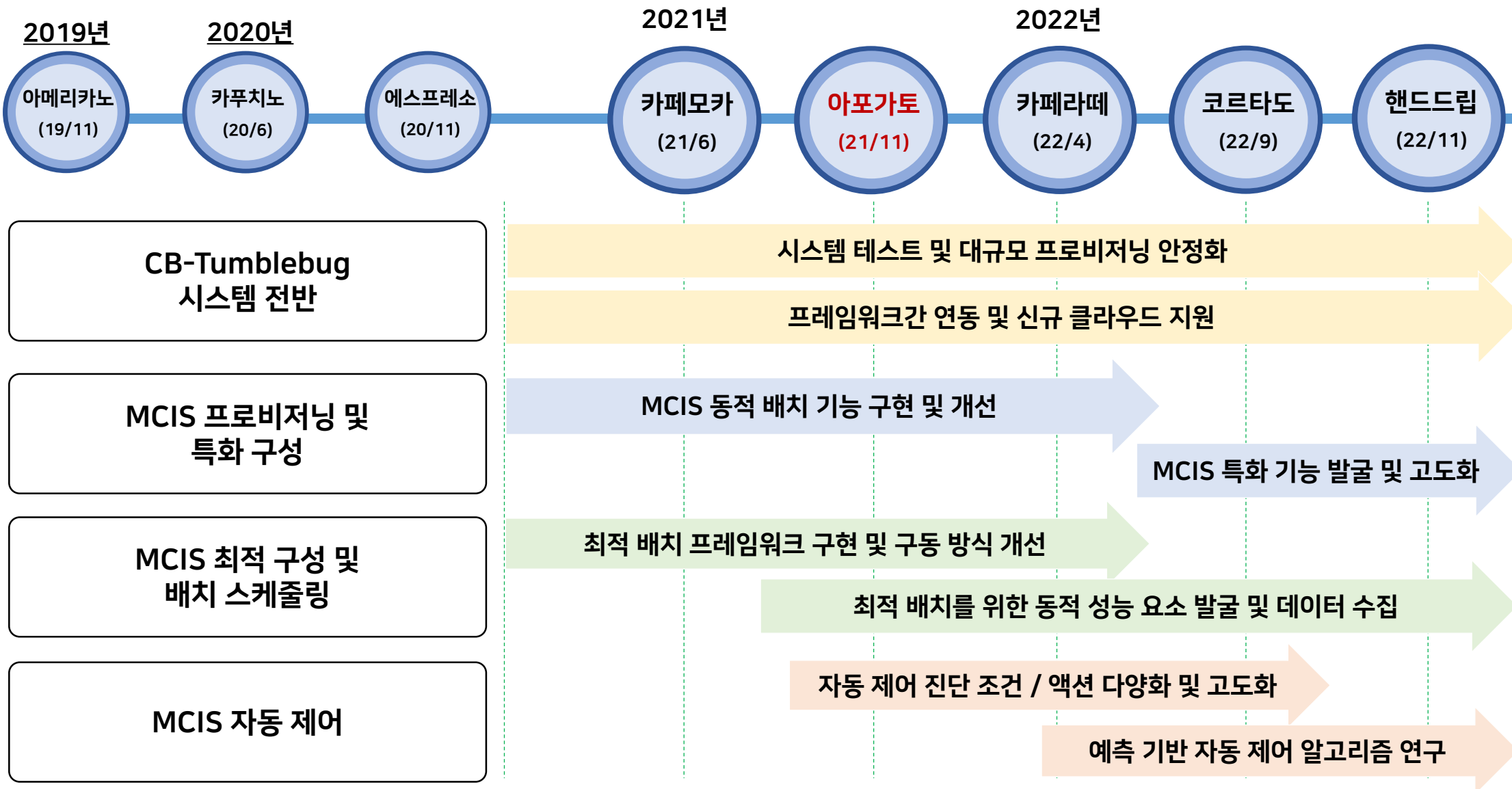
 YouTube : Cloud-Barista 커뮤니티(<https://cloud-barista.github.io/youtube>)

CB-Tumblebug 개발 로드맵

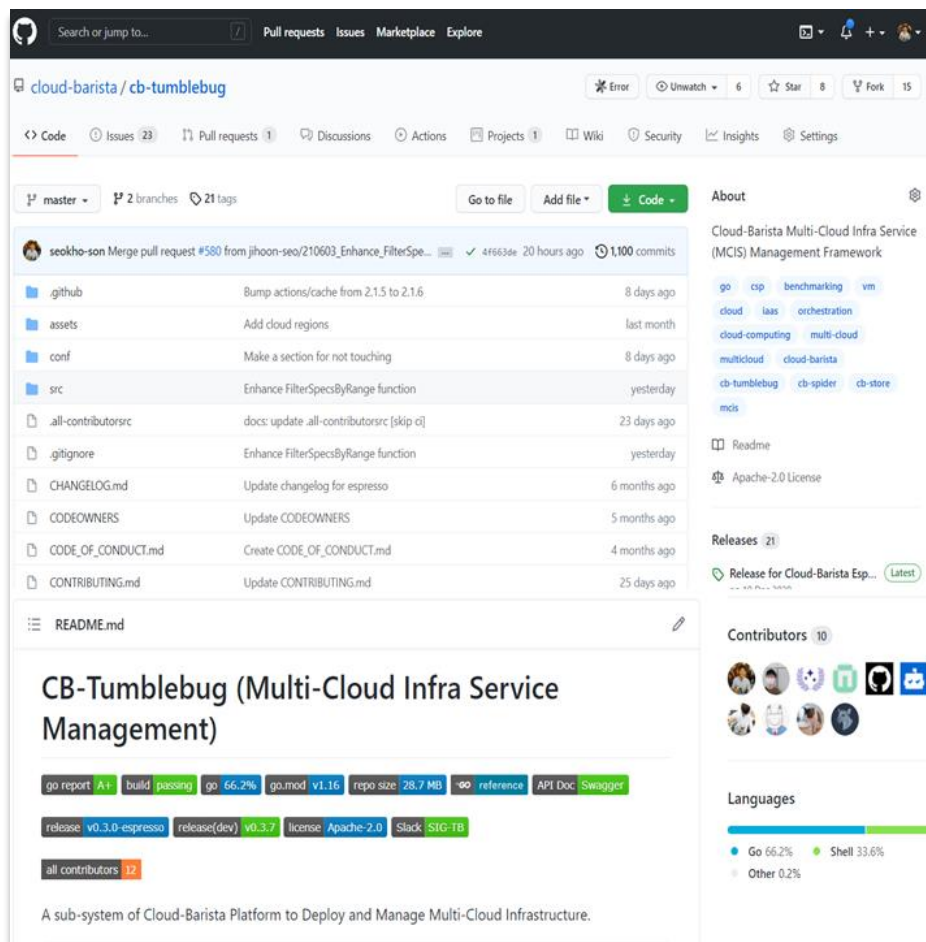
Remark: in Affogato release

동적 MCIS 프로비저닝

MCIS 최적 배치



[참고] CB-Tumblebug 참여 방법 (환영합니다..^^)



cloud-barista / cb-tumblebug

Search or jump to... Pull requests Issues Marketplace Explore

Code Issues (23) Pull requests (1) Discussions Actions Projects (1) Wiki Security Insights Settings

master 2 branches 21 tags

Go to file Add file Code

About

Cloud-Barista Multi-Cloud Infra Service (MCIS) Management Framework

seokho-son Merge pull request #580 from jihoon-seo/Enhance_FilterSpe... ✓ 4f636e 20 hours ago 1,100 commits

.github Bump actions/cache from 2.1.5 to 2.1.6 8 days ago

assets Add cloud regions. last month

conf Make a section for not touching 8 days ago

src Enhance FilterSpecsByRange function yesterday

.all-contributorsrc docs: update .all-contributorsrc [skip ci] 23 days ago

.gitignore Enhance FilterSpecsByRange function yesterday

CHANGELOG.md Update changelog for espresso 6 months ago

CODEOWNERS Update CODEOWNERS 5 months ago

CODE_OF_CONDUCT.md Create CODE_OF_CONDUCT.md 4 months ago

CONTRIBUTING.md Update CONTRIBUTING.md 25 days ago

README.md

CB-Tumblebug (Multi-Cloud Infra Service Management)

go report A+ build passing go 66.2% go.mod v1.16 repo size 28.7 MB API Doc Swagger

release v0.3.0-espresso (release(dev) v0.3.7 license Apache-2.0 Slack SIG-TB

all contributors 12

A sub-system of Cloud-Barista Platform to Deploy and Manage Multi-Cloud Infrastructure.

