



# 멀티클라우드, “새로운 생태계를 향한 클라우드 비긴어게인”

## 클라우드바리스타 커뮤니티 제5차 컨퍼런스

[세션] CB-Ladybug:

## 멀티클라우드 애플리케이션 통합 관리

김수영

CB-Ladybug 프레임워크 리더

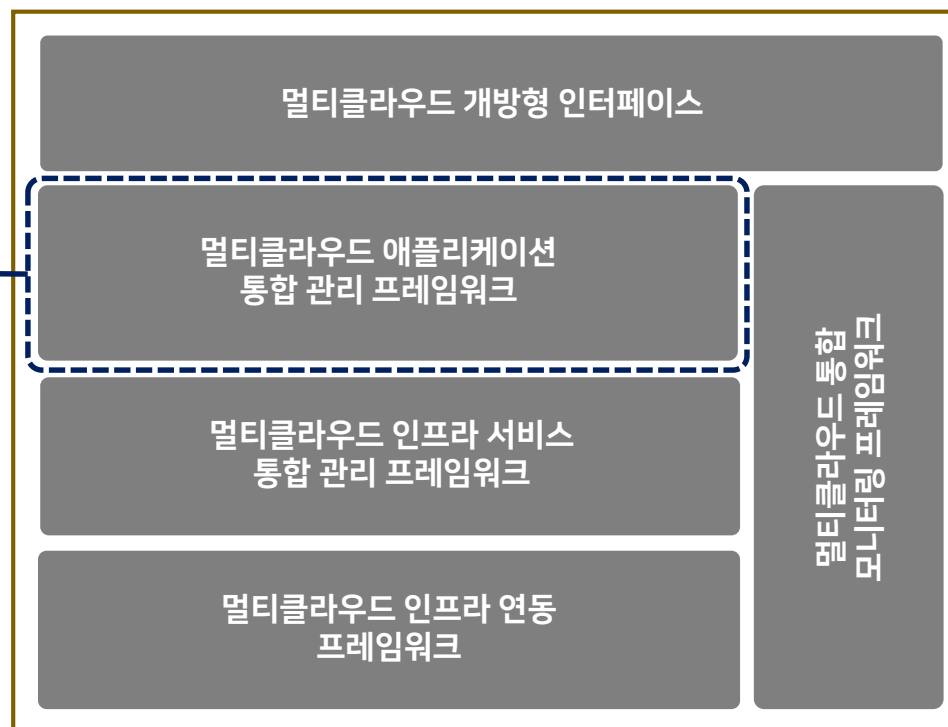
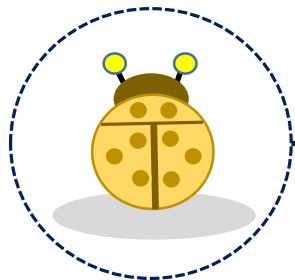
아포가토(Affogato) 한잔 어떠세요?

# 이번 세션은...

응용/도메인/기관 특화 SW



CB-Ladybug



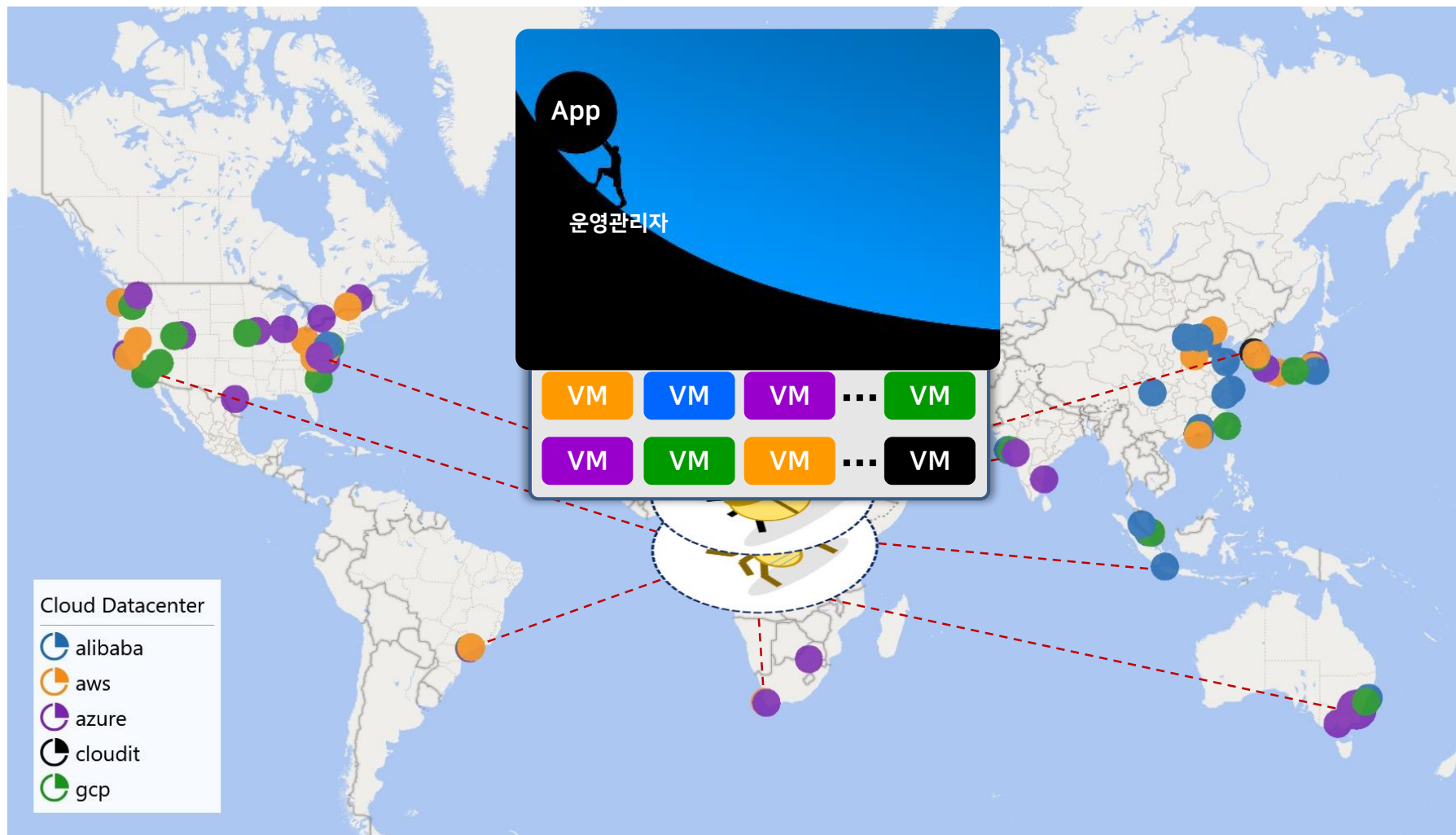
멀티클라우드 서비스 공통 플랫폼

# 목 차

---

- I** CB-Ladybug 개요 및 주요 기능
- II** CB-Ladybug 개발 전략
- III** 멀티클라우드 애플리케이션 실행환경 및 쿠버네티스 서비스
- IV** 멀티클라우드 애플리케이션 서비스
- V** CB-Ladybug 개발 로드맵

# 멀티클라우드 애플리케이션의 배포와 관리는?



?????

멀티클라우드 자원  
통합 컴퓨팅 인프라

멀티클라우드 인프라 서비스 (MCIS)

컴퓨팅  
자원

컴퓨팅  
자원

컴퓨팅  
자원

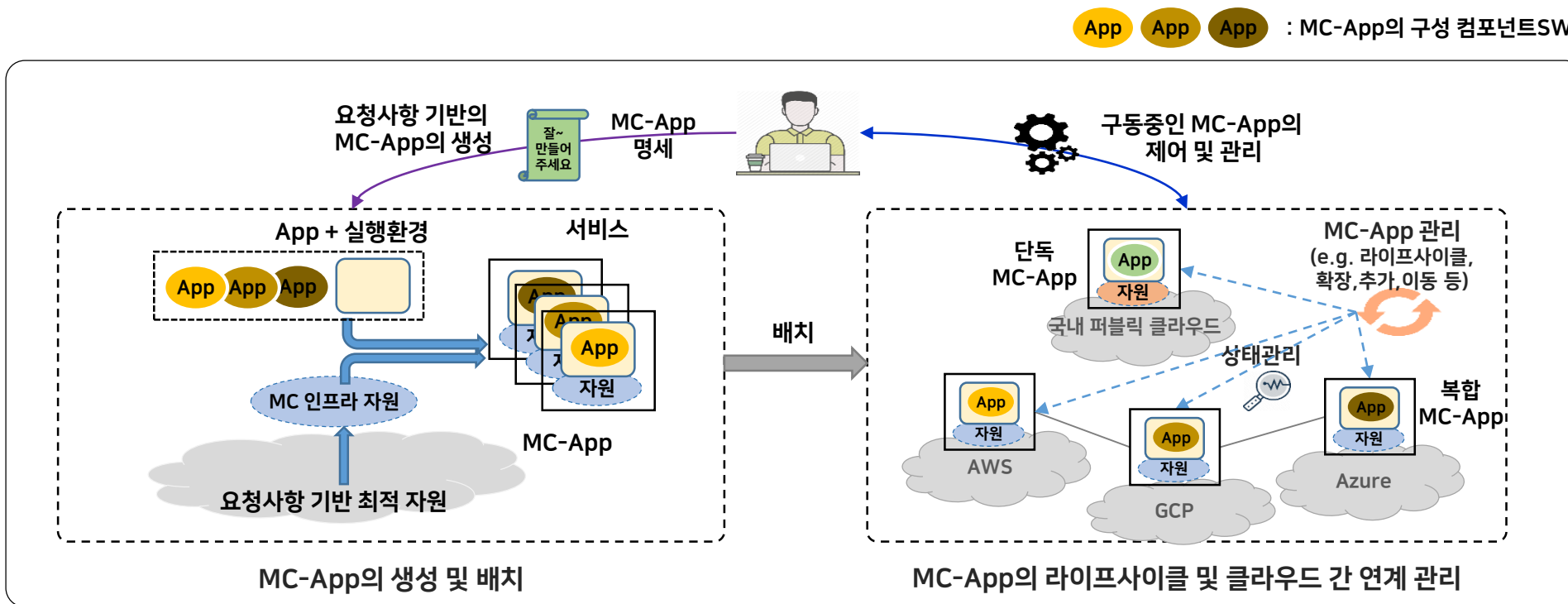
이중 멀티클라우드 연동



멀티클라우드 인프라를 기반으로, 글로벌 서비스를 위한 애플리케이션을 보다 쉽고 빠르게 배포하고 운영/관리할 방법이 필요

# CB-Ladybug 개요

- 멀티클라우드 애플리케이션 통합 관리 프레임워크(CB-Ladybug)
  - 멀티클라우드 인프라 상에서 운용되는, 멀티클라우드 애플리케이션(MC-App)(\*)의 생성, 배포, 실행 및 라이프사이클 제어와 클라우드 간 연계 관리 기능을 제공하는 프레임워크
  - 주요 활용 분야: 사용자/데이터 근접 처리 응용, 가용성 극대화 응용, 초광역 커버리지 필요 응용, 분산 병렬 처리 응용 등

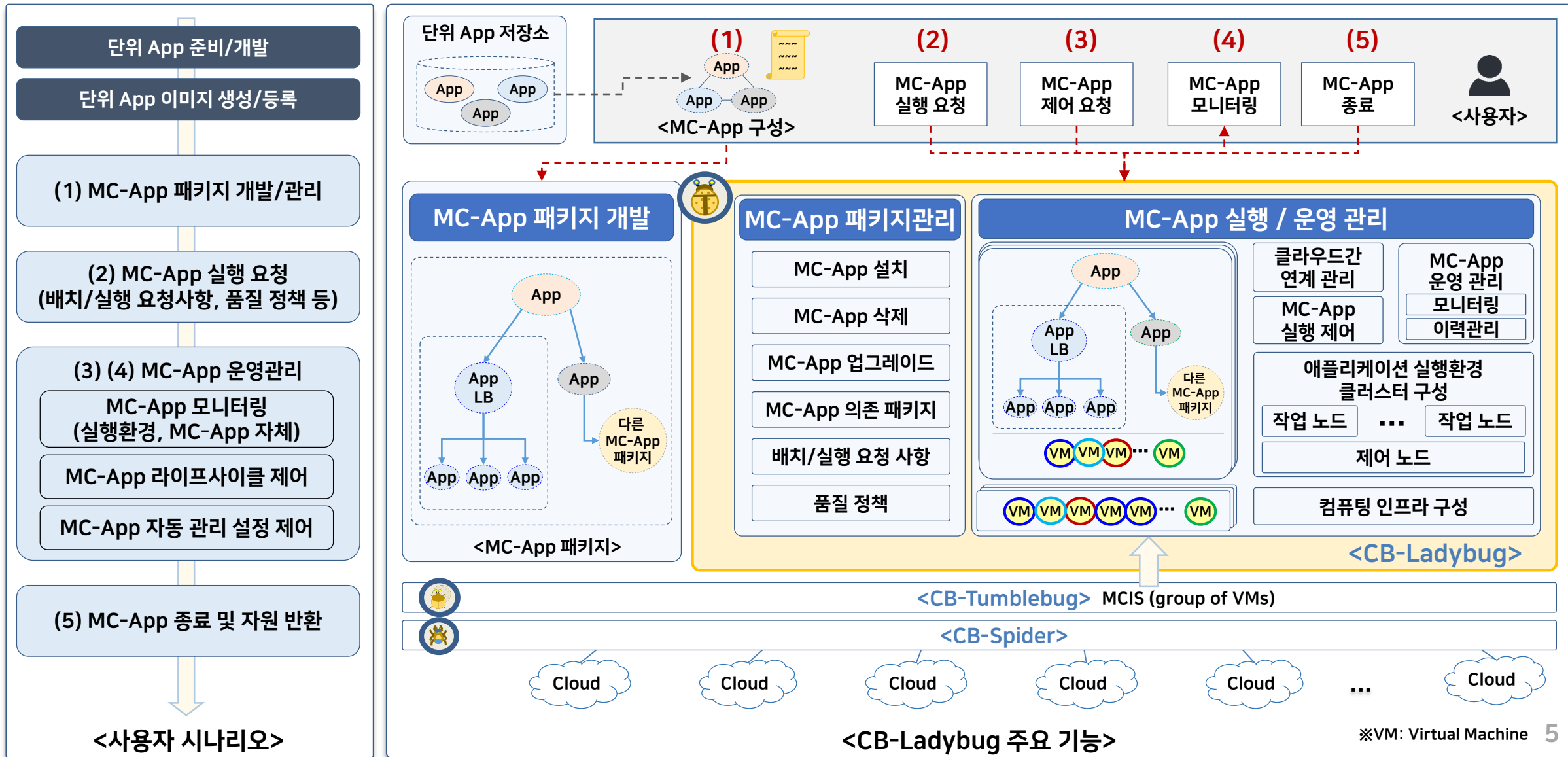


## <CB-Ladybug 개념도>

(\*) 멀티클라우드 애플리케이션(MC-App): 컨테이너 기술과 마이크로서비스 아키텍처를 기반으로 데브옵스와 지속적 배포를 통해 서비스 중지 없이 쉽고 빠르게 배포하고, 피드백을 즉각 수정/반영할 수 있는 구조를 제공하는 멀티클라우드 환경에 적합한 클라우드 네이티브 애플리케이션

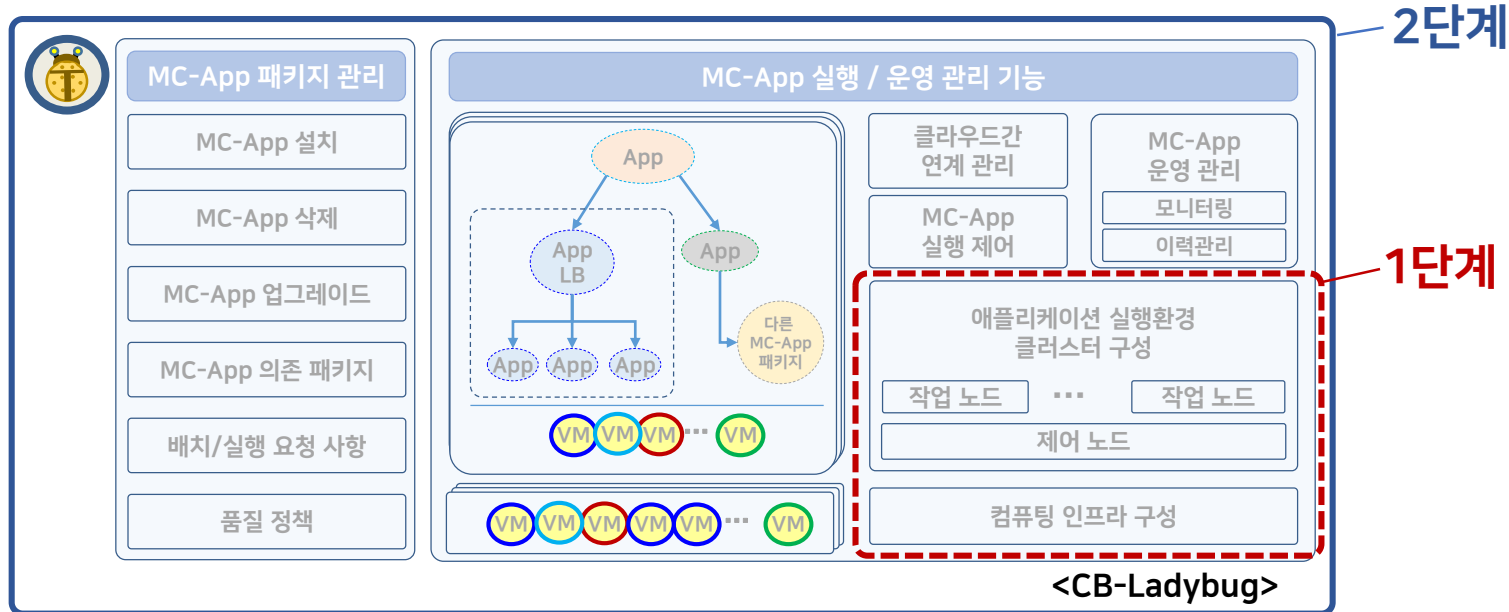


# CB-Ladybug 주요 기능



# CB-Ladybug 개발 전략

- 1단계: 멀티클라우드 환경에 적합한 애플리케이션 실행환경 제공 기술 개발
  - 애플리케이션 실행환경(쿠버네티스, 아파치 메소스, 도커 스웍 등) 중 **쿠버네티스를 대상으로** 멀티클라우드 인프라 상에서 운용 가능성/적합성 확인 및 검증
- 2단계: 멀티클라우드 환경에 최적화된 애플리케이션 통합 관리 기술 개발
  - 멀티클라우드 환경을 고려한 애플리케이션 실행 요청 사항의 도출
  - 클라우드를 인지하는 **멀티 클러스터 간/클러스터 내 MC-App 최적 배치 알고리즘 개발**

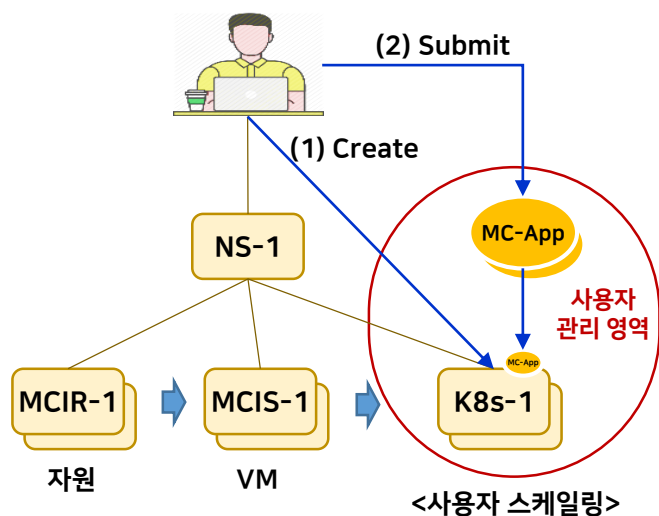


# CB-Ladybug 단계별 결과물

- 1단계: 멀티클라우드 애플리케이션 실행환경 및 멀티클라우드 쿠버네티스 서비스 제공
- 2단계: 멀티클라우드에 최적화된 애플리케이션 통합 관리 서비스 제공

## 멀티클라우드 애플리케이션 실행환경 (MCAR) 멀티클라우드 쿠버네티스 서비스 (MCKS)

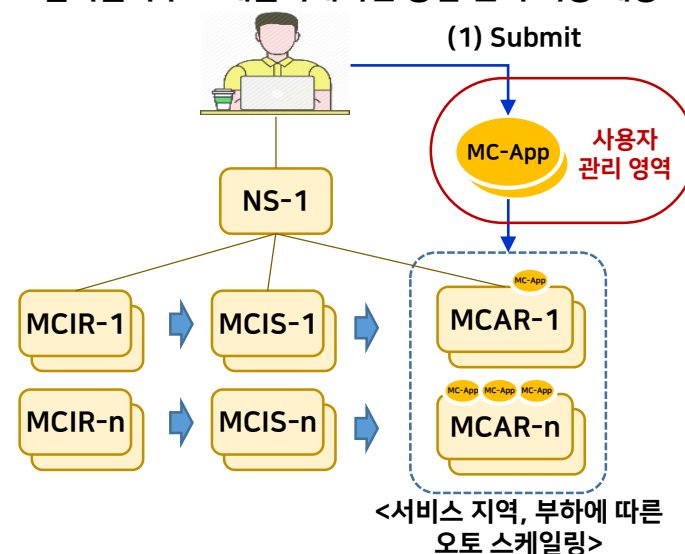
- 멀티클라우드 애플리케이션 실행환경(쿠버네티스) 제공
- 멀티클라우드 대상 사용자 관리형 쿠버네티스 클러스터 제공



• 기술/노하우

## 멀티클라우드 애플리케이션 서비스 (MCAS)

- 멀티클라우드 애플리케이션에 적합한 최적 실행환경 (MCAR) 동적 제공
- 멀티클라우드 애플리케이션 통합 관리 기능 제공



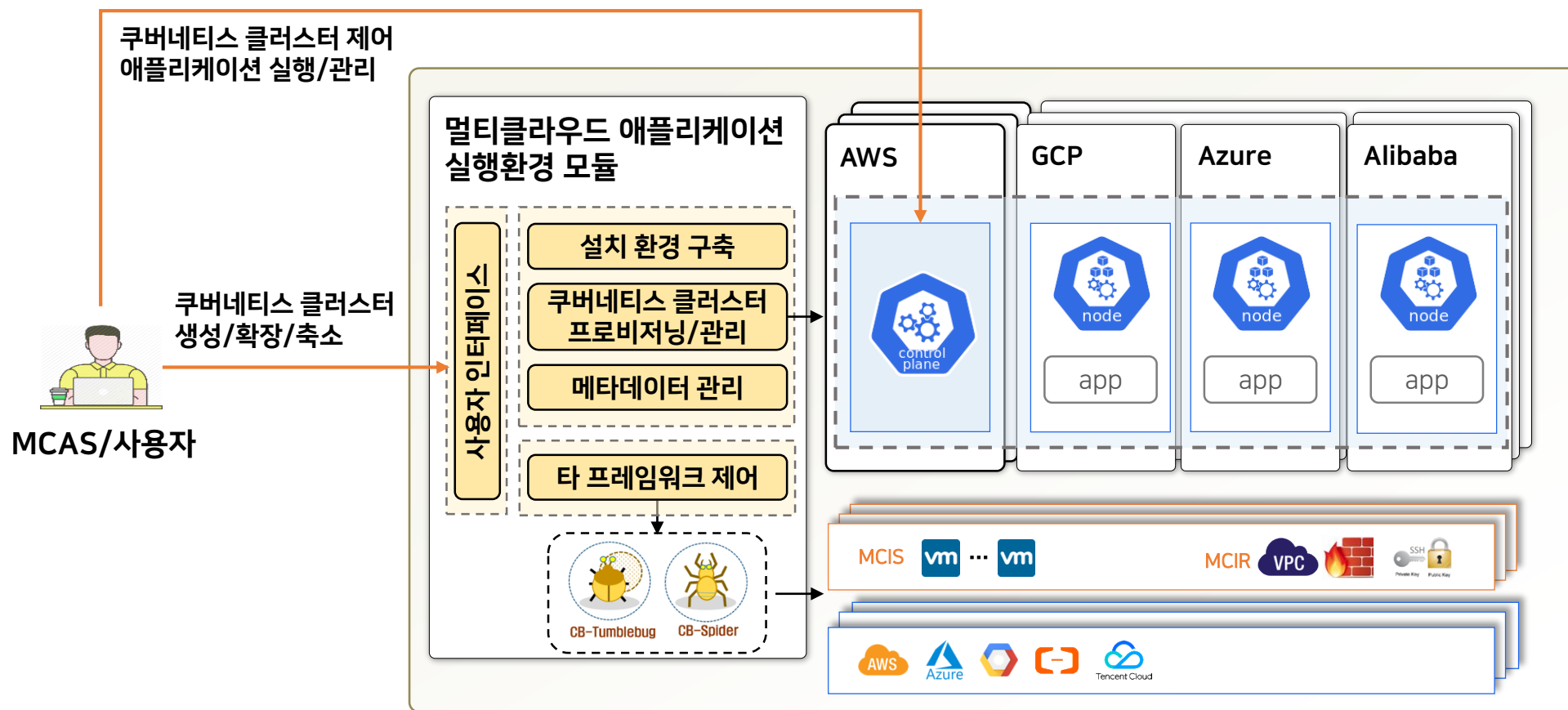
※ NS: Namespace  
※ MCIR: Multi-Cloud Infra Resource  
※ MCIS: Multi-Cloud Infra Service

※ MCKS: Multi-Cloud Kubernetes Service  
※ MCAS: Multi-Cloud Application Service  
※ MCAR: Multi-Cloud Application Runtime



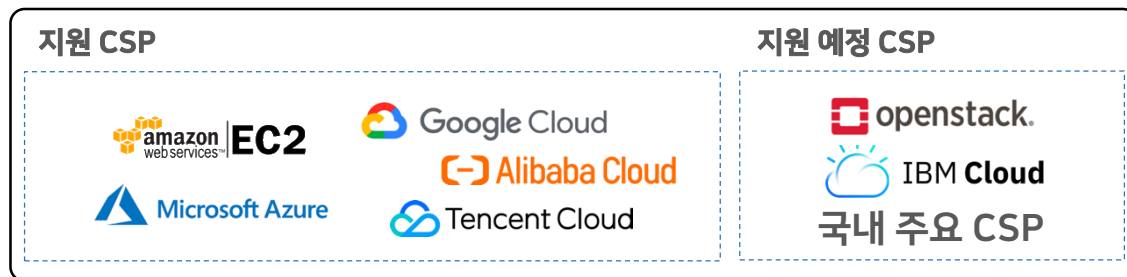
# 멀티클라우드 애플리케이션 실행환경(MCAR)/쿠버네티스 서비스(MCKS) 개요

- 멀티클라우드 애플리케이션 서비스의 기반 멀티클라우드 애플리케이션 실행환경(MCAR) 제공
- 멀티클라우드 환경에서 실행되는 사용자 관리형 쿠버네티스 클러스터 제공 서비스(MCKS)



# MCAR/MCKS 기술 현황

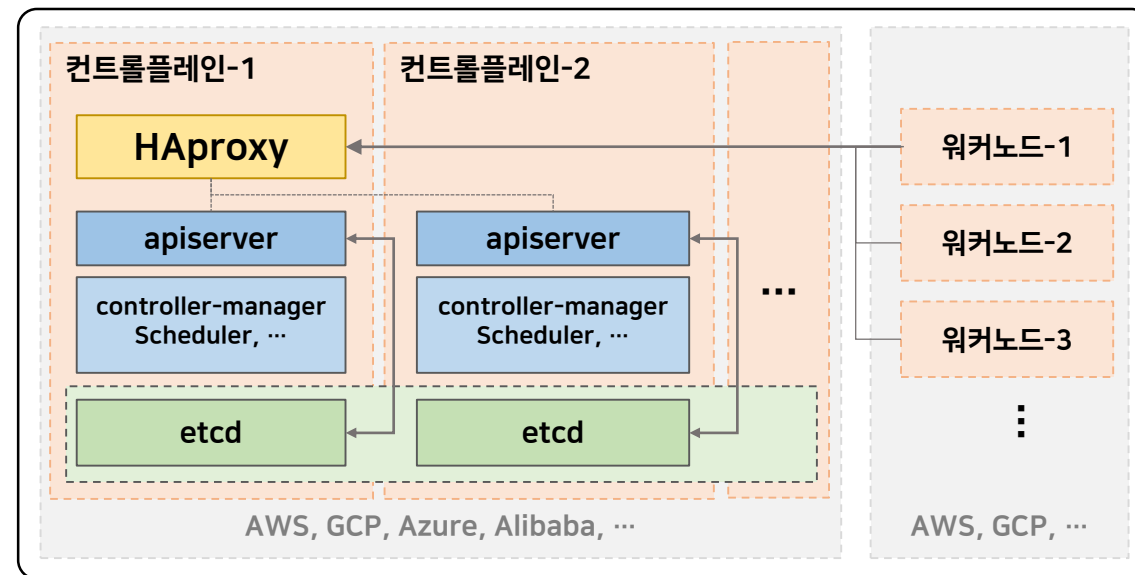
- 멀티클라우드 대상 쿠버네티스 클러스터 프로비저닝 지원



- 멀티클라우드 운영 가능 쿠버네티스 네트워크 플러그인 2종 (Canal, Kilo) 적용 가능

- Canal (Flannel+Calico): 다양한 네트워크 정책 적용
- Kilo: VPN(Wireguard) 활용 서로 다른 클라우드의 노드 간 통신 보안 제공

- 쿠버네티스 컨트롤플레인의 HA 구성을 통한 가용성 제공

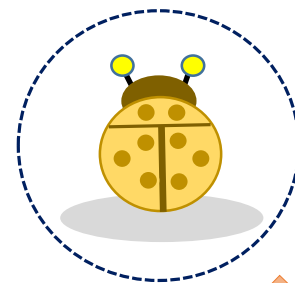


# 멀티클라우드 애플리케이션 서비스(MCAS) 개요

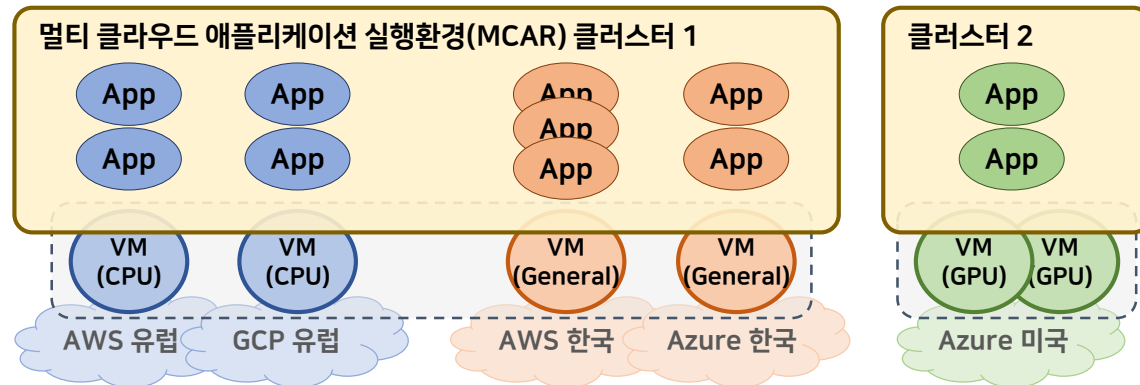
my-app-instance.yaml

- MC-App 인스턴스 이름
- MC-App 실행 이미지 정보 (위치, 버전, 접속 포트 등)
  - 단일 컨테이너 타입
  - 실행환경 명세 타입 (ex. K8s resource yaml)
  - 실행환경 패키지 타입 (ex. helm chart) PoC
- 실행 요청 사항
  - 서비스 요청 지역
  - 시스템 타입
    - General/CPU/Memory/GPU/...
  - 스케줄링/품질 정책
    - 서비스 지역 우선/리소스 균등 우선/비용 우선 등

MCAS

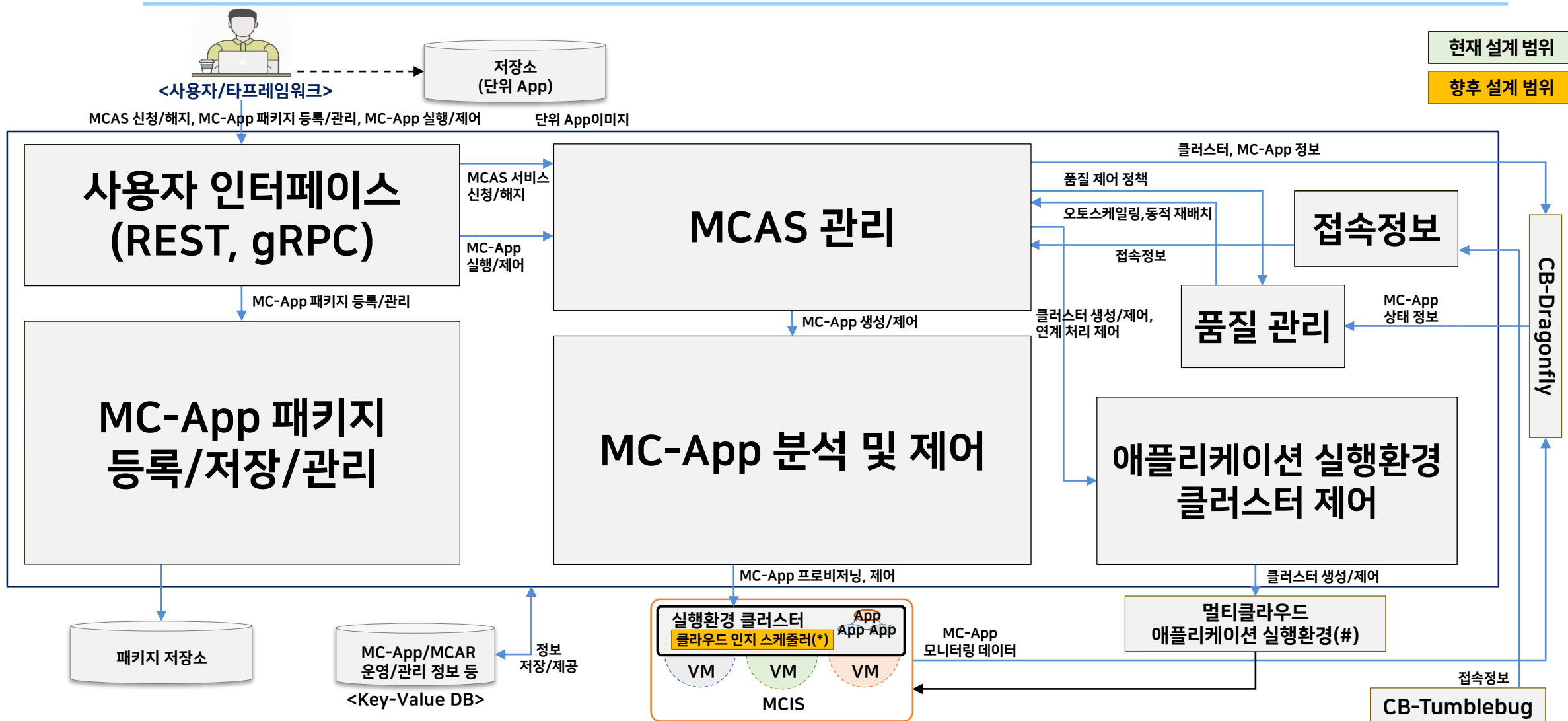


- MC-App 인스턴스의 클러스터 단위, 노드 단위의 최적 배치
- 서비스 요청 지역/부하에 따른 클러스터/노드 단위 스케일링



# MCAS 주요 구조

VM: Virtual Machine  
MC-App: Multi-Cloud Application  
MCAR: Multi-Cloud Application Runtime  
MCKS: Multi-Cloud Kubernetes Service  
MCAS: Multi-Cloud Application Service



(\*) 클라우드 인지 스케줄러: 클러스터에 포함된 노드들의 위치나 역량, 자원 현황, 서비스 클라이언트들의 상태 등에 따라 MC-App의 단위 App들을 적절한 위치에 배치 수행  
(#) 멀티클라우드 애플리케이션 실행환경 모듈: 애플리케이션 서비스(MCAS)와 쿠버네티스 서비스(MCKS)에서 효과적 활용을 위해 별도 모듈로 분리

# MCAS 개념 검증 (PoC)

```
[CLOUD-BARISTA].[DEBUG]: 2021-11-18 10:41:11 app_instance.go:145, github.com/cloud-barista/cb-ladybug/pkg/core/model.printAppInstanceReq() - AppInstanceReq.Version:
[CLOUD-BARISTA].[DEBUG]: 2021-11-18 10:41:11 app_instance.go:146, github.com/cloud-barista/cb-ladybug/pkg/core/model.printAppInstanceReq() - AppInstanceReq.Wait: true
[CLOUD-BARISTA].[DEBUG]: 2021-11-18 10:41:11 app_instance.go:147, github.com/cloud-barista/cb-ladybug/pkg/core/model.printAppInstanceReq() - AppInstanceReq.Timeout: 5m0s
[CLOUD-BARISTA].[DEBUG]: 2021-11-18 10:41:11 app_instance.go:148, github.com/cloud-barista/cb-ladybug/pkg/core/model.printAppInstanceReq() - AppInstanceReq.Force: false
[CLOUD-BARISTA].[DEBUG]: 2021-11-18 10:41:11 app_instance.go:149, github.com/cloud-barista/cb-ladybug/pkg/core/model.printAppInstanceReq() - AppInstanceReq.UpgradeCRDs: false
[CLOUD-BARISTA].[INFO]: 2021-11-18 10:41:11 nutsdb-driver.go:160, github.com/cloud-barista/cb-store/store-drivers/nutsdb-driver.(*NUTSDBDriver).GetList() - Key:/ns/lb-ns/clusters
[CLOUD-BARISTA].[DEBUG]: 2021-11-18 10:41:11 app_instance.go:207, github.com/cloud-barista/cb-ladybug/pkg/core/service.installAppInstance() - ChartSpec.ReleaseName: nginx-01, ChartSpec.ChartName: lb-ns/nginx
[CLOUD-BARISTA].[INFO]: 2021-11-18 10:41:11 app_instance.go:209, github.com/cloud-barista/cb-ladybug/pkg/core/service.installAppInstance() - try to install the application 'nginx-01'
2021-11-18 10:41:11.675960 I | getting history for release nginx-01
2021-11-18 10:41:13.774690 I | creating 1 resource(s)
2021-11-18 10:41:13.823840 I | creating 3 resource(s)
2021-11-18 10:41:13.895020 I | beginning wait for 3 resources with timeout of 5m0s
2021-11-18 10:41:13.967743 I | Deployment is not ready: lb-ns/nginx-01. 0 out of 4 expected pods are ready
2021-11-18 10:41:15.977712 I | Deployment is not ready: lb-ns/nginx-01. 0 out of 4 expected pods are ready
2021-11-18 10:41:17.978639 I | Deployment is not ready: lb-ns/nginx-01. 0 out of 4 expected pods are ready
2021-11-18 10:41:19.977455 I | Deployment is not ready: lb-ns/nginx-01. 0 out of 4 expected pods are ready
2021-11-18 10:41:21.983086 I | Deployment is not ready: lb-ns/nginx-01. 0 out of 4 expected pods are ready
2021-11-18 10:41:23.980664 I | Deployment is not ready: lb-ns/nginx-01. 0 out of 4 expected pods are ready
2021-11-18 10:41:25.978247 I | Deployment is not ready: lb-ns/nginx-01. 0 out of 4 expected pods are ready
2021-11-18 10:41:27.977657 I | Deployment is not ready: lb-ns/nginx-01. 0 out of 4 expected pods are ready
2021-11-18 10:41:30.019801 I | release installed successfully: nginx-01/nginx-9.5.12
{"time": "2021-11-18T10:41:30.019923791Z", "id": "", "remote_ip": "127.0.0.1", "host": "localhost:1592", "method": "POST", "uri": "/ladybug/ns/lb-ns/apps", "user_agent": "curl/7.58.0", "status": 200, "error": "", "latency": 18367843797, "latency_human": "18.367843797s", "bytes_in": 33751, "bytes_out": 5}
```

로그 출력 화면

```
+ scripts git:(add-dependabot-yml) x ./enable-mcas.sh lb-ns
[INFO]
- Namespace is 'lb-ns'
-----
"enabled"
+ scripts git:(add-dependabot-yml) x ./install-app-instance.sh lb-ns nginx-01 nginx nginx-values.yaml
[INFO]
- Namespace is 'lb-ns'
- App instance name is 'nginx-01'
- App package name is 'nginx'
- values.yaml file name is 'nginx-values.yaml'
-----
null
+ scripts git:(add-dependabot-yml) x
```

MC-App 실행 요청 화면

- 헬름(Helm): 쿠버네티스 전용 패키지 관리자
- 차트뮤지엄(Chartmuseum): 헬름 패키지 저장소 서버

<쿠버네티스 클러스터>

※ MCAS: Multi-Cloud Application Service  
※ MCAR: Multi-Cloud Application Runtime

제공  
가능

상 관리

# CB-Ladybug 개발 로드맵

## ~ 에스프레소 '2020.11.

- 공인IP 기반 쿠버네티스 프로비저닝, 데이터 통신 / 성능, 가용 CNI 검증
- MCKS 규격 정의
- 단일 클라우드 대상 MCKS 프로토타입 개발
- AWS, GCP 지원(총 2종)

## 카페모카 '2021.6.

- MCKS 멀티클라우드 모델 확장: **단일 쿠버네티스 on 3 CSPs**
- MCKS 컨트롤플레인 HA 구성 지원
- MCKS 멀티클라우드 네트워크 플러그인 지원
- MCAS 설계
- Azure 추가 지원(총 3종)

## 아포가토 '2021.11.

- MCKS 자원 롤백 등 기능 안정화
- 쿠버네티스 Addon 관리 지원
- **MCAS 싱글 클러스터 모델 기반 MC-App 프로비저닝/제어 (개념검증)**
- Alibaba, Tencent 추가 지원 (총 5종)

## 카페라떼 '2022.4.

- **로드밸런서 기반 컨트롤 플레인 HA** 지원
- 컨트롤플레인-노드간 통신 보안성 강화
- **MCAS 멀티클러스터 모델 확장**
- MC-App의 클러스터 수준 최적 배치(개념검증)
- MC-App **서비스 요청 지역 배치** 지원
- Openstack, IBM, 국내 CSP 지원

## 중장기 로드맵

- MC-App의 **클러스터 단위 동적 최적 배치** 지원
- MC-App의 **노드 단위 동적 최적 배치** 지원 (클라우드 인지 스케줄러)
- 부하 기반 **클러스터 단위 스케일링** 지원
- 부하 기반 **노드 단위 스케일링** 지원
- 국내외 CSP & 리전 확장



# 감사합니다.

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

(김 수 영 / [contact-to-cloud-barista@googlegroups.com](mailto:contact-to-cloud-barista@googlegroups.com))

## 멀티클라우드, “새로운 생태계를 향한 클라우드 비긴어게인”

클라우드 바리스타들의 다섯번째 이야기

Cloud-Barista Community the 5th Conference