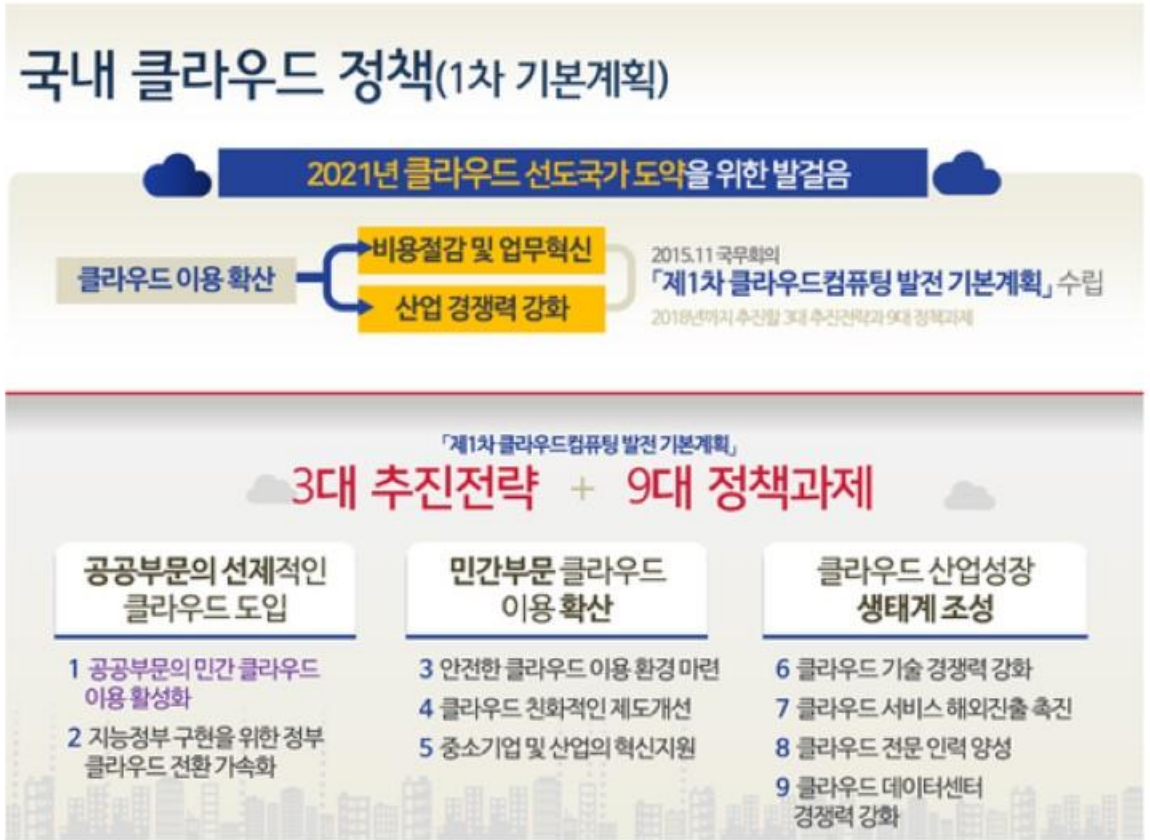

openmaru
APM
클라우드 네이티브를 위한
SaaS 형 OPENMARU APM

공공 클라우드 전환 로드맵 손질...

클라우드 네이티브 부처별 추진·2030년 완료



1차 클라우드 컴퓨팅 발전 기본계획 당시 정부의 클라우드 방향성(출처: NIA)

클라우드 네이티브 전환 로드맵 수립 설명회

– 행정안전부 , 2023년 5월 3일

공공 클라우드 네이티브 전환 로드맵	
추진 배경	• 정부 재정투자 방향 변화 • 보안인증제 개편 • 신기술 보편화 등
추진 환경	• 행정안전부 전환 사업 예산 축소 • 정부 정책 "민간 클라우드 우선 이용" • "클라우드 네이티브 우선 적용"으로 발전 등
추진 방향	• 범정부 정보자원 등록 및 관리시스템에 등록된 모든 시스템의 클라우드 네이티브 전환
전환기간	• 2024년부터 2030년까지 7개년 추진
추진 목표	• 2023년 시범 사업 • 2024년 10% 달성 • 2025년 30% 달성 • 2026년 이후 70% 달성

디지털플랫폼정부는 '클라우드 네이티브'로 구축하는 게 핵심이다

단순 클라우드 전환이 아닌 , 클라우드 네이티브 전환 부터 다시 작성해야 함

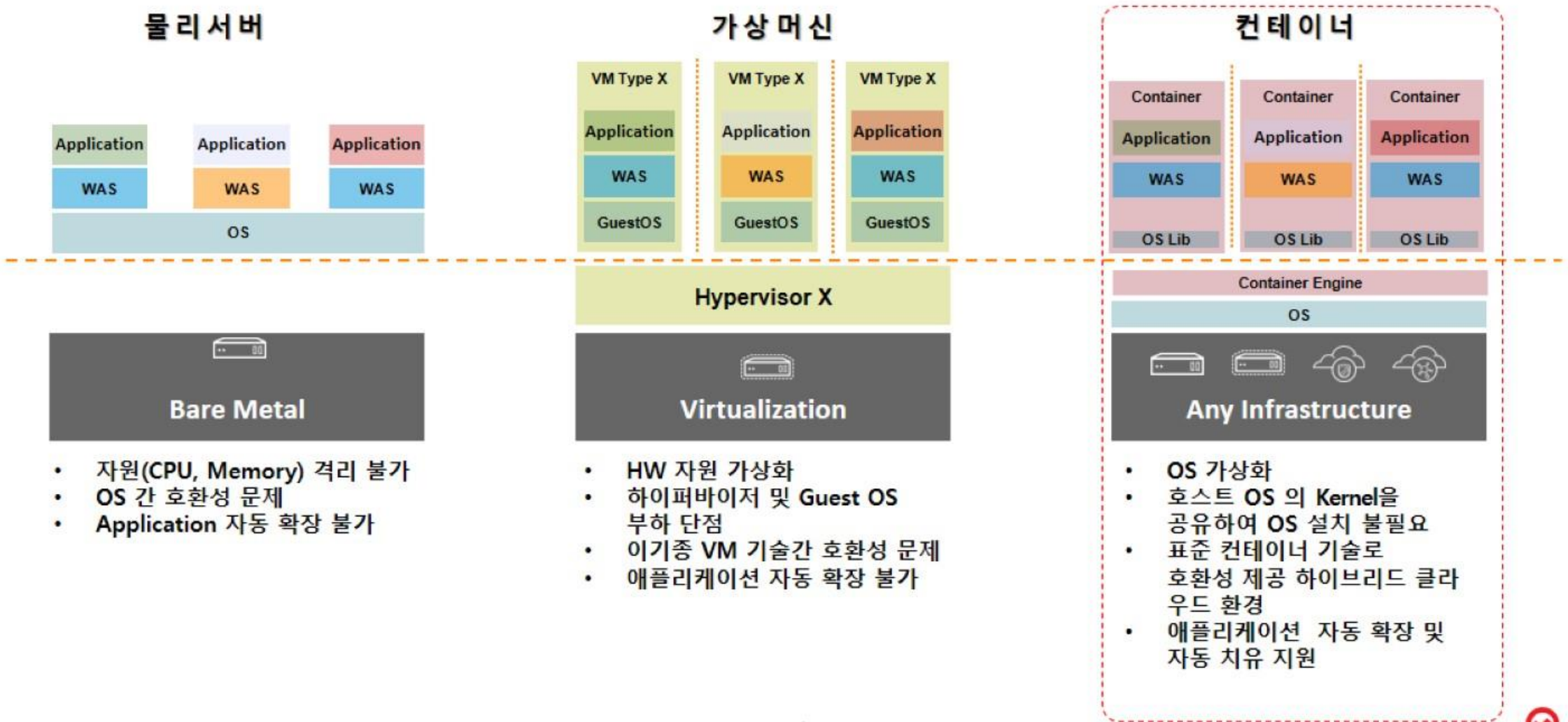
디지털플랫폼정부는 초거대AI를 비롯한 디지털 기술을 적극 도입, 칸막이를 없애고 '원팀 정부'로 거듭나는 것을 목표한다. 이를 위한 **공공 플랫폼을 단순 클라우드 전환이 아닌, 클라우드를 클라우드 답게 쓸 수 있도록 '클라우드 네이티브'로 구축하는 게 이번 정책의 핵심**이다. 공공SW(소프트웨어) 사업 고질병을 극복할 해법을 제시하면서 각종 미래 IT 산업 육성의 요람이 될 것으로 기대된다.

Application Performance Management

Container

컨테이너 차별성

자원 효율성, 자원 격리, 호환성, Auto Scaling, DevOps, MSA, 관리 편의성



- 자원(CPU, Memory) 격리 불가
- OS 간 호환성 문제
- Application 자동 확장 불가

- HW 자원 가상화
- 하이퍼바이저 및 Guest OS 부하 단점
- 이기종 VM 기술간 호환성 문제
- 애플리케이션 자동 확장 불가

- OS 가상화
- 호스트 OS 의 Kernel을 공유하여 OS 설치 불필요
- 표준 컨테이너 기술로 호환성 제공 하이브리드 클라우드 환경
- 애플리케이션 자동 확장 및 자동 치유 지원

VM 의 고질적인 문제점

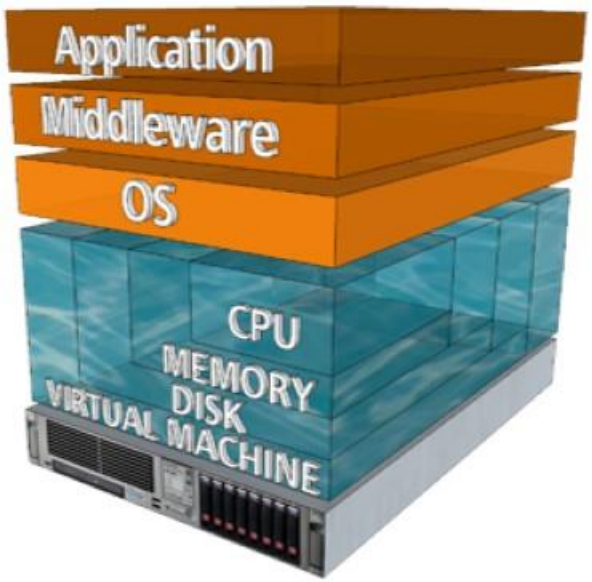
가상화 기술은 Guest OS 로 인하여 VM 간 불일치와 느린 시작시간과 이미지가 비대함

거대한 이미지 사이즈

- VM 을 템플릿 관리는 하지만 사이즈가 커서 재사용성을 높이기 어려움

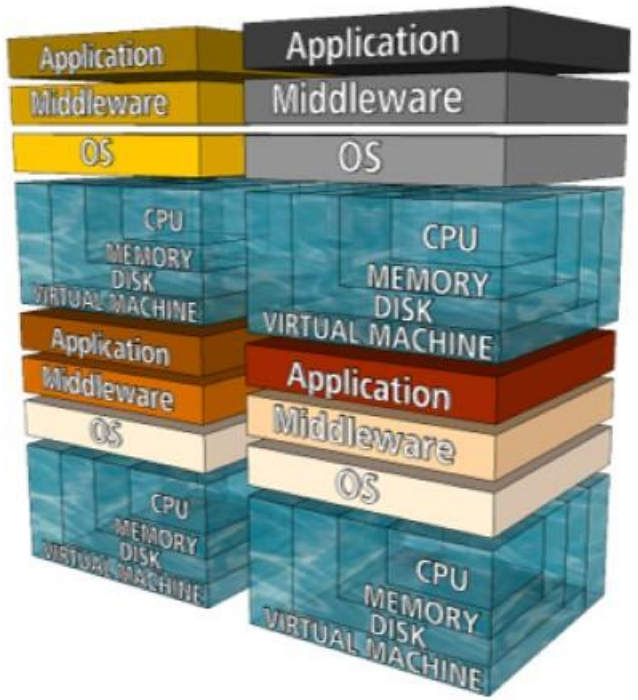
느린 시작 시간

- 부팅 시 Hypervisor – OS- 미들웨어 – 애플리케이션까지 실행 되어야함



VM 간의 환경 불일치

- VM 생성 후 개별로 변경 사항을 관리하기 때문에 VM 간 구성이나 환경이 불일치



오버헤드 - Containers vs. VMs

Guest OS 의 부팅으로 인하여 시간이 지연됨

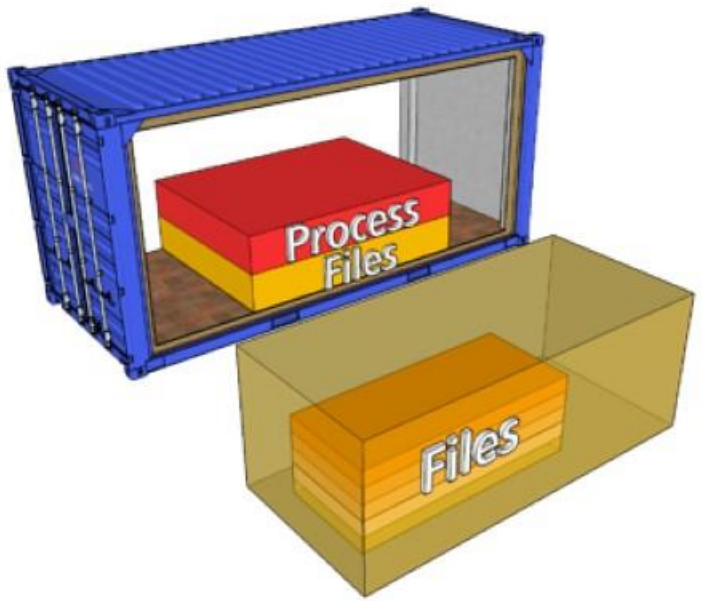
- OS에서 응용 프로그램을 작동하는 경우, 하드웨어 가상화에서는 가상화 된 하드웨어 및 하이퍼바이저를 통해 처리하기 때문에 물리적 시스템보다 처리에 부가적인 시간 (오버 헤드)가 필요
- 컨테이너 형 가상화 커널을 공유하고 개별 프로세스가 작업을 하는 것과 같은 정도의 시간 밖에 걸리지 않기 때문에 대부분 오버 헤드가 없음

컨테이너를 통한 가상서버 문제점 해결

컨테이너 기술은 하이브리드 클라우드 와 DevOps 그리고 MSA 구현을 필수

작은 이미지 사이즈

- 컨테이너는 레이어 개념으로 이미지에 파일을 추가/삭제하여 관리함
- 레이어 사이즈를 최적화하여 이미지 사이즈를 최소화



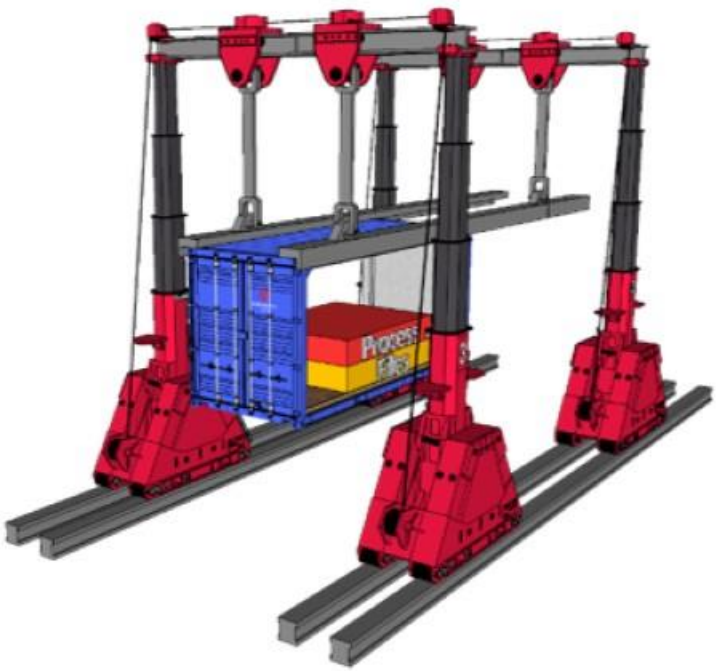
빠른 시작 시간

- 컨테이너는 분리된 프로세스 형식으로 OS 부팅이 필요 없기 때문에 부팅 시간을 최소화 할 수 있음

Container = Process

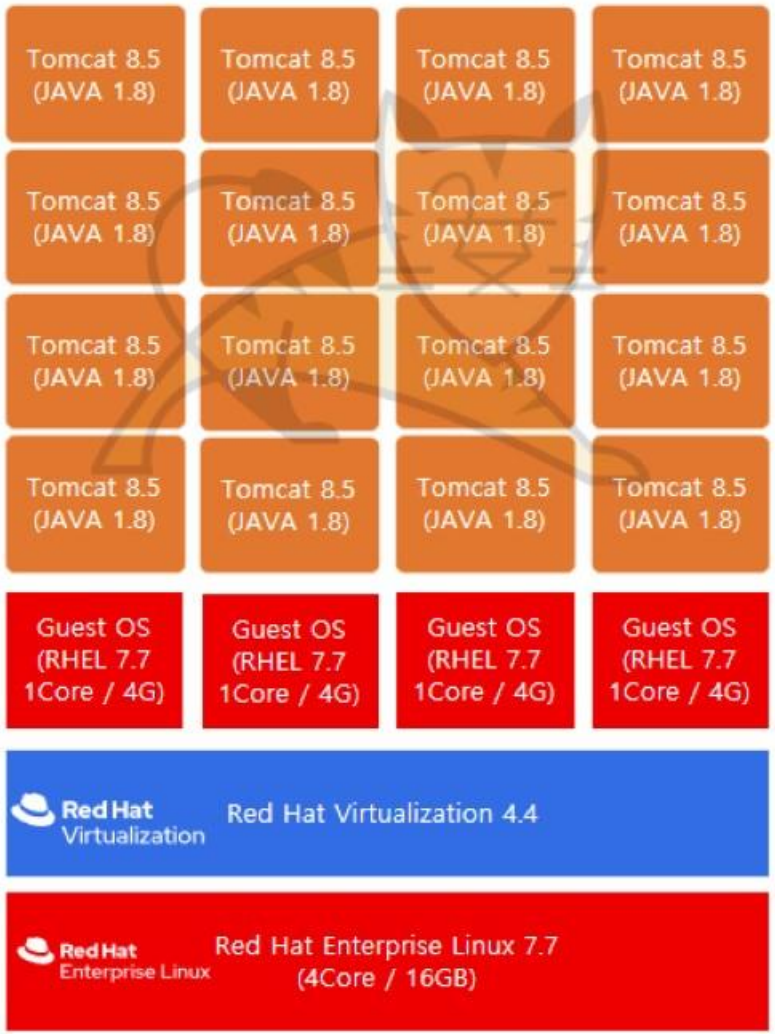
높은 이동성(Portability)

- 애플리케이션에 필요한 라이브러리나 의존 파일들을 이미지에 포함하기 때문에 환경에 의한 발생되는 문제가 거의 없음



가상화와 컨테이너 집적도 비교

동일한 H/W 에서 가상화 대비 컨테이너 집적도가 수 배 이상 높음



Application Performance Management

Cloud Native Computing

클라우드 네이티브를 저해하는 요인

생각과 시스템을 클라우드 네이티브 하게 전환하지 못하면, 클라우드라도 개선이 없음

클라우드를 임대하여 사용하는 것일 뿐

- 가상머신과 스토리지를 임대하여 사용하고 있을 뿐, 기존의 인프라와 다르지 않음

클라우드 특징에 맞게 설계하고 운영 하지 않음

- 클라우드 특성을 이해하지 못하고 기존 인프라를 단순히 대체하여 설계
- 비용 부분에서만 정액제 클라우드로 전환하였으나, 벤더 종속성과 비용만 높아짐

인프라만 클라우드 일 뿐 조직은 그대로

- 기존의 개발팀과 운영팀이 수행하던 역할과 프로세스 그대로 운영

클라우드로 전환했으나 구인난과 고비용 구조로 더 큰 문제

- 클라우드를 이용하고 있음에도 불구하고 수작업 프로세스에서 벗어나지 못함
- 운영 인력 부족과 업무 효율성을 개선하지 못함

클라우드 네이티브 시대

가상화 시대를 넘어 클라우드 네이티브 시대로

- **애플리케이션을 실행하기 위한 최적의 인프라 최적 솔루션 중 하나**
  - **컨테이너 기술**
    - 애플리케이션이 동작하기 위한 운영 환경을 함께 패키징
    - 개발자를 위한 이미지 빌드/배포 용이성
    - 빠른 애플리케이션 실행과 낮은 오버헤드
- **Kubernetes(컨테이너 오케스트레이션)**
  - **애플리케이션 실행을 먼저 생각할 때**
  - **어떤 인프라를 만들지 주축으로 설계된 인프라 기반**

Develper Experience장점

Reconcilliation model의 정교함

Cloud Native 에서 Native 는?

클라우드 네이티브는 "클라우드가 '클라우드 다울 수 있도록' 애플리케이션을 구축, 실행하는 방식"

네이티브(Native)의 사전적 의미는 '선천적인', '본래' 등이다.

'어린이 또는 성인이 되어 언어를 배운 것이 아닌 태어나서 부터 특정 언어를 사용해 온 사람'

네이티브 스피커

'어린이 또는 성인이 되어 스마트폰을 접한 것이 아닌 유아기부터 스마트폰을 사용해 온 사람'

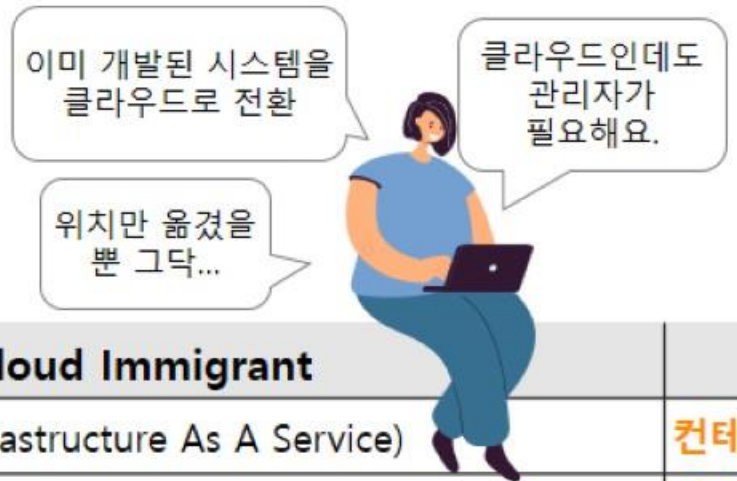
스마트폰 네이티브

'애플리케이션을 계획/설계할 때부터 클라우드 특징과 장점을 기반으로 개발/운영'

클라우드 네이티브

Cloud Native vs. Cloud Immigrant

가상화 기반 IaaS vs. 컨테이너 기반 PaaS



구분	Cloud Immigrant	Cloud Native
서비스 모델	**가상화** 기반 IaaS (Infrastructure As A Service)	**컨테이너** 기반 PaaS (Platform As A Service
디자인	**On Premise 에 구축된 시스템을 클라우드로 이전하여** 운영	**시작 단계부터 클라우드의 장점인** 민첩성, 확장성 그리고 이동성을 최대한 활용할 수 있도록 설계
구현	특정 클라우드 벤더에 의존적인 설정이 있어 구축에 시간이 걸림	어떤 클라우드 환경에서도 빠르고 효율적으로 전환 (**Portability**)
확장성	애플리케이션 업데이트가 **수작업이기** 때문에 장시간의 다운타임일 필요하고 **Scale In/Out 이 어려움**	**컨테이너와 MSA 기반으로 서비스에** 영향을 주지 않고, 업데이트가 필요한 서비스만 변경할 수 있으며, 서비스 단위의 **Scale In/out** 지원
비용	애플리케이션이 커질 수록 인프라 비용이 상승	인프라 부분의 종속성이 없어 비용이 저렴
유지보수	**버전관리, 설치 그리고 구성관리가 수작업**이며 복잡함	**CI** (Continuous Integration) / **CD** (Continuous Delivery)

CNCF Cloud Native Definition v1.0

클라우드 네이티브 선언문 1.0 - 클라우드 네이티브 컴퓨팅 재단

클라우드 네이티브 기술을 사용하는 조직은 현대적인 퍼블릭, 프라이빗, 그리고 하이브리드 클라우드와 같이 동적인 환경에서 확장성 있는 애플리케이션을 만들고 운영할 수 있다.

컨테이너, 서비스 메시, 마이크로서비스, 불변의 인프라스트럭처, 그리고 선언적 API가 전형적인 접근 방식에 해당한다.

이 기술은 회복성이 있고, 관리 편의성을 제공하며, 가시성을 갖는 느슨하게 결합된 시스템을 가능하게 한다.

견고한 자동화와 함께 사용하면, 엔지니어는 영향이 큰 변경을 최소한의 노력으로 자주, 예측 가능하게 수행할 수 있다.

Cloud Native Computing Foundation은 **벤더 중립적인 오픈소스 프로젝트 생태계**를 육성하고 유지함으로써 해당 패러다임 채택을 촉진한다.

우리 재단은 최신 기술 수준의 패턴을 대중화하여 이런 혁신을 누구나 접근 가능하도록 한다.

Source:https://github.com/cncf/toc/blob/master/DEFINITION.md#%ED%95%9C%EA%B5%AD%EC%96%B4

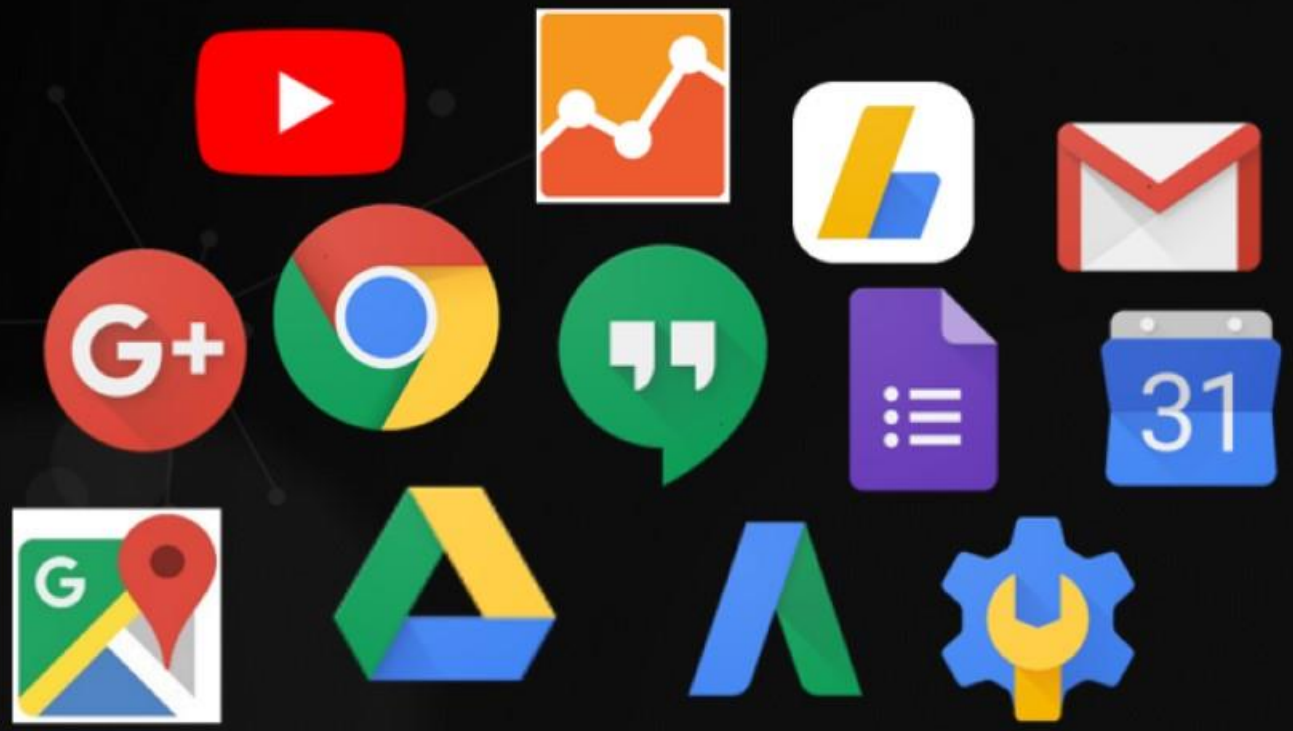
Cloud Native

Kubernetes

Google 의 모든 서비스는 컨테이너 에서 실행

openmaru

- Gmail , 검색, 지도 ...
- MapReduce , GFS , Colossus ...
- Google Compute Engine 가상 머신도 컨테이너 에서 실행!
- 매주 20 억개 이상의 컨테이너 를 실행 중



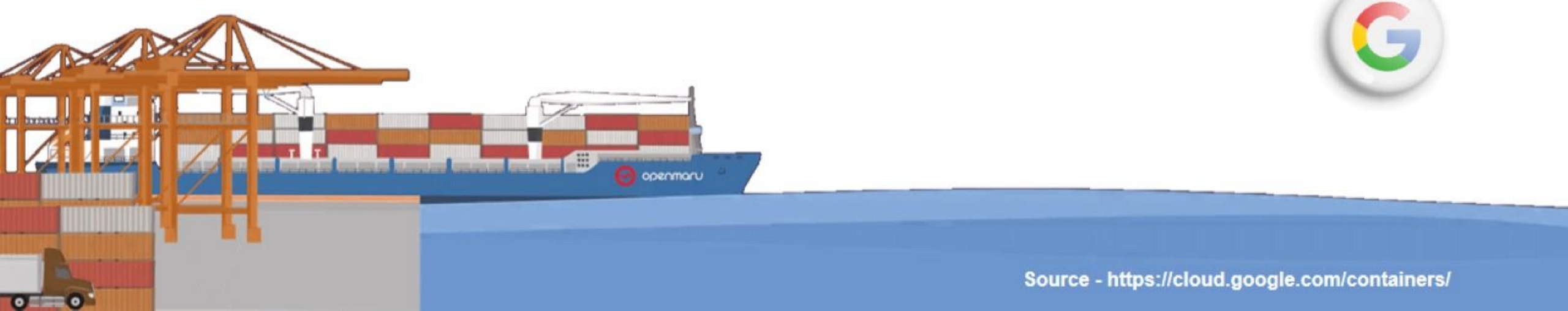
GOOGLE 과 컨테이너

Google의 업무 방식

Gmail에서 YouTube, 검색에 이르기까지 Google의 모든 제품은 컨테이너에서 실행됩니다.

개발팀은 컨테이너화를 통해 더욱 신속하게 움직이고, 효율적으로 소프트웨어를 배포하며 전례 없는 수준의 확장성을 확보할 수 있게 되었습니다. Google은 매주 수십억 개가 넘는 컨테이너를 생성합니다. 지난 10여 년간 프로덕션 환경에서 컨테이너화된 워크로드를
실행하는 방법에 관해 많은 경험을 쌓으면서 Google은 커뮤니티에 계속 이 지식을 공유해 왔습니다.

초창기에 cgroup 기능을 Linux 커널에 제공한 것부터 내부 도구의 설계 소스를 Kubernetes 프로젝트로 공개한 것까지 공유의 사례는 다양합니다. 그리고 이 전문 지식을 Google Cloud Platform으로 구현하여 개발자와 크고 작은 규모의 회사가 최신의 컨테이너 혁신 기술을 쉽게 활용할 수 있도록 하였습니다.



Source - https://cloud.google.com/containers/

About Kubernetes

- 쿠버네티스(K8s)는 컨테이너화된 애플리케이션을 자동으로 배포, 스케일링 및 관리해주는 오픈소스 소프트웨어
- 쿠버네티스", "쿠베르네테스","K8s", "쿠베", "쿠버", "큐브"라고 부르며
- Go로 작성된 오픈 소스 , 오픈소스 S/W (Apache License 2.0) 라이선스
- 리눅스 재단 (Linux Foundation)산하 Cloud Native Computing Foundation (CNCF) 에서 관리
- 구글에서 개발하고 설계한 플랫폼으로서 사내에서 이용하던 컨테이너 클러스터 관리 도구인 "Borg"의 아이디어를 바탕으로 개발

> "Kubernetes is open source-a contrast to Borg and Omega, which were developed as purely Google-internal systems. "
>
> *- Borg, Omega, and Kubernetes*

Open Source Software

"누군가가 나의 등잔의
심지에서 불을 붙여가도
내 등잔의 불은 여전히
빛나고 있습니다."

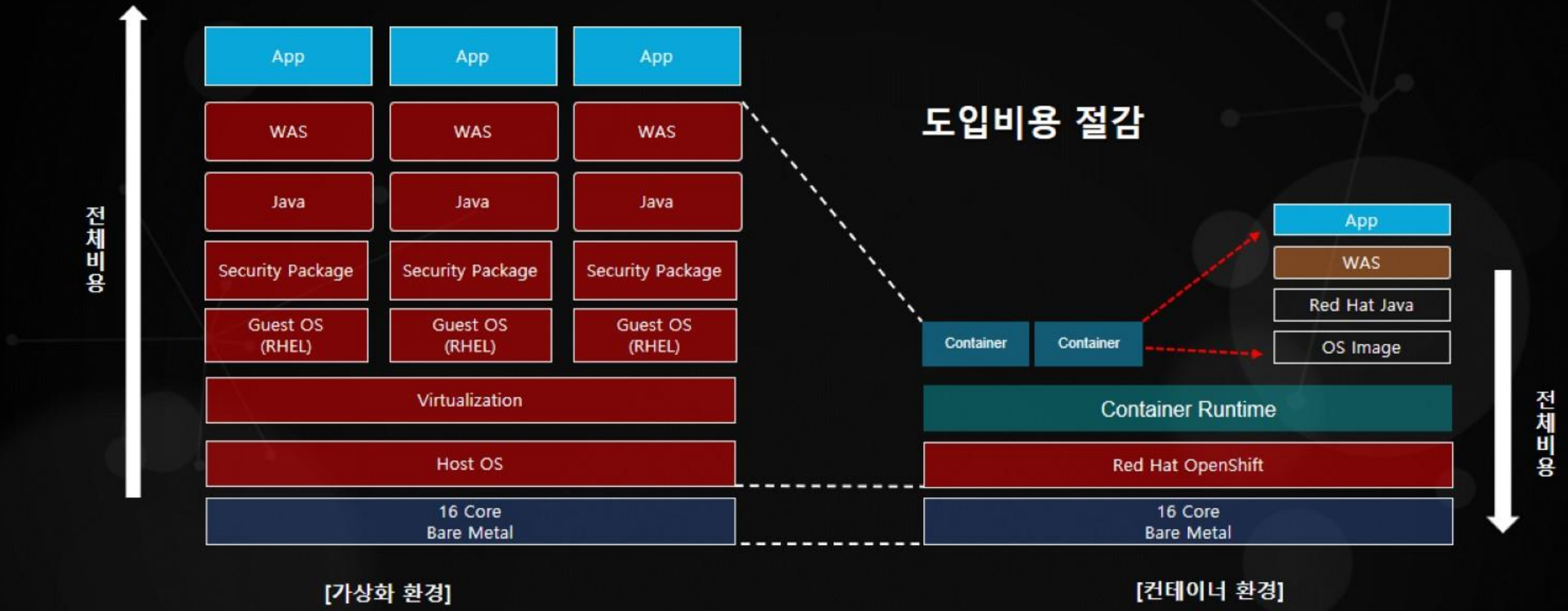
미국의 정치가 토머스 제퍼슨

openmaru
APM


컨테이너 기술의 차이점
인프라 운영의 변화
openmaru
APM

가상화 VS 컨테이너 비교 – 비용적인 측면

- 가상화 대비 Guest OS 유지보수, 라이선스, 관리비용 제거
- 서버 접근제어를 비롯한 보안 솔루션 제거



아직도 WAS BMT 나 POC를 하시나요?

- WAS 의 가용성/확장성/성능/신뢰성 등의 기능은 플랫폼으로 이전
- 더 가볍고 더 빠르고, 자동화에 친화적인 WAS 로 전환

WAS 제품 BMT

ORACLE WEBLOGIC

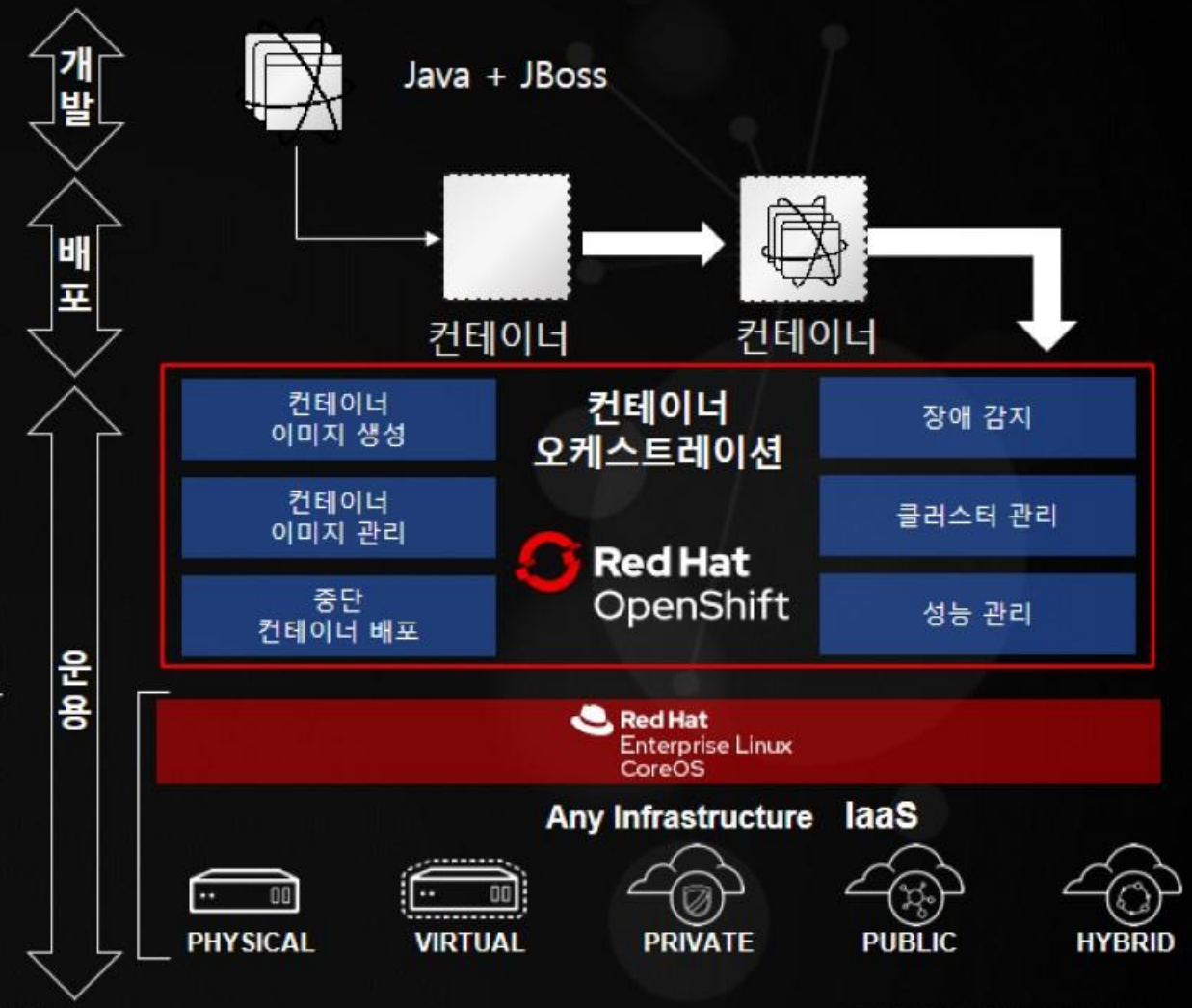
JBoss by Red Hat

WebSphere.

Tmax JEUS

RASP 에 의한 WAS평가

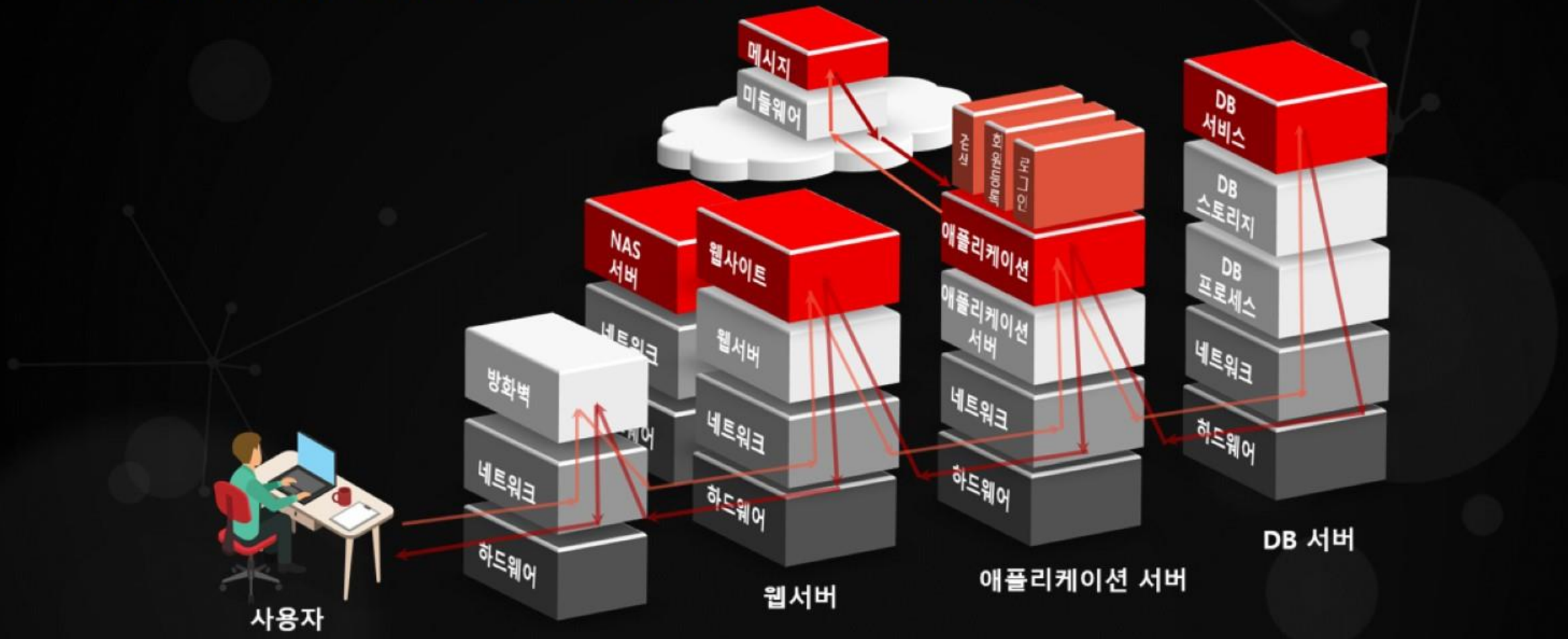
- 신뢰성 (Reliability):
  - 과부하 테스트
- 가용성 (Availability):
  - 일부 장애 발생시 전체 시스템 영향 최소화
- 확장성 (Scalability):
  - 자원 추가에 따른 선형적인 성능 개선
- 성능 (Performance):
  - TPS, 응답시간 등 평가
- 보안 기능 (Security)
  - (RBAC 등)
- WAS 도구 평가 (Manageability)
  - Admin Console



APM

미들웨어는 시스템 장애의 관문이자 시작점

- 데이터베이스가 50% 느려진다면 사용자 응답시간은 어떻게 될까요?
- DB 테이블 변경으로 SQL 에서 오류로 인하여 페이지가 오류가 난다면?



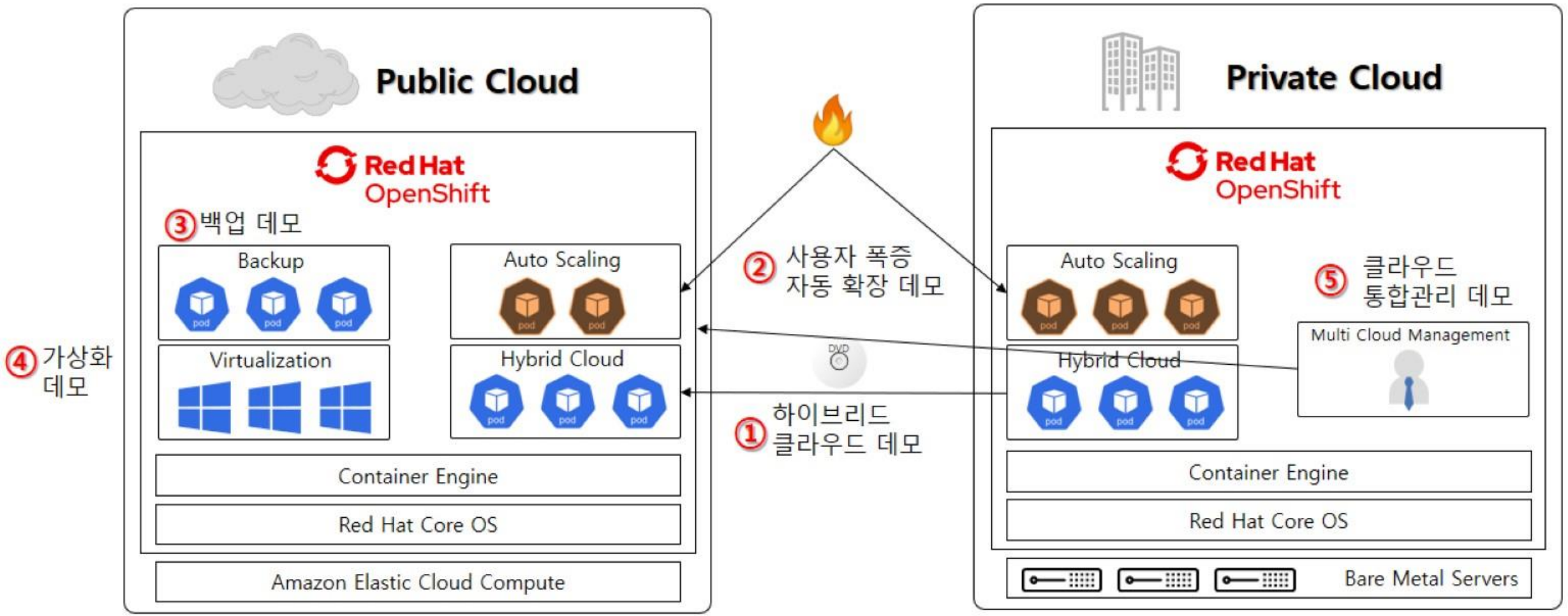

EBS 온라인 클래스
APM01
[긴급] 비정상 상황 발생 – 대시보드
[심각-CRITICAL] 'Worker Usage %' (평균값: 98.9)이 심각(CRITICAL) 임계값 '95'을 넘었습니다.
발생에이전트 : apache@EBS-OC-PROD2-WEB05[172.17.11.35]
클릭하여 상세한 정보를 확인하세요,... 10s.
[심각-CRITICAL] 'Worker Usage %' (평균값: 99.95)이 심각(CRITICAL) 임계값 '95'을 넘었습니다.
발생에이전트 : apache@EBS-OC-PROD2-WEB02[172.17.11.5]
클릭하여 상세한 정보를 확인하세요,... 10s.
Request Viewer
115098.0
3115
Request Velocity
Label
Ball
Satisfying
Tolerating
Frustrating
APDEX
APDEX 93.3 %
TPS 1071.8
Error Rate 0.1 %
Average Response Time 2,884 ms
TPS
Active Users
Transaction Heatmap (T-Map)
Avg. Res. Time
Error Rates
JVM Heap Usage
Active DB Connection
openmaru APM


정상 서비스 상황
WAS Dashboard
Request Viewer
2034.0
96
Request Velocity
Label
power
Satisfying
Tolerating
Frustrating
APDEX
Unacceptable
Info
99.6
%
1934.7
438.0
0.0
202
TPS
Active Users
Transaction Heatmap (T-Map)
Avg. Res. Time
Error Rates
JVM Heap Usage
Active DB Connection


Application Performance Management
Cloud Native
OPENMARU APM DEMO
openmaru
APM

오픈 하이브리드 클라우드 데모

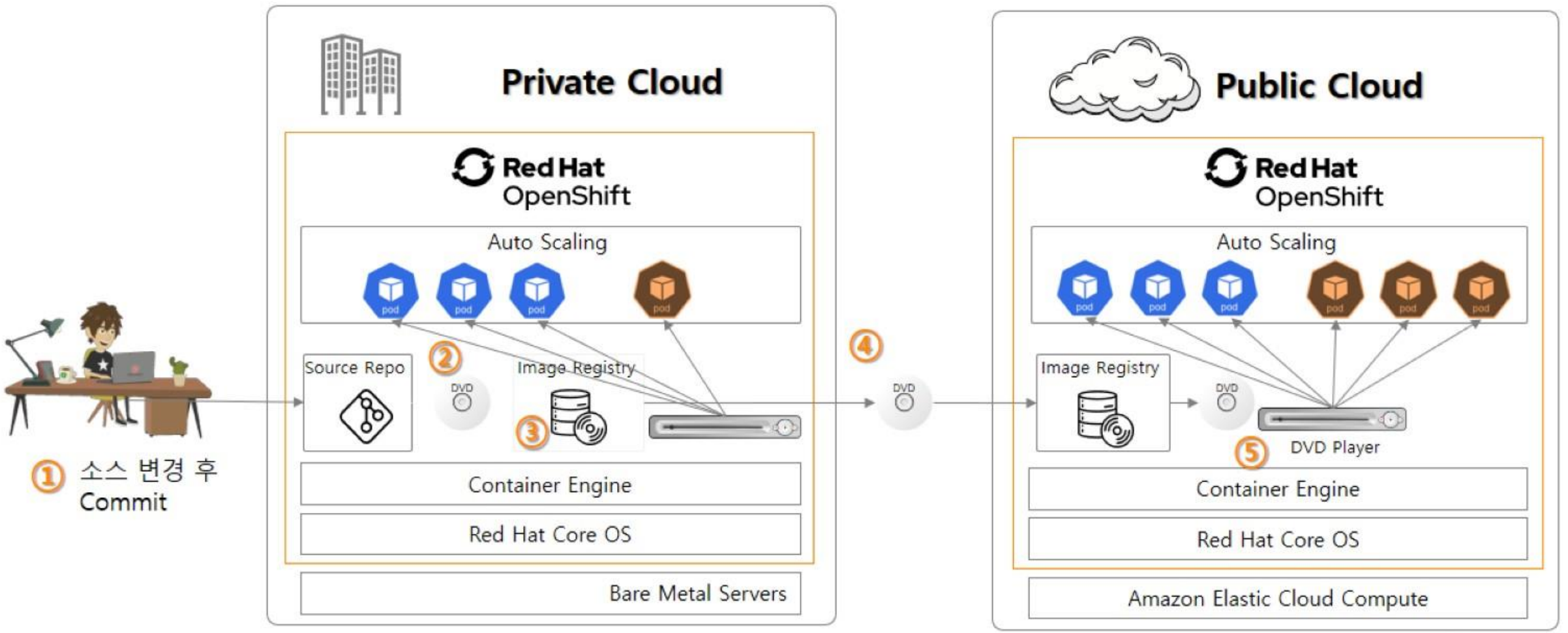
하이브리드 클라우드에서 같은 업무를 통합 운영, 관리하는 5가지 데모




openmaru
APM
조달청 디지털서비스몰 등록
오픈마루
디지털서비스몰에서오픈마루를 검색하세요.
Cloud Native 환경에서 애플리케이션 배포 데모
openmaru

개발은 내부 클라우드에서 운영은 외부 클라우드로

개발은 Private Cloud 에서 운영은 하이브리드로 Private/Public Cloud