Contributors (10)

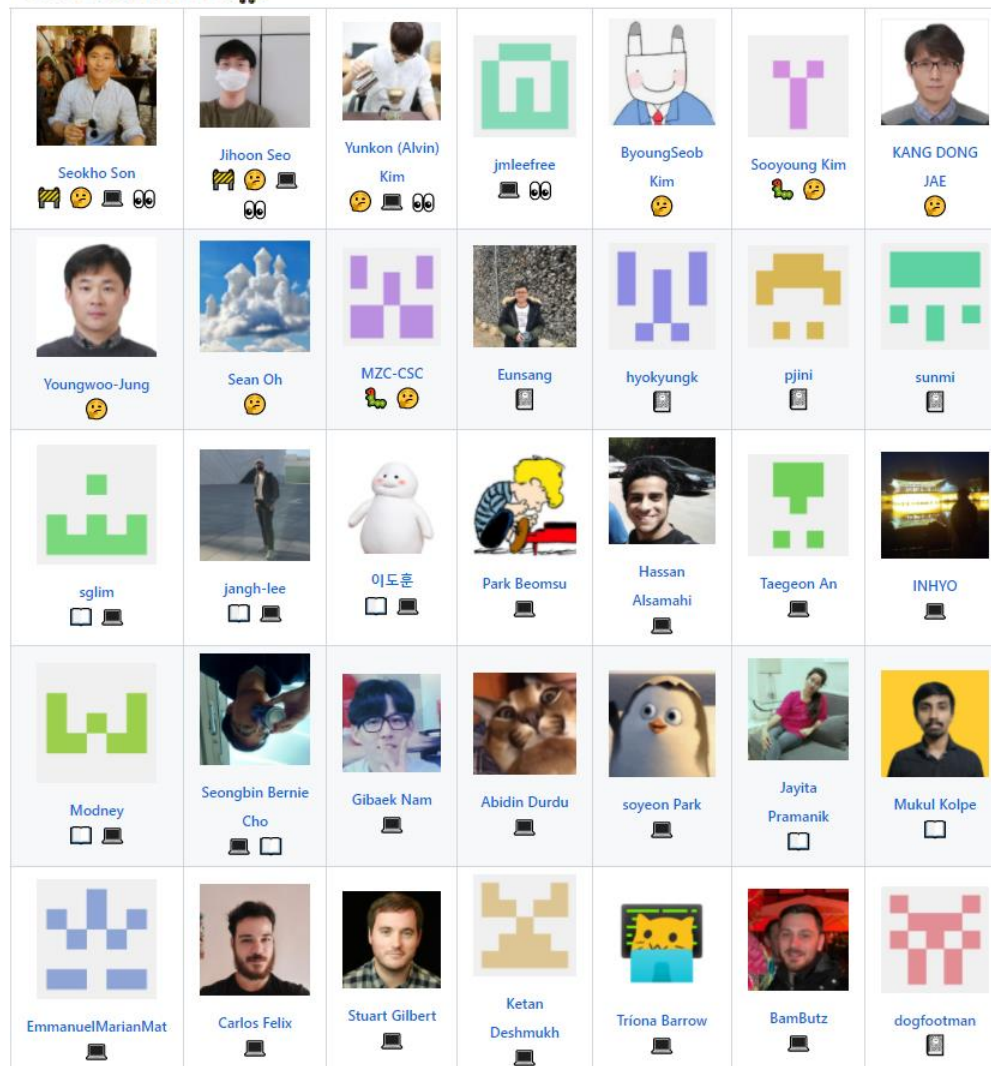
Languages
















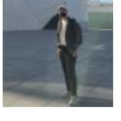






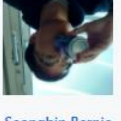
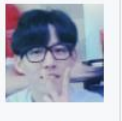
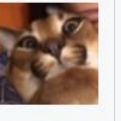
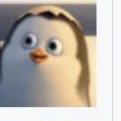

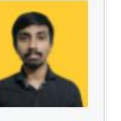

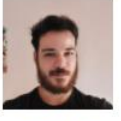



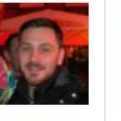

Go 66.2% Shell 33.6% Other 0.2%

CB-Tumblebug GitHub

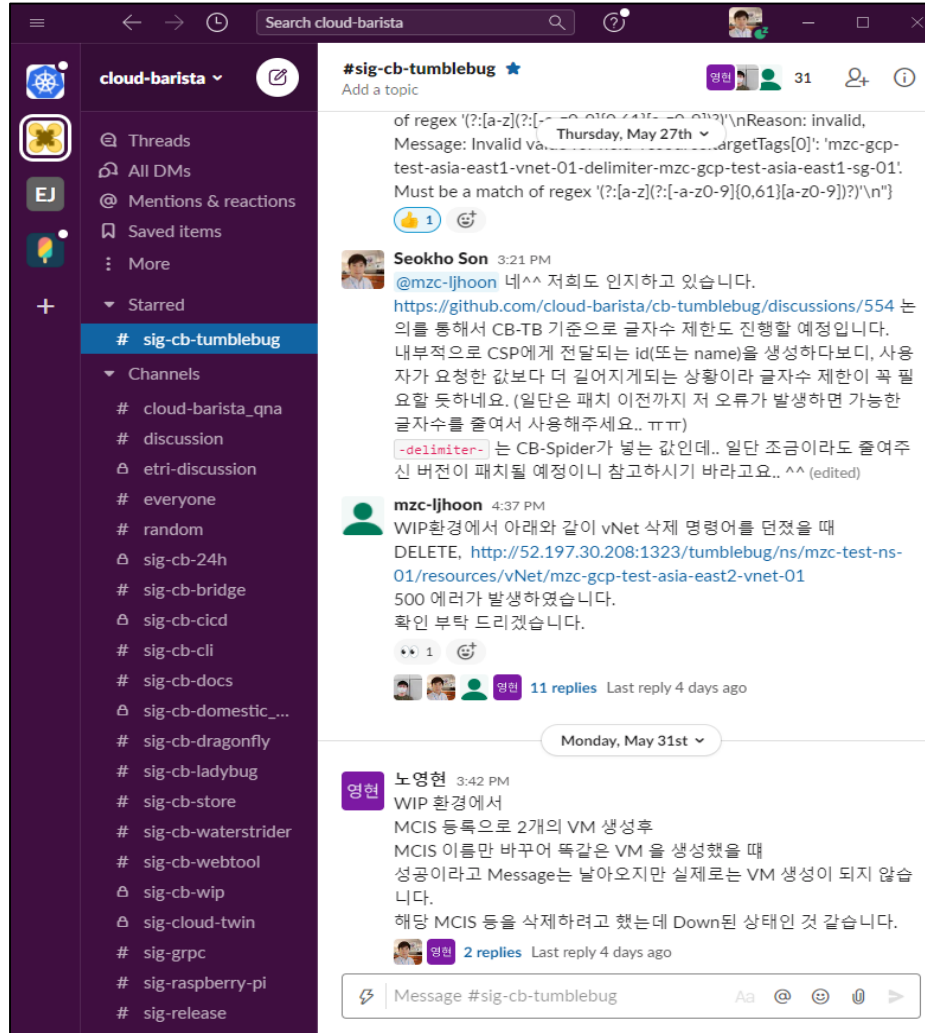
<https://github.com/cloud-barista/cb-tumblebug>

Contributors



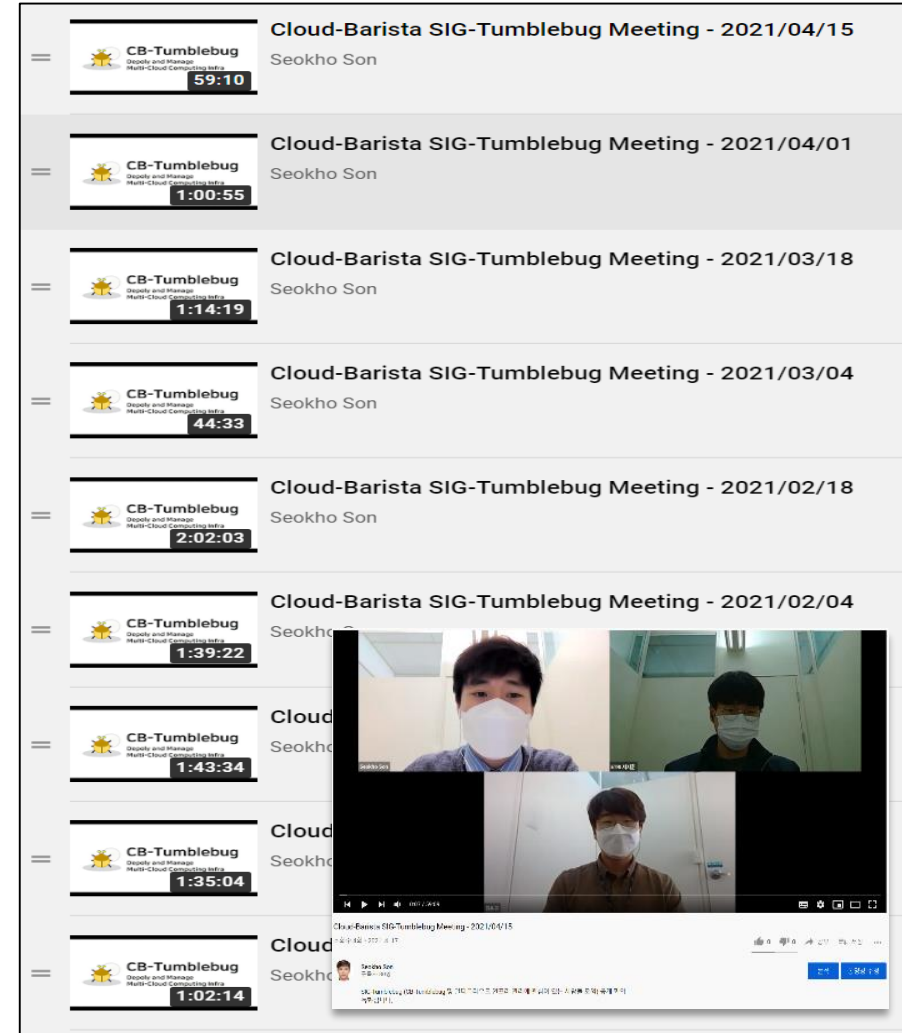
 Seokho Son	 Jihoon Seo	 Yunkon (Alvin) Kim	 jmleefree	 ByoungSeob Kim	 Sooyoung Kim	 KANG DONG JAE
 Youngwoo-Jung	 Sean Oh	 MZC-CSC	 Eunsang	 hyokyungk	 pjini	 sunmi
 sglim	 jangh-lee	 이도훈	 Park Beomsu	 Hassan Alsamahi	 Taegeon An	 INHYO
 Modney	 Seongbin Bernie Cho	 Gibaek Nam	 Abidin Durdu	 soyeon Park	 Jayita Pramanik	 Mukul Kolpe
 EmmanuelMarianMat	 Carlos Felix	 Stuart Gilbert	 Ketan Deshmukh	 Triona Barrow	 BamButz	 dogfootman

[참고] CB-Tumblebug 참여 방법 (환영합니다..^^)



CB-Tumblebug Slack Channel

<https://cloud-barista.slack.com/archives/CJQ7575PU>



CB-Tumblebug 정기 공개 회의 (1회/3주)

<https://www.youtube.com/playlist?list=PL9Bd5opncjJAHYr9oPn8i9kClBacgBw8>

감사합니다.

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

(손 석 호 / contact-to-cloud-barista@googlegroups.com)

멀티클라우드, “새로운 생태계를 향한 클라우드 비긴어게인”

클라우드 바리스타들의 다섯번째 이야기

Cloud-Barista Community the 5th Conference