

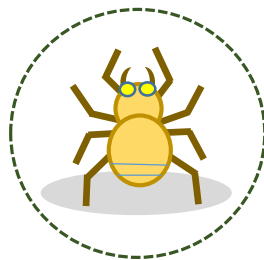


# Journey to the Multi-Cloud

클라우드바리스타 커뮤니티 제3차 컨퍼런스

## CB-Spider : 전세계 클라우드를 엮는 그날까지

(멀티 클라우드 인프라 연동)



김 병 섭 / CB-Spider 프레임워크 리더



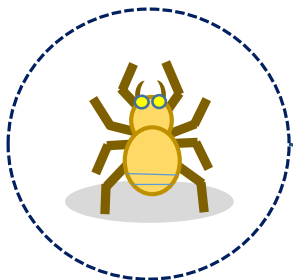
에스프레소(Espresso) 한잔 어떠세요 ?

# 이번 세션은...

응용/도메인/기관 특화 SW



CB-Spider



멀티클라우드 서비스 개방형 인터페이스

멀티 클라우드 애플리케이션  
통합관리 프레임워크

멀티 클라우드 인프라 서비스  
통합 관리 프레임워크

멀티 클라우드 인프라 서비스  
연동 프레임워크

멀티 클라우드 통합  
모니터링 프레임워크

멀티 클라우드 서비스 공통 플랫폼

# 목 차

---

**I**

CB-Spider 프레임워크 기술 개요

**II**

CB-Spider 프레임워크 주요 기술 및 개발 현황

**III**

CB-Spider 프레임워크 기술 개발 로드맵

**IV**

CB-Spider 프레임워크 기술 시연

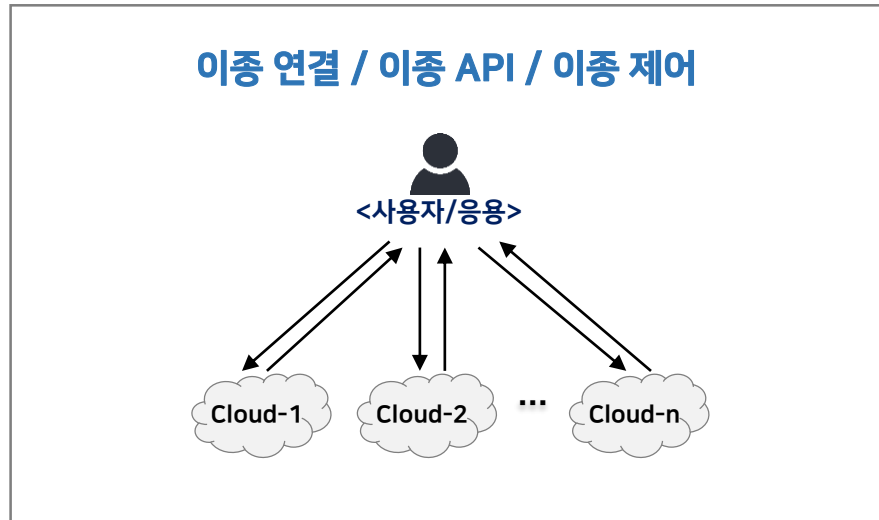
# 멀티 클라우드 인프라 연동 프레임워크 개요 및 필요성

- 컴퓨팅 인프라 변화 추이 및 현황:



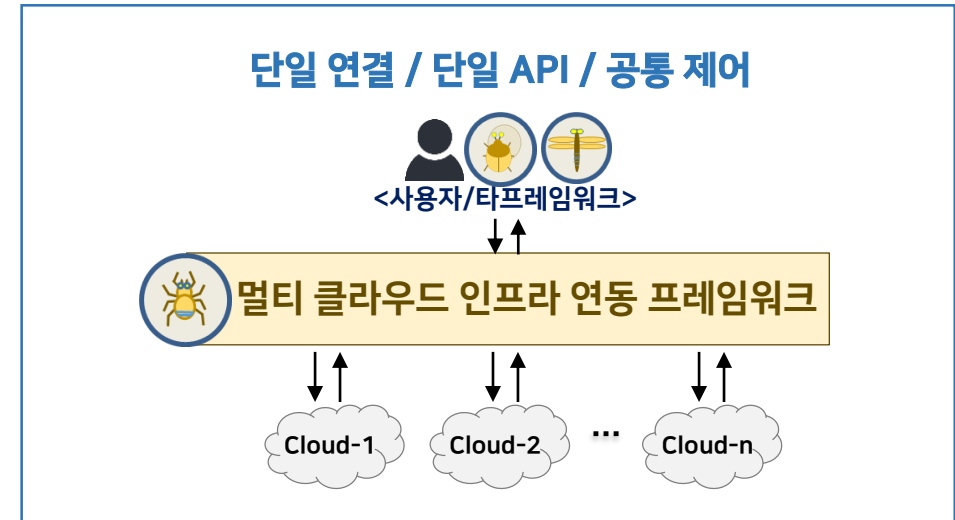
- CB-Spider**: 플러그인-드라이버 기반의 클라우드 연동 확장성을 제공하는 멀티 클라우드 인프라 연동 프레임워크

신규 클라우드 연동 필요 → 활용방법/전용 API 분석  
→ 응용 추가 개발 → 활용



<기존 멀티 클라우드 서비스 사용자 환경>

신규 클라우드 연동 필요 → 클라우드 드라이버 등록  
→ 활용



<멀티 클라우드 인프라 연동 프레임워크 사용자 환경>



# CB-Spider 프레임워크 최종 목표



- 전세계 이중 클라우드를 엮을 수 있게 한다.
- 전세계 클라우드를 하나처럼 쓸 수 있게 한다.

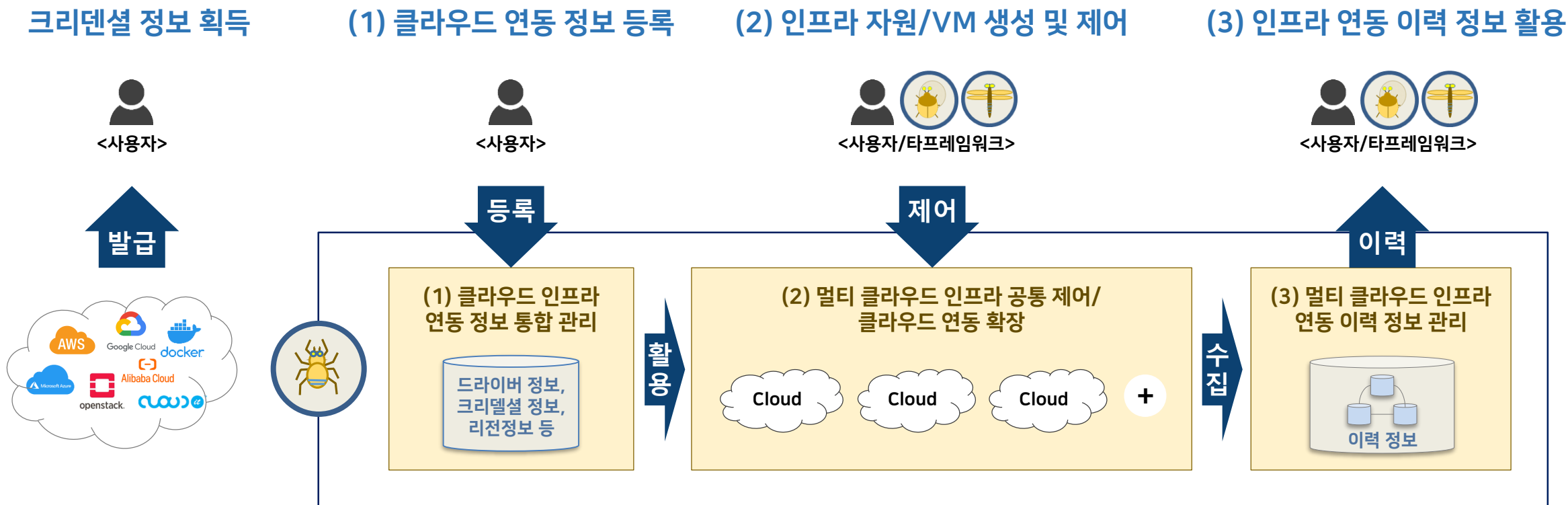
무한  
확장

공통  
제어

이중  
연동

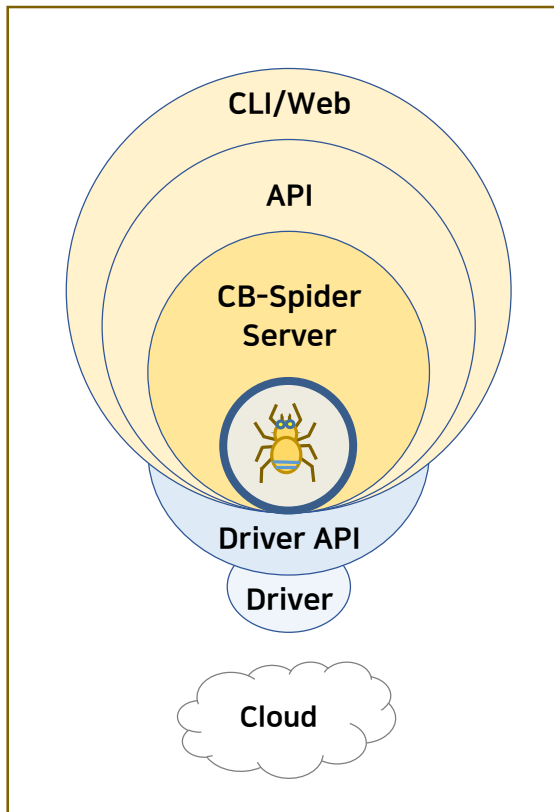
# 멀티 클라우드 인프라 사용자의 서비스 활용 시나리오

- 멀티 클라우드 인프라 자원/VM 활용 시나리오
- 시나리오 분석을 통한 {요구분석 → 기능도출 → 설계}



# 멀티 클라우드 연동 제공을 위한 프레임워크 구성 및 기능

- 프레임워크 구성: Interface – Server – Driver API – Drivers – {Clouds}
- 프레임워크 특징: 드라이버-플러그인 기반 클라우드 연동, 단일 API/통합 제어, CSP API 호출 이력 정보 제공



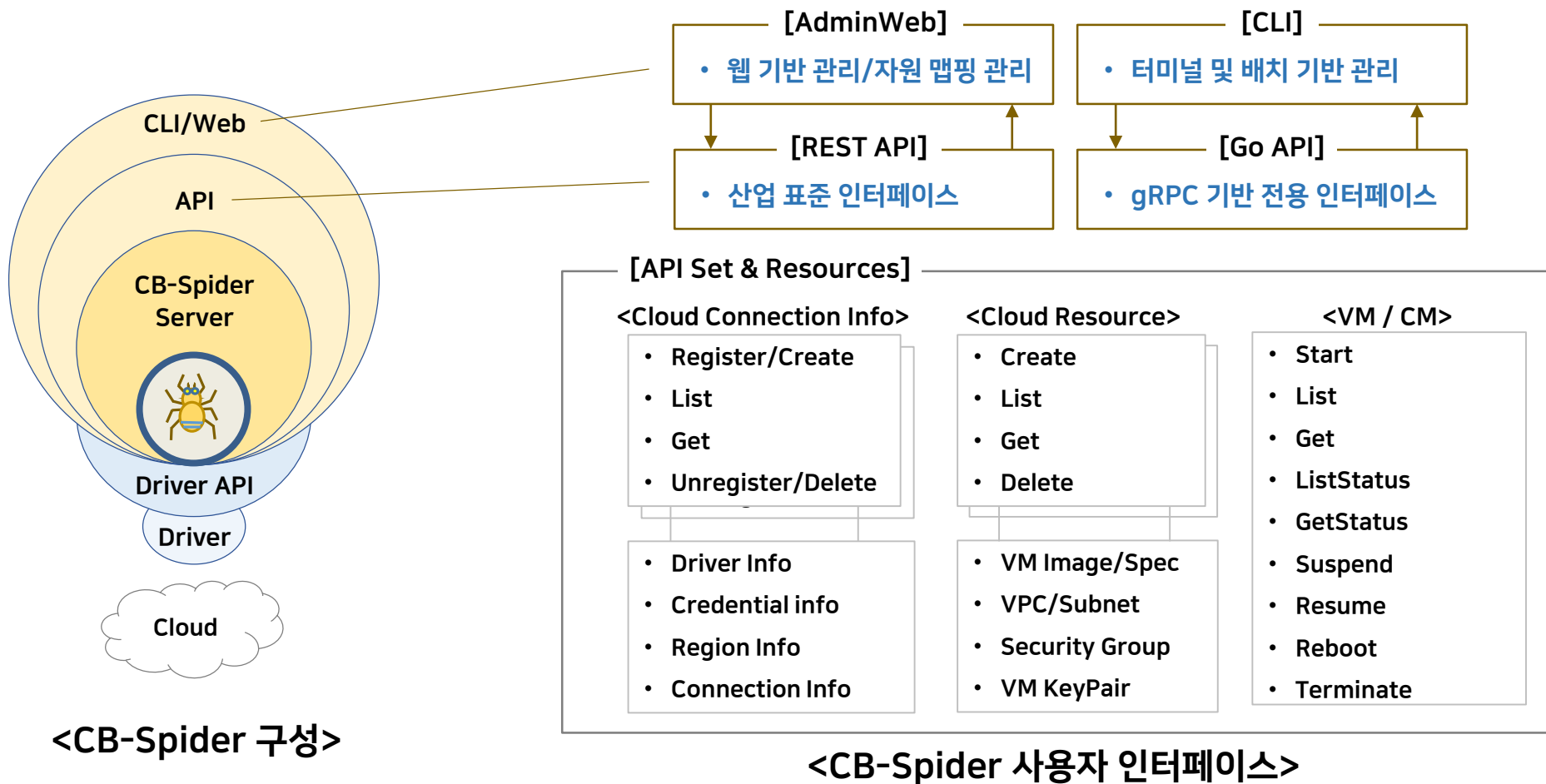
<CB-Spider 프레임워크 구성>

- 다양한 사용자 인터페이스 제공
  - CLI 및 AdminWeb 제공
  - REST 및 Go API 제공
- 멀티 클라우드 연동 정보 통합 관리 제공
  - 연동정보: 클라우드 드라이버 정보, 크리덴셜 정보, 리전/존 정보 등
- 멀티 클라우드 컴퓨팅 인프라 자원 및 VM 라이프사이클 등 공통 제어 제공
- 멀티 클라우드 연동 동적 확장 지원
- 클라우드 드라이버 공통 인터페이스 및 플러그인 방식 제공
  - 9종 드라이버 제공(AWS, GCP, Azure, Alibaba, Cloudfit, OpenStack, Docker, \* Emulation: Cloud-Twin, Mock)
- 대상: 이중 멀티 클라우드 컴퓨팅 인프라 자원
  - 7종 클라우드 연동: AWS, GCP, Azure, Alibaba, Cloudfit, OpenStack, Docker
- 멀티 클라우드 호출 이력 정보 수집 및 제공

<CB-Spider 프레임워크 주요 기능>

# 다양하고 심플한 사용자 인터페이스 제공

- 개발자를 위한 REST 및 Go API, 터미널 및 웹 사용자를 위한 CLI 및 AdminWeb 등 다양한 인터페이스 제공
- 대상별 Create-Get-List-Delete 등 심플한 인터페이스 제공

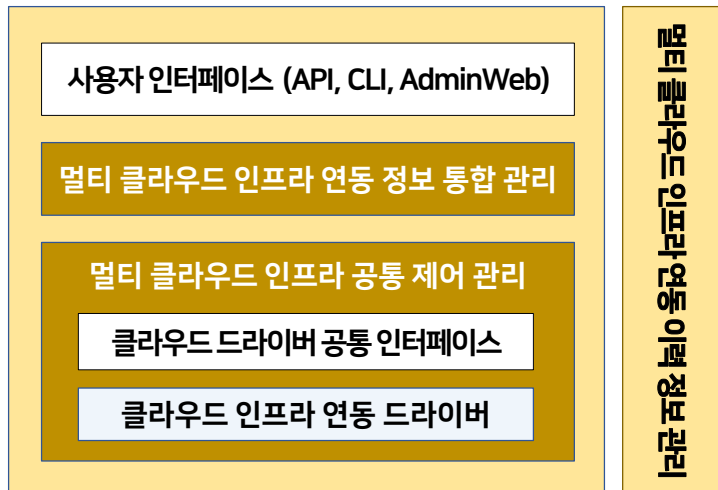




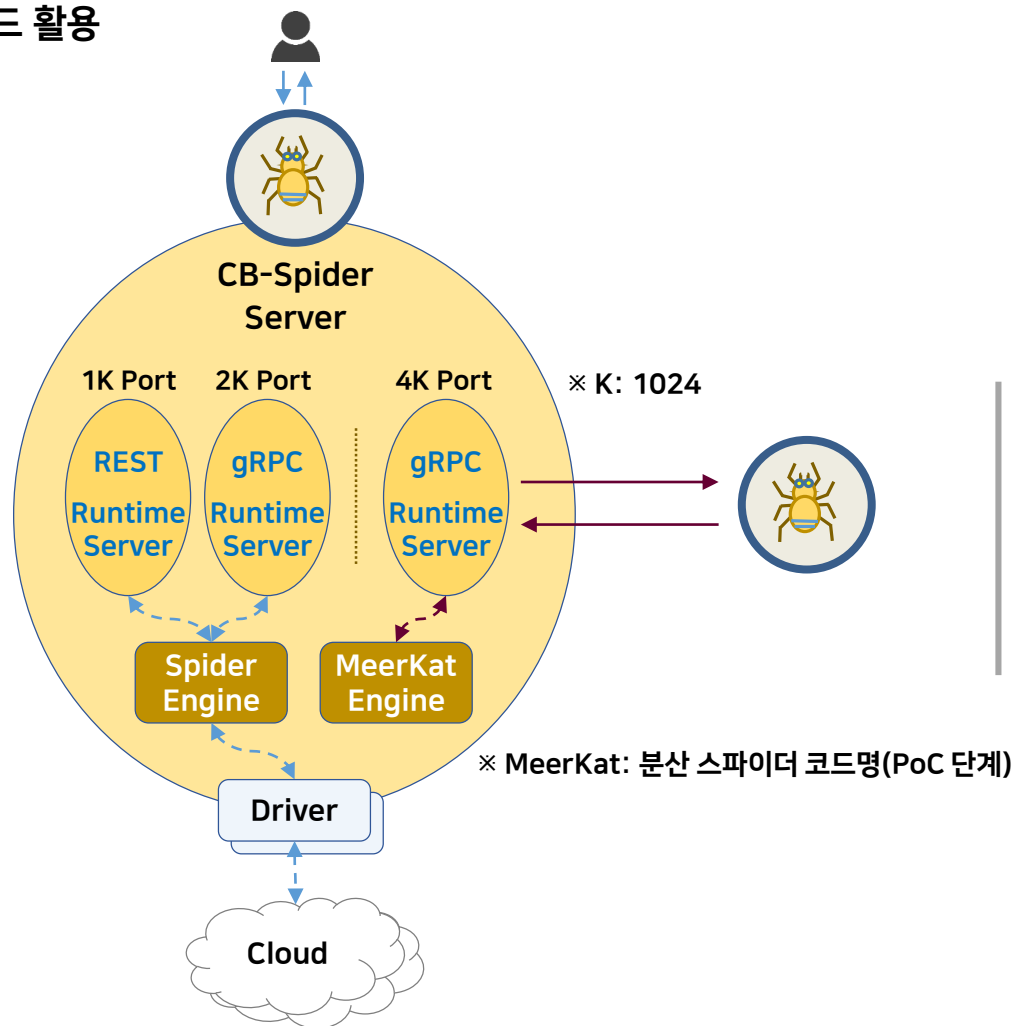
# 역할 기반의 서버 프로세스/쓰레드 구조

- 메인 프로세스 및 런타임 서버 쓰레드들로 구성

※ 드라이버: fetch 성능 개선을 위하여 필요시 병렬 쓰레드 활용



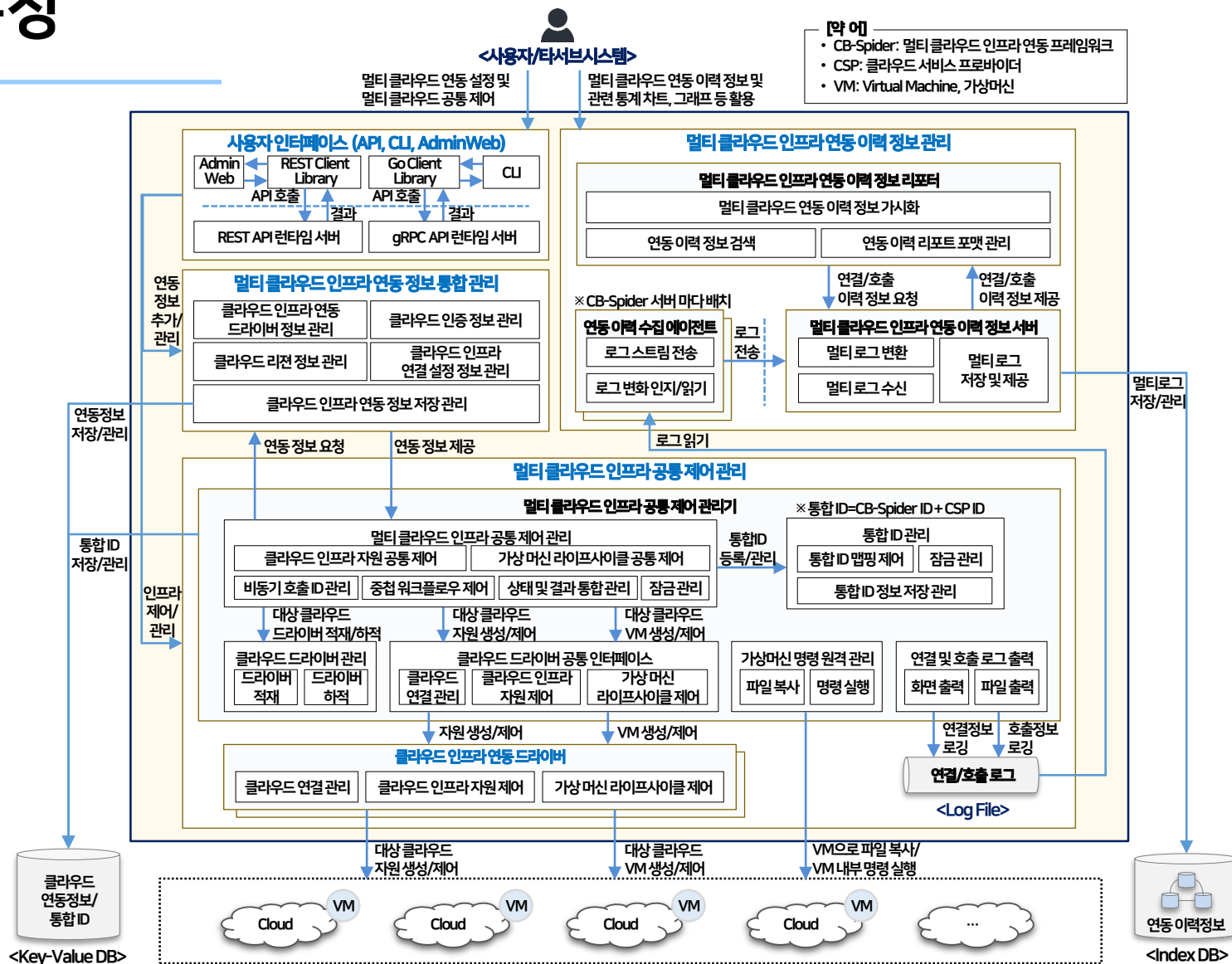
<CB-Spider 주요 컴포넌트>



[범례]

# 유연한 내부 구조 및 특징

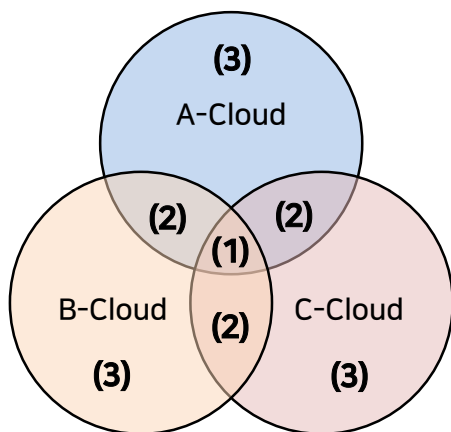
- 컴포넌트 기반의 계층화 구조
- 드라이버-플러그인 구조(9종)
- 메타정보-플러그인 구조(2종)
- 주요 컴포넌트
  - 사용자 인터페이스
  - 멀티 클라우드 인프라 연동 정보 통합 관리기
  - 멀티 클라우드 인프라 공통 제어 관리기
  - 클라우드별 인프라 연동 드라이버
  - 멀티 클라우드 인프라 연동 이력 정보 관리기



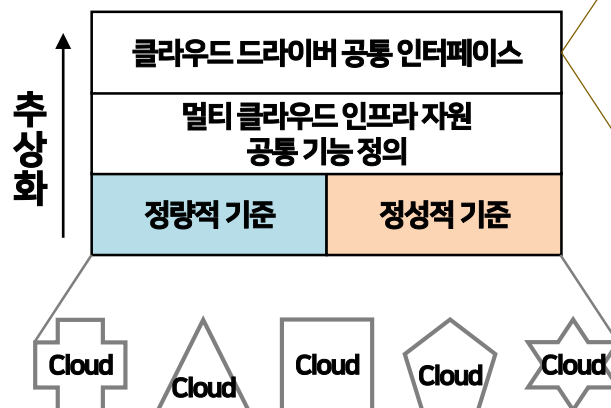
# 멀티 클라우드를 엮는 기술 - 추상화/공통인터페이스

- 드라이버 수준에서 CSP 기능 보정을 통한 확장된 기능을 제공하는 **기능 추상화** 제공 (예시: VM 키페어 제공)
- 정량적 기준 및 정성적 기준을 함께 수용한 **의미 추상화** 제공 (예시: VPC ← SG 의존 관계 추상화 등)
- 클라우드 연동 확산을 위한 드라이버 **공통 인터페이스** 규격 제공

- (1) 모든 Cloud 공통 기능 (Spider: O)  
(2) 일부 Cloud 공통 기능 (Spider: O)  
(3) 개별 클라우드 특화 기능 (Spider: X)



<멀티 클라우드 인프라 자원 기능 관계>



<멀티 클라우드 인프라 자원 추상화>

## [Cloud Driver Common API 예시]

```
type CloudConnection interface {
    CreateImageHandler() (irs.ImageHandler, error)
    CreateVPCHandler() (irs.VPCHandler, error)
    CreateSecurityHandler() (irs.SecurityHandler, error)
    CreateKeyPairHandler() (irs.KeyPairHandler, error)
    CreateVMHandler() (irs.VMHandler, error)
    CreateVMSpecHandler() (irs.VMSpecHandler, error)
    IsConnected() (bool, error)
    Close() error
}
```

```
type VPCHandler interface {
    CreateVPC(vpcReqInfo VPCReqInfo) (VPCInfo, error)
    ListVPC() ([]*VPCInfo, error)
    GetVPC(vpcIID IID) (VPCInfo, error)
    DeleteVPC(vpcIID IID) (bool, error)
}
```

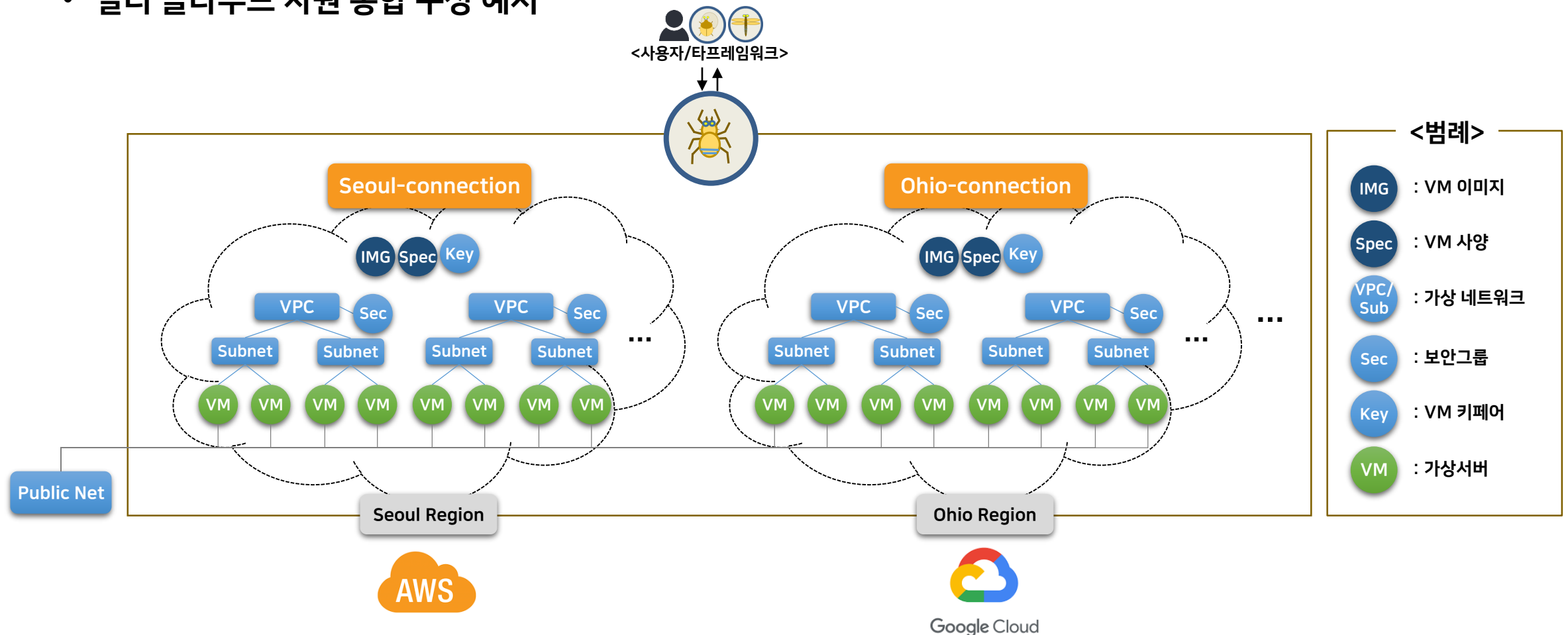
```
type VMHandler interface {
    StartVM(vmReqInfo VMReqInfo) (VMInfo, error)
    SuspendVM(vmlID IID) (VMStatus, error)
    ResumeVM(vmlID IID) (VMStatus, error)
    RebootVM(vmlID IID) (VMStatus, error)
    TerminateVM(vmlID IID) (VMStatus, error)
    ListVMStatus() ([]*VMStatusInfo, error)
    GetVMStatus(vmlID IID) (VMStatus, error)
    ListVM() ([]*VMInfo, error)
    GetVM(vmlID IID) (VMInfo, error)
}
```

...

<클라우드 드라이버 공통 인터페이스>

# 멀티 클라우드를 엮는 기술 - 멀티 클라우드 인프라 자원 구성

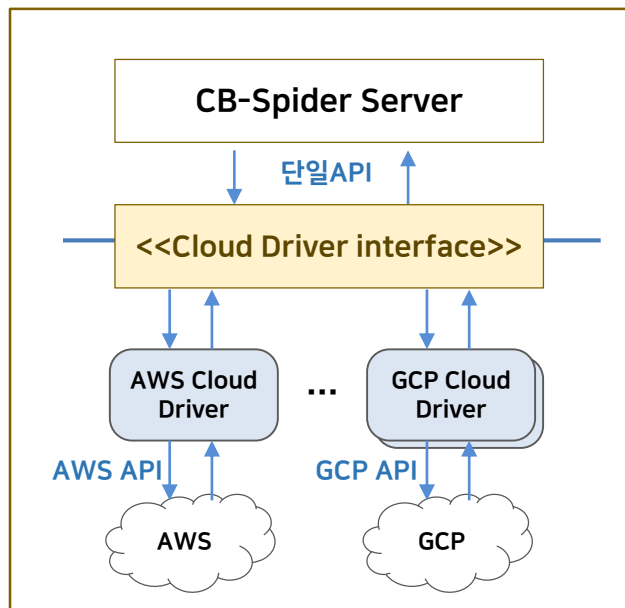
- 이종 멀티 클라우드 자원 통합 구성 제공 → 동일 CSP 처럼 공통 제어 가능
- 클라우드 '연결 설정' 중심의 자원 독립 구성 제공 (연결 설정: 연동정보: 클라우드 드라이버 정보, 크리덴셜 정보, 리전/존 정보 등 포함)
- 멀티 클라우드 자원 통합 구성 예시



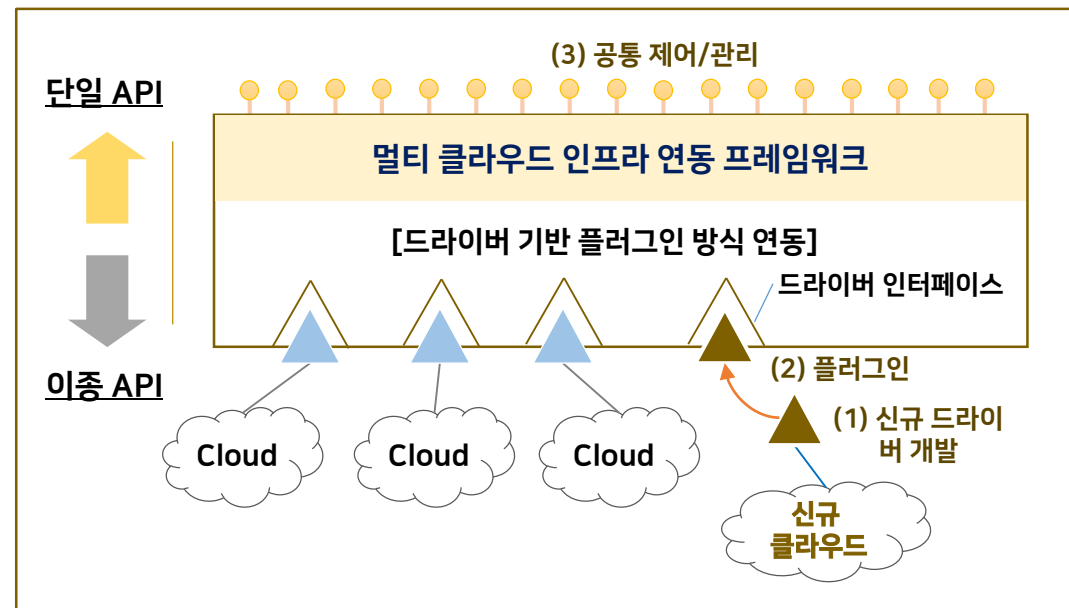
# 멀티 클라우드를 엮는 기술 - 드라이버/플러그인

- 플러그인 기반 멀티 클라우드 인프라 연동 및 공통 제어 방식  
→ 프레임워크 운영 중 신규 클라우드 동적 연동 가능

- 드라이버 공통 인터페이스
- 드라이버 플러그인 방식
- 공통 인터페이스 구현
- CSP 전용 API 사용한 구현
- 필요시 CSP기능 보정



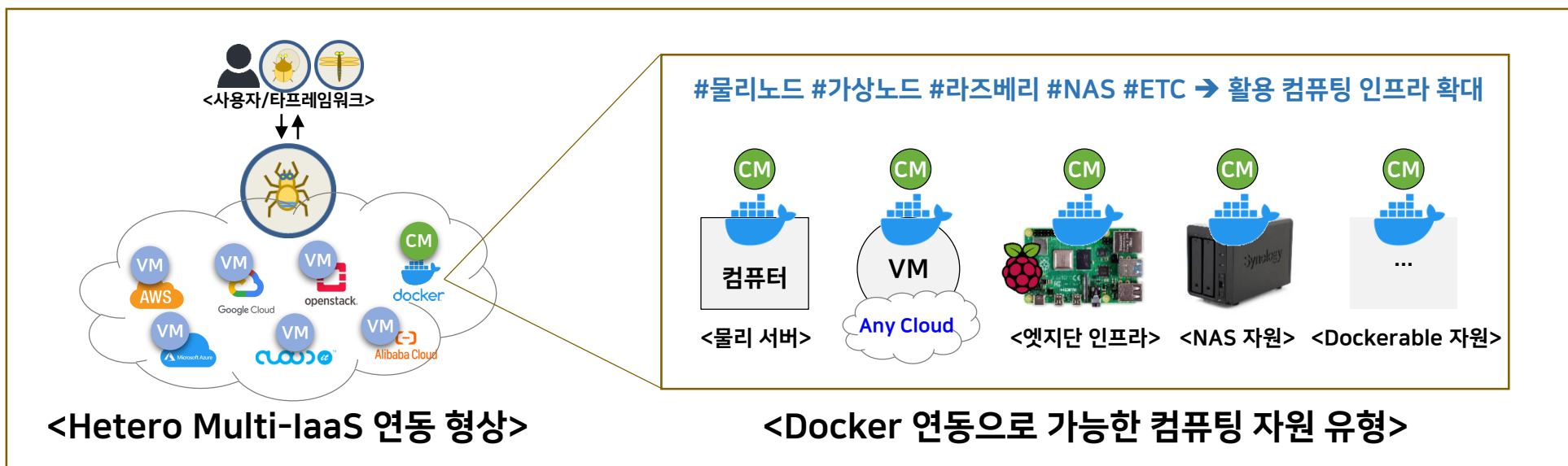
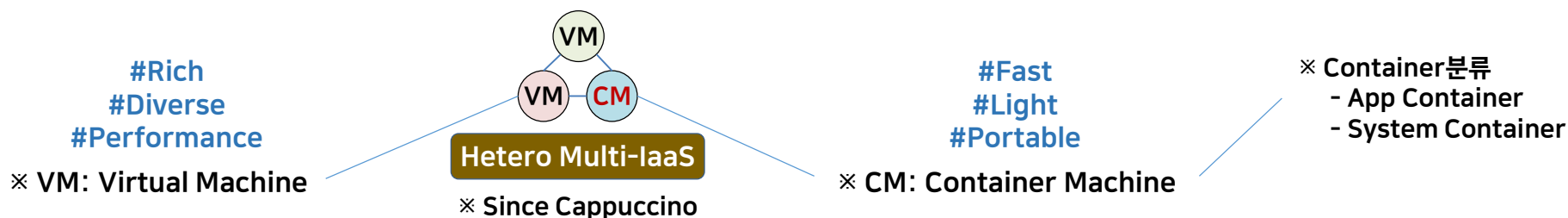
<인터페이스-드라이버 구조>



<클라우드 드라이버 플러그인 연동 개념>

# 다양한 인프라 제공 기술 - Hetero Multi-iaaS

- 국내외 퍼블릭 클라우드 연동, 대중화된 공개SW 오픈스택 클라우드 연동 (Multi-iaaS)
- 대중화된 도커 컨테이너(🐳) 연동 (Hetero Multi-iaaS)



# 단일 API 제공 기술 - 활용 예시

## [이종 클라우드-동일 코드] Image/Spec/VPC/Subnet/SecurityGroup/KeyPair/VM/CM 제어



```
export CONN_CONFIG=aws-ohio-config
export IMAGE_NAME=ami-f4f4cf91
export SPEC_NAME=t3.micro
```

```
./full_test.sh
```



```
export CONN_CONFIG=azure-northeu-config
export IMAGE_NAME=Canonical:UbuntuServer:18.04-LTS:latest
export SPEC_NAME=Standard_B1ls
```

```
./full_test.sh
```

※ 동일 코드: 3개의 인자 값 설정으로 멀티 클라우드 공통 제어 가능

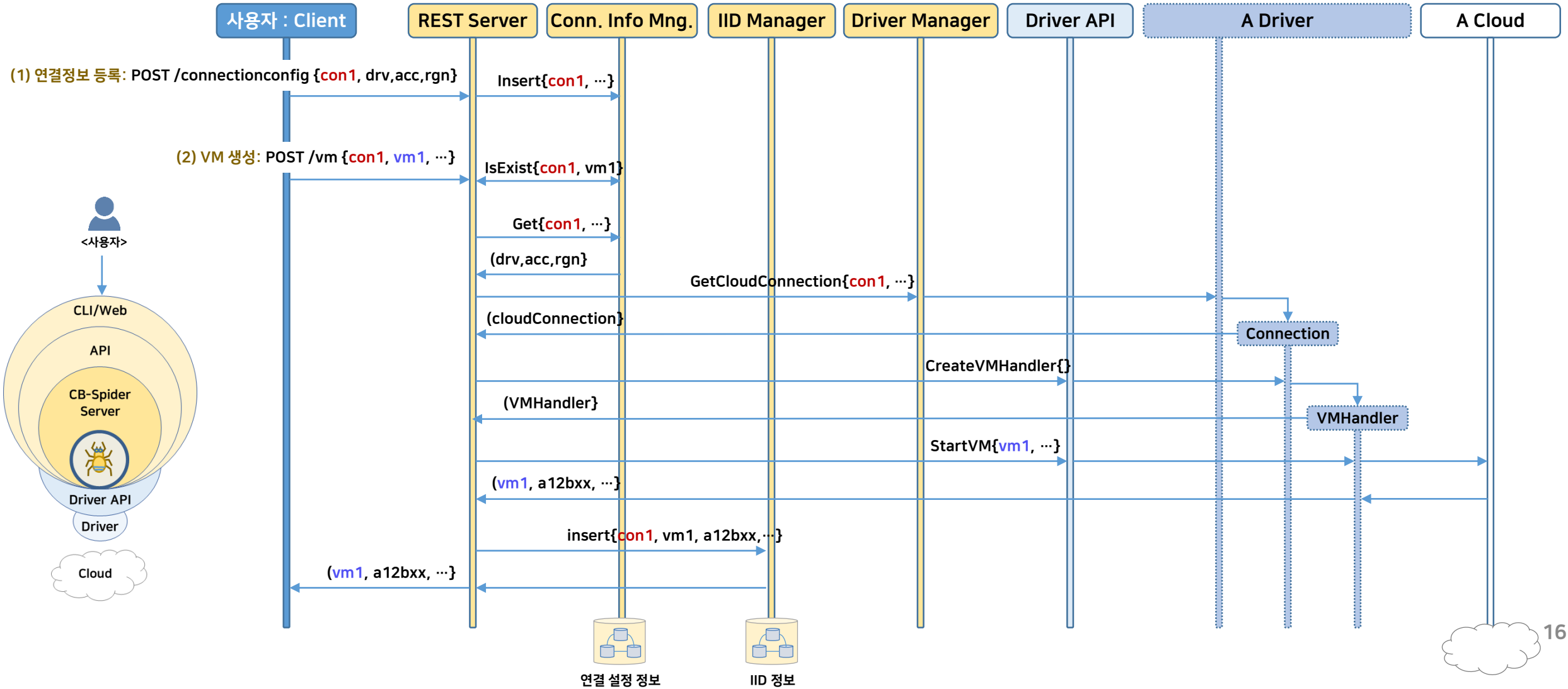
full\_test.sh:

```
echo "#####"
echo "## Full Test Scripts for CB-Spider IID Working Version - 2020.04.22."
echo "## 1. VPC: Create -> List -> Get"
echo "## 2. SecurityGroup: Create -> List -> Get"
echo "## 3. KeyPair: Create -> List -> Get"
echo "## 4. VM: StartVM -> List -> Get -> ListStatus -> GetStatus -> Suspend -> Resume -> Reboot"
echo "## -----"
echo "## 4. VM: Terminate(Delete)"
echo "## 3. KeyPair: Delete"
echo "## 2. SecurityGroup: Delete"
echo "## 1. VPC: Delete"
echo "#####"

echo "#####"
echo "## 1. VPC: Create -> List -> Get"
echo "#####"
curl -sX POST http://localhost:1024/spider/vpc -H 'Content-Type: application/json'
-d '{ "ConnectionName": "${CONN_CONFIG}",
      "ReqInfo": { "Name": "vpc-01",
                  "IPv4_CIDR": "192.168.0.0/16",
                  "SubnetInfoList": [ { "Name": "subnet-01", "IPv4_CIDR": "192.168.1.0/24" } ] } }'

...
```

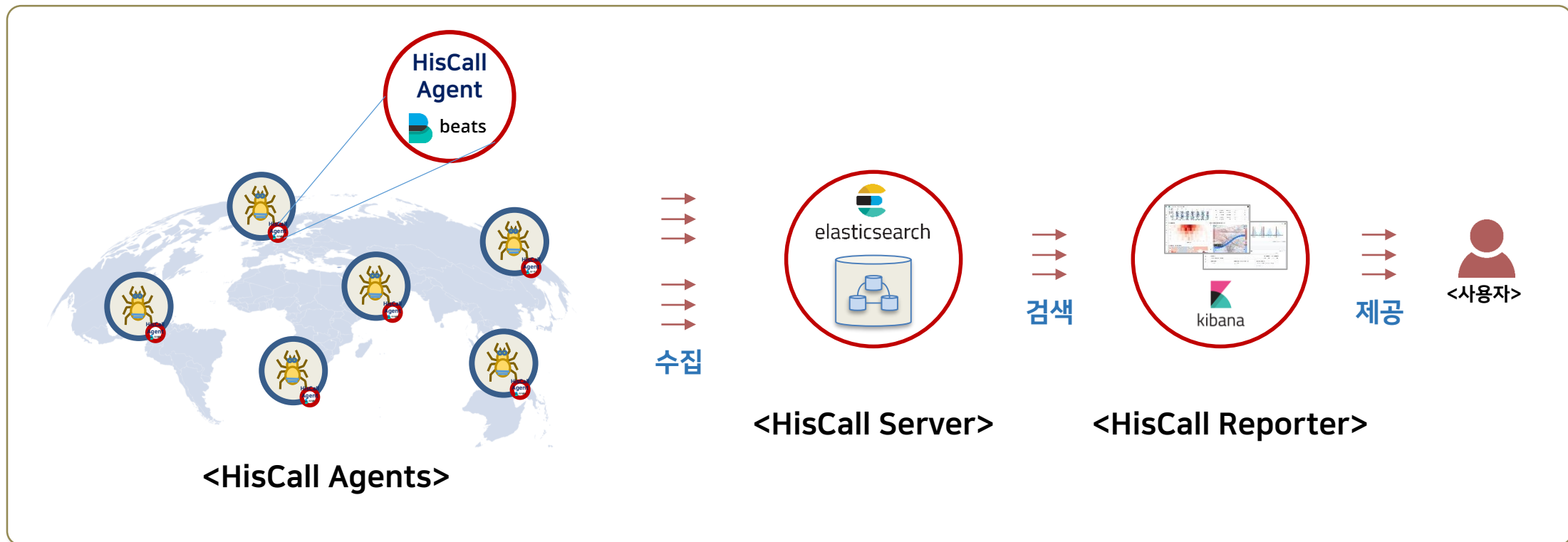
# 멀티 클라우드 공통 제어 기술 - VM 생성 예시





# 활용 극대화 지원 기술 - HisCall(History of Call)

- 모든 CB-Spider 프레임워크들의 멀티 클라우드 호출 이력 정보 수집 및 제공
- 개별 CSP API 호출 응답 시간 등의 연동 이력 정보를 아카이빙
- 아카이빙한 연동 이력 정보의 검색 및 다양한 차트를 제공
- CSP 연동 품질 분석, 인프라 자원 스케줄링 등에 활용 예정



# HisCall - 로그 출력 및 수집 정보 예시

## <Agent의 Call Log 출력 예시>

```
$ tail -f cb-spider/log/callog/callogs.log
```

```
[HISCALL].[13.124.44.241] 2020-11-05 12:27:46 (Thursday) ./cloud-driver/drivers/aws/resources.(*AwsVPCHandler).GetVPC():492 - "CloudOS" : "AWS", "RegionZone" : "us-east-2a", "ResourceType" : "VPC/SUBNET", "ResourceName" : "vpc-836f39ea", "CloudOSAPI" : "DescribeVpcs()", "ElapsedTime" : "0.2347", "ErrorMsg" : ""
```

```
[HISCALL].[13.124.44.241] 2020-11-05 12:27:47 (Thursday) ./cloud-driver/drivers/aws/resources.(*AwsVPCHandler).ListSubnet():959 - "CloudOS" : "AWS", "RegionZone" : "us-east-2a", "ResourceType" : "VPC/SUBNET", "ResourceName" : "ListSubnet", "CloudOSAPI" : "DescribeSubnets()", "ElapsedTime" : "0.2541", "ErrorMsg" : ""
```

```
[HISCALL].[13.124.44.241] 2020-11-05 12:33:39 (Thursday) ./cloud-driver/drivers/aws/resources.(*AwsVPCHandler).CreateVPC():75 - "CloudOS" : "AWS", "RegionZone" : "us-east-2a", "ResourceType" : "VPC/SUBNET", "ResourceName" : "vpc-01", "CloudOSAPI" : "CreateVpc()", "ElapsedTime" : "0.7470", "ErrorMsg" : "AuthFailure: AWS was not able to validate the provided access credentials status code: 401, request id: 0971a033-7b9e-4d11-8296-528bfb389063"
```

수 집

## HisCall 로그 Schema 규격 및 예시

ServerIP	TimeStamp	WeekDay	CloudOS	RegionZone	Resource Type	Resource Name	DriverAPI:Line	CloudOSAPI	ElapsedTime(sec)	ErrorMSG
13.124.44.241	2020-11-05 12:27:46	Thursday	AWS	us-east-2a	VPC/SUBNET	vpc-836f39ea	GetVPC():492	DescribeVpcs()	0.2347	-
13.124.44.241	2020-11-05 12:27:47	Thursday	AWS	us-east-2a	VPC/SUBNET	ListSubnet	ListSubnet():959	DescribeSubnets()	0.2541	-
13.124.44.241	2020-11-05 12:33:39	Thursday	AWS	us-east-2a	VPC/SUBNET	vpc-01	CreateVPC():75	CreateVpc()	0.7470	AuthFailure: AWS ...

# 활용 장벽 파괴 기술 - AdminWeb

- 목표: 웹 기반의 CB-Spider 관리 및 기능 시험이 가능한 개발자 지원 도구
- 특징
  - CB-Spider 서버와 밀접합된 경량의 embedded Admin용 Web 도구
  - CB-Spider 개발자 및 응용 개발자를 위한 간결하고 개발자스러운 웹 인터페이스 제공
  - 첫 사용자가 CB-Spider 프레임워크를 쉽게 사용해볼 수 있는 Start Guide 도구로 활용 가능
- 기능
  - 드라이버 정보, 크리덴셜 정보 및 리전/존 정보 등 클라우드 연결 설정 정보 관리
  - 멀티 클라우드 인프라 자원 생성 및 삭제 등 통합 제어
  - 사용자 선택에 적합한 입력 정보 샘플 제공
  - 손상된 ID 및 자원의 강제 삭제 등 Admin 기능 제공
- 향후
  - 서버 오류 메시지 출력 창 제공
  - 긴 실행 작업에 대한 progress bar 제공 등
  - 그 외 화면 및 편의성 개선 등

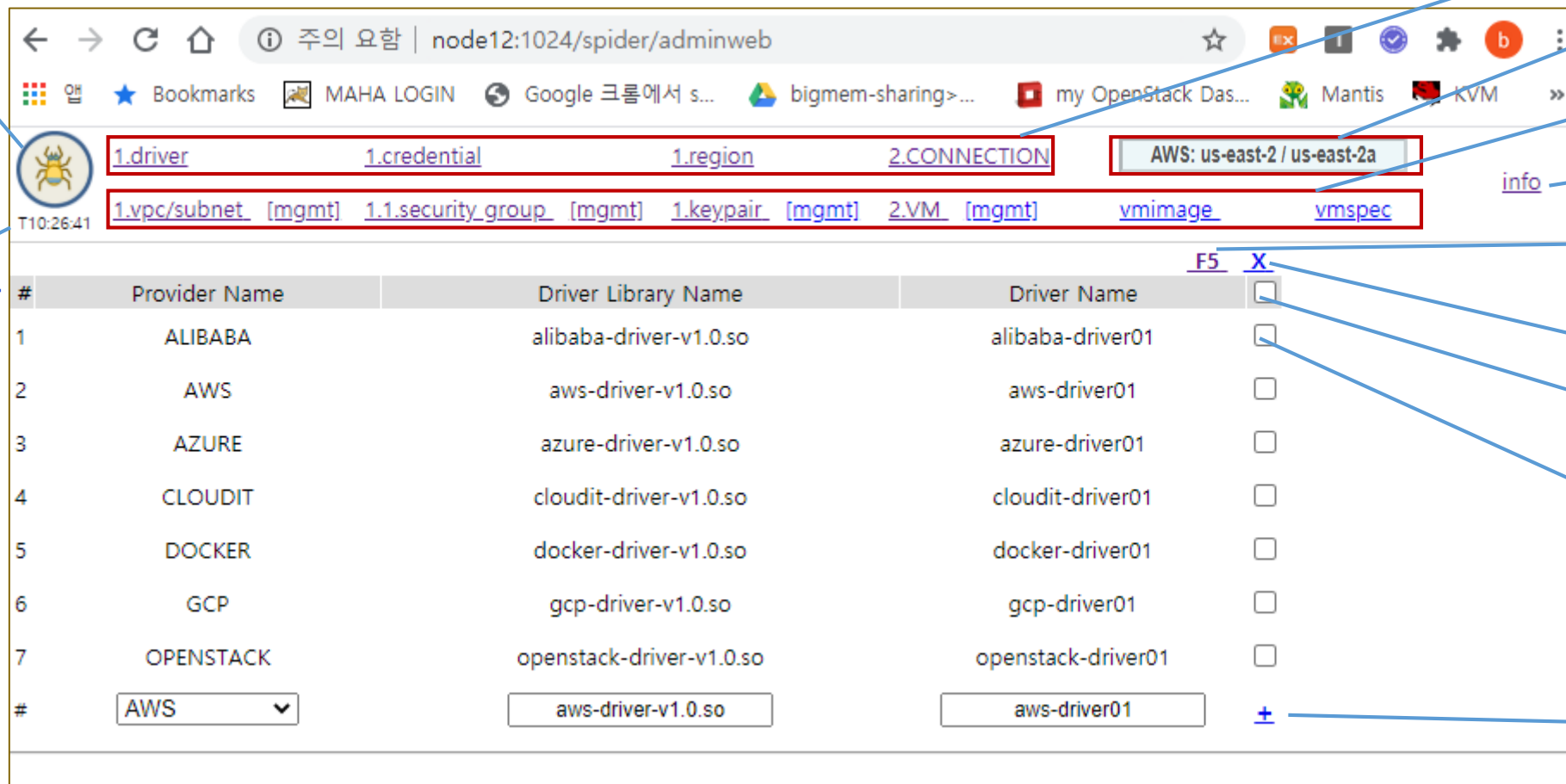


# AdminWeb - 접근 주소 및 화면 구성

- 접근주소: `http://{spider-server}:1024/spider/adminweb`
- 화면구성

• 홈버튼

• 시작시간



The screenshot shows the AdminWeb interface in a web browser. The address bar displays `node12:1024/spider/adminweb`. The top navigation bar contains several links: `1.driver`, `1.credential`, `1.region`, `2.CONNECTION`, and `AWS: us-east-2 / us-east-2a`. Below these links is a section with `1.vpc/subnet [mgmt]`, `1.1.security_group [mgmt]`, `1.keypair [mgmt]`, `2.VM [mgmt]`, `vmimage`, and `vm-spec`. The main content area features a table with the following columns: #, Provider Name, Driver Library Name, Driver Name, and a checkbox. The table lists seven providers: ALIBABA, AWS, AZURE, CLOUDIT, DOCKER, GCP, and OPENSTACK. Each provider has a corresponding driver library name and driver name. The checkbox column contains checkboxes for each provider. At the bottom of the table, there is a section for the selected provider (AWS) with a dropdown menu, a text input field for the driver library name, and a text input field for the driver name. To the right of these input fields is a button labeled 'Create Resources'.

• 클라우드 연결 설정 정보 관리

• 현재 연결 설정 정보 표시

• 클라우드 자원/VM 관리

• 서버 및 API 정보

• refresh

• delete

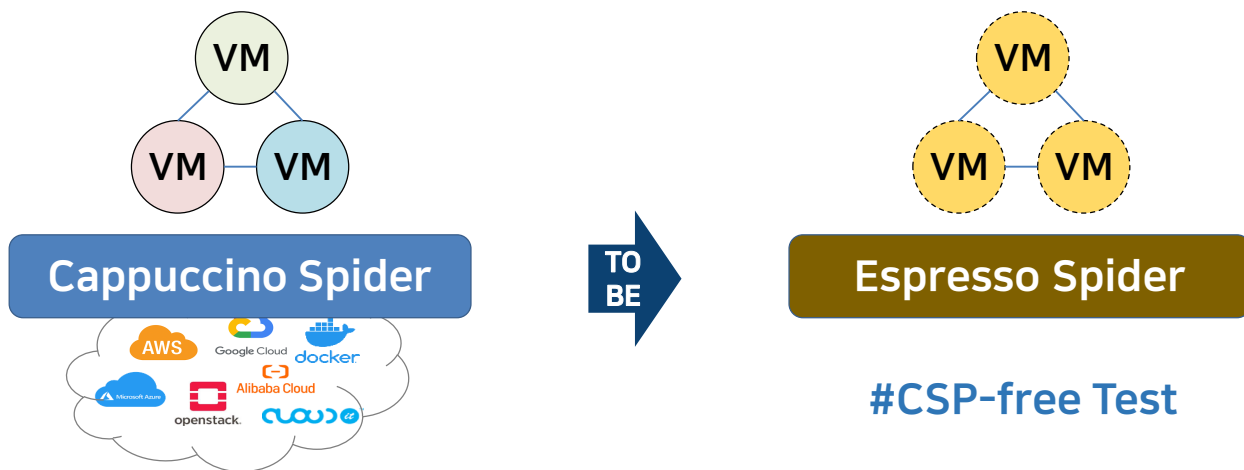
• Select/Unselect all

• Select/Unselect

• Create Resources

# 활용 장벽 파괴 기술 - Mock Driver

- 활용측면: 활용 장벽 해소를 위하여 Mock Driver 기반의 쉬운 사용 경험 제공 가능 #User-Friendly  
→ CSP 없이(계정/크리덴셜 정보 발급 없이) Spider 기능 맛보기
- 개발측면: CSP 연동 없는 시험/디버깅 등 쉬운 개발 환경 지원 가능 #Developer-Friendly  
→ GitHub:PR/MERGE → CI&Test workflow 구축 기반 마련

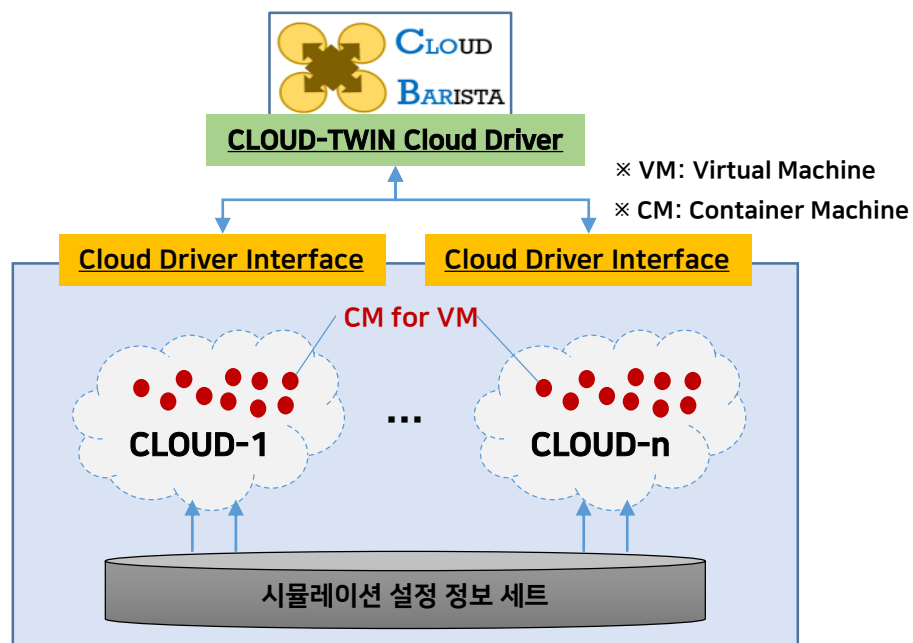


※ Download = 스파이더 시험 및 개발 환경

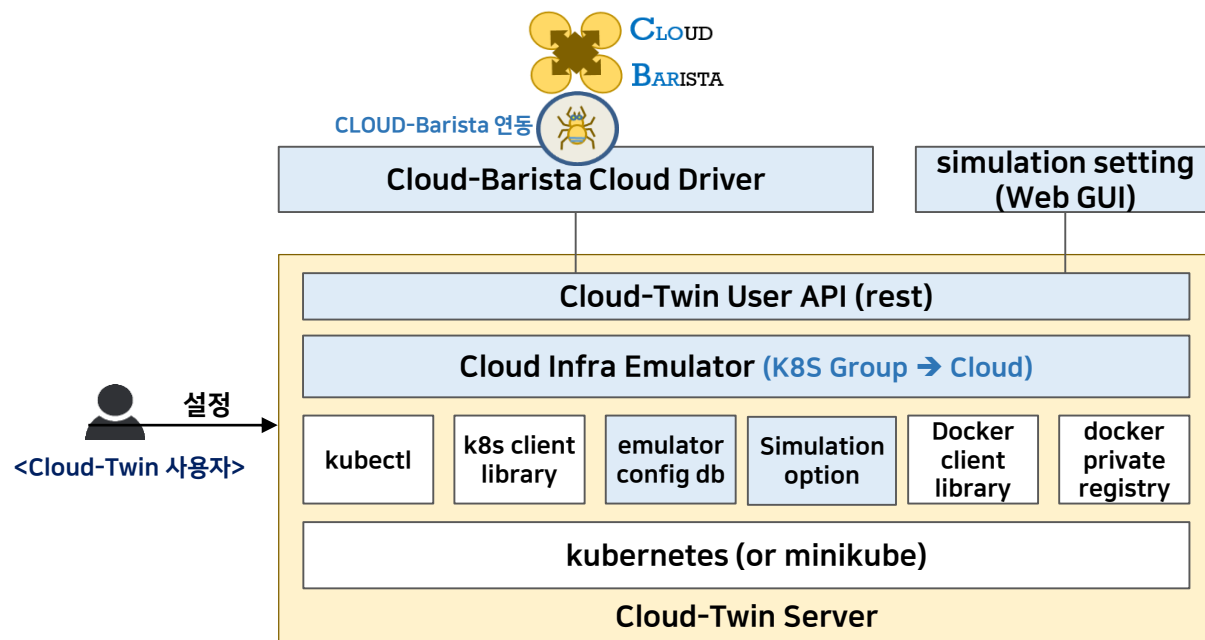


# 활용 장벽 파괴 기술 - Cloud-Twin

- 목적: 대규모 멀티 클라우드 인프라 가상 시험 환경을 제공하기 위한 에뮬레이터 시스템(저비용, 대규모 가상 클라우드)
- 특징: Kubernetes 기반 대규모 가상 인프라 환경 주요 기능 제공(에이전트 탑재 및 대규모 모니터링 정보 제공)
- Cappuccino : 대규모 모니터링 데이터 시험 환경 제공을 위한 Dragonfly-Agent Container 버전 통합 PoC 완료
- Espresso: Dragonfly 정식 통합 및 대규모 모니터링 데이터 부하 시험을 위한 가상 클라우드 인프라 환경 제공



<Cloud-Twin 개념>



□ : 주 개발 모듈 □ : 공개SW 활용

<Cloud-Twin 내부 구조>

# CB-Spider History 및 개발 현황

2019.03. ~

## farmoni

- CB-Spider 전신
- Static 연동(하드코딩)
- 3종 클라우드 연동

farmoni



2019.06. ~

## poc-cb-spider

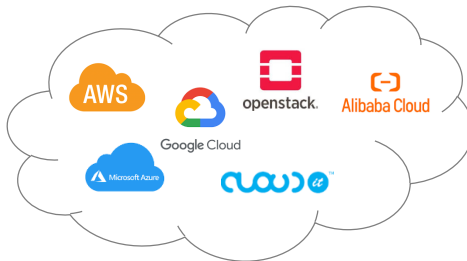
- CB-Spider PoC
- Driver Plugin 도입
- 4종 클라우드 연동



2019.09. ~

## Americano

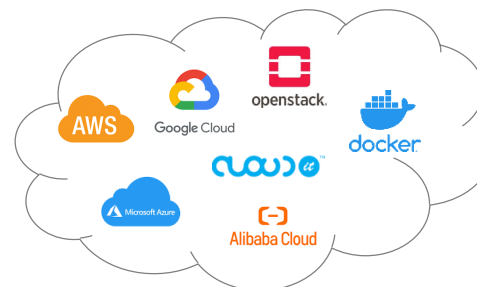
- MC-laaS 제공
- cb-store, cb-log 도입
- 6종 클라우드 연동



2019.12. ~

## Cappuccino

- Hetero MC-laaS 제공
- Americano 리모델링
- 7종 클라우드 연동



2020.06. ~

## Espresso

- 활용.확산 극대화 도구
- 분산 스파이더 PoC 검증
- 7종 클라우드 연동

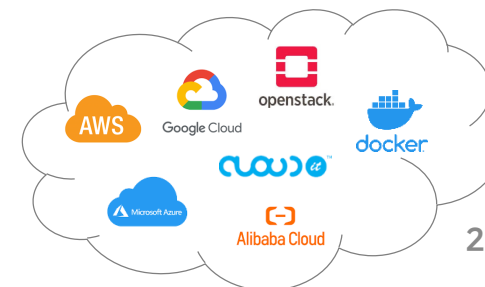
#AdminWeb

#HisCall










#MeerKat

#Mock Driver



## <연동 클라우드>

## <공통 제어 자원>

	IMG	Spec	VPC/ Sub	Sec	Key	VM
	IMG	Spec	VPC/ Sub	Sec	Key	VM
	IMG	Spec	VPC/ Sub	Sec	Key	VM
	IMG	Spec	VPC/ Sub	Sec	Key	VM
	IMG	Spec	VPC/ Sub	Sec	Key	VM
	IMG	Spec	VPC/ Sub	Sec		VM
	IMG					VM
Mock-Driver	IMG	Spec	VPC/ Sub	Sec	Key	VM
Cloud-Twin	IMG					VM

## <범례>

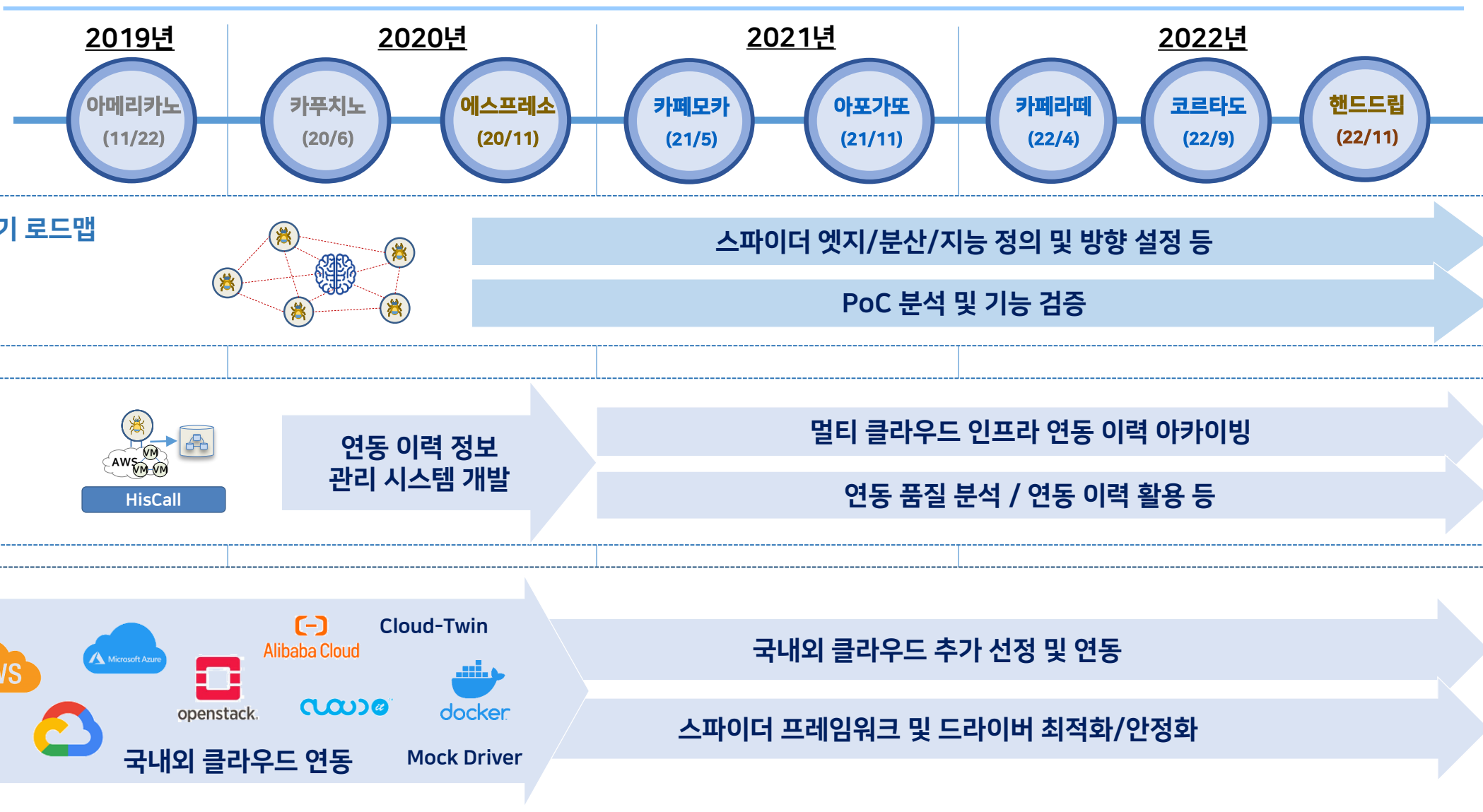
- IMG : VM 이미지
- Spec : VM 사양
- VPC/  
Sub : 가상 네트워크
- Sec : 보안그룹
- Key : VM 키페어
- VM : 가상서버(VM, CM)

※ 기능 에뮬레이션

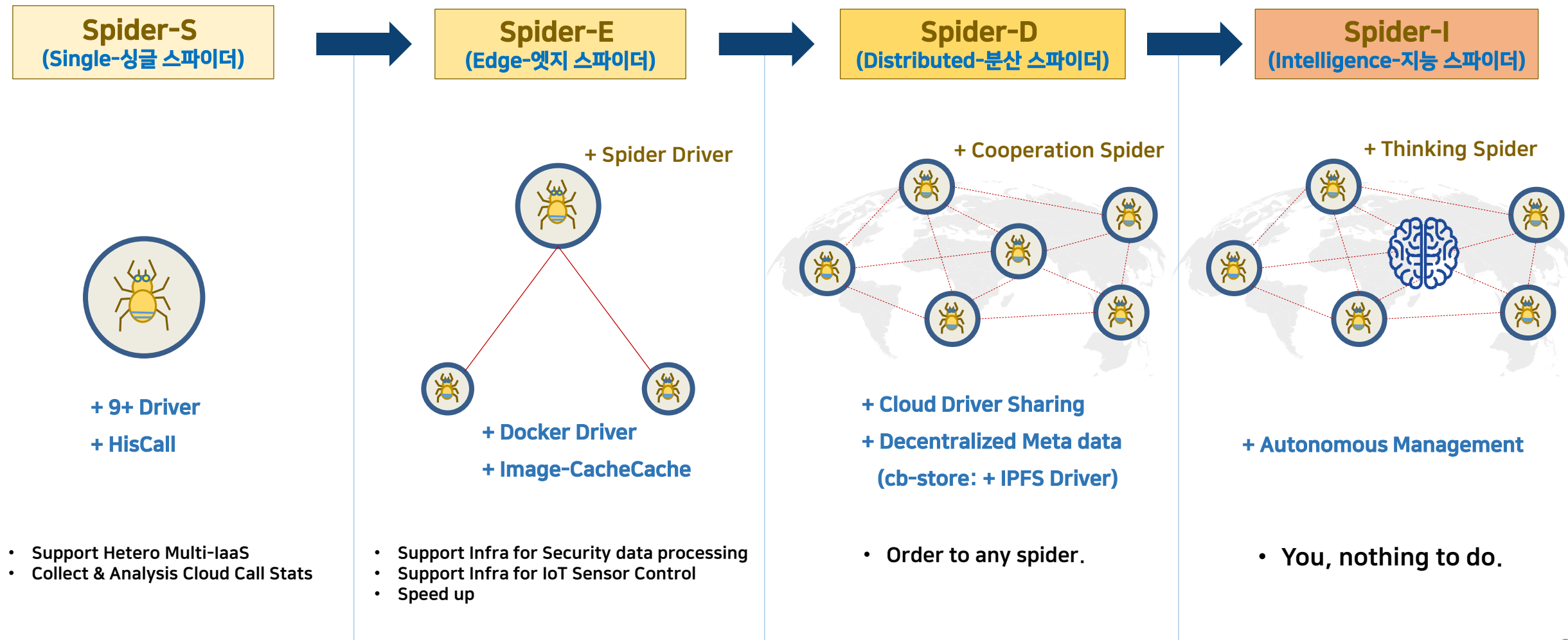
※ VM 모니터링  
※ 기능 시뮬레이션



# CB-Spider 프레임워크 개발 로드맵



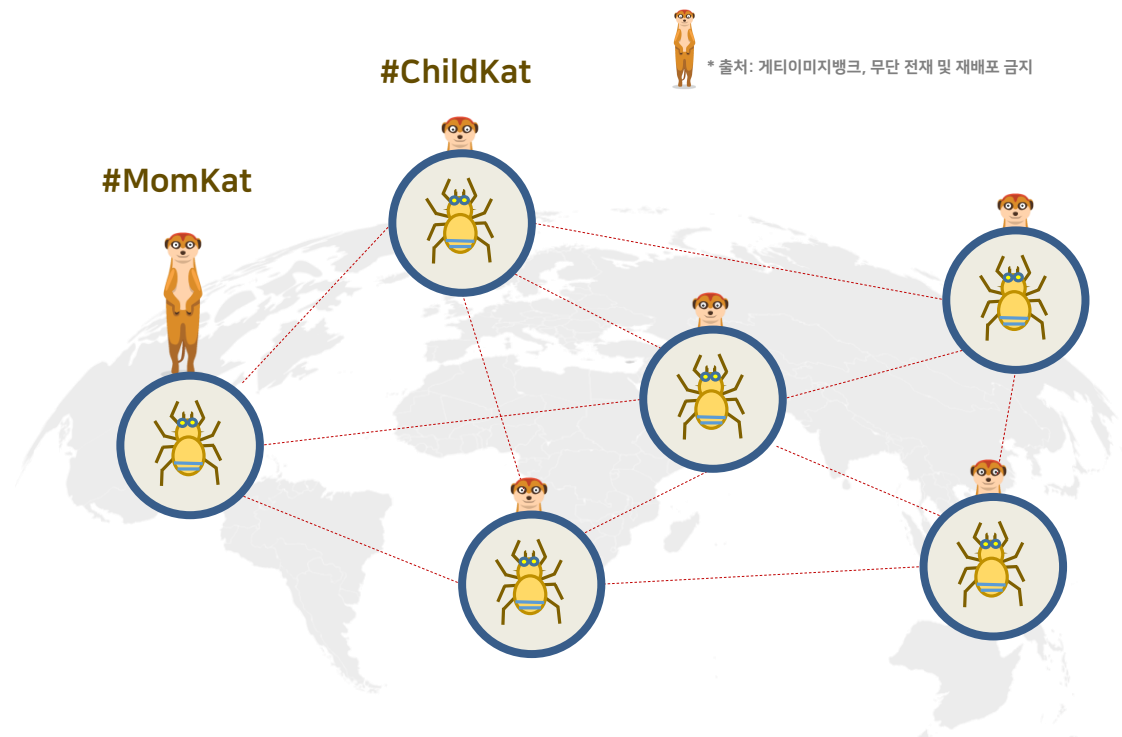
# 장기 로드맵 - 엣지→분산→지능



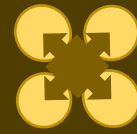
※ 전 세계 클라우드를 엮는 그날까지...

# 분산 스파이더 PoC - 미어캣 프로젝트

- 목표: CB-Spider 간 분산/협업을 위한 핵심 기능 제공
- 특징
  - 탈중앙 방식의 분산 관리 미어캣 메커니즘 제공
    - 탈중앙 방식: Decentralized, Masterless
  - 타 활용.확산을 고려한 탈부착 구조
- 기능
  - 분산 Spider 목록 관리
  - 분산 Spider 상태 관리
  - 분산 Spider 명령 제어
- 기타: 공유 저장소
  - PoC: Google Sheets 기반 신속한 MeerKat 메커니즘 검증
  - 향후: memcache(성능), IPFS(분산) 등 향후 문제 분석에 따른 개선



<분산 스파이더: MeerKat 개념도>



CLOUD

BARISTA

멀티클라우드 서비스 공통 플랫폼

# CB-Spider 기술 시연

- 무한 확장의 첫걸음, 분산 스파이더 PoC 기술 시연 -

에스프레소(Espresso) 한잔 어떠세요 ? ^^

# CB-Spider 기술 시연

- 진입 장벽 해소를 위한, AdminWeb 기반 스파이더 활용 시연 -

에스프레소(Espresso) 한잔 어떠세요 ? ^^

# CB-Spider를 만들어 가는 사람들



Jeongwoon



swsok



KANG DONG JAE  
dongjae



Seokho Son  
seokho-son



Yoo Jae-Sung  
dev4unet



jaehong park  
iworkist



hellowkorea



CAN(고정완)  
ghojeong



sejinP  
sejin-P



ByoungSeob Kim  
powerkimhub



Jihoon Seo  
jihoon-seo



Jin Park  
engineer-pjin



kyongminkwon



Sean Oh  
innodreamer



hyokyungk



zephy-Jeong  
zephy-mzc



Jin-Whee-Park  
Jin-Whee-Park



ChoiNakSoo



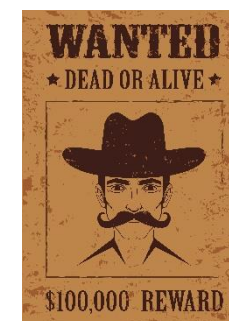
yoonjiO  
yoonjiOh



cloud-barista-hub



jmleefree



# 감사합니다.

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

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

## “Journey to the Multi-Cloud”

클라우드 바리스타들의 세 번째 이야기

### Cloud-Barista Community the 3<sup>rd</sup> Conference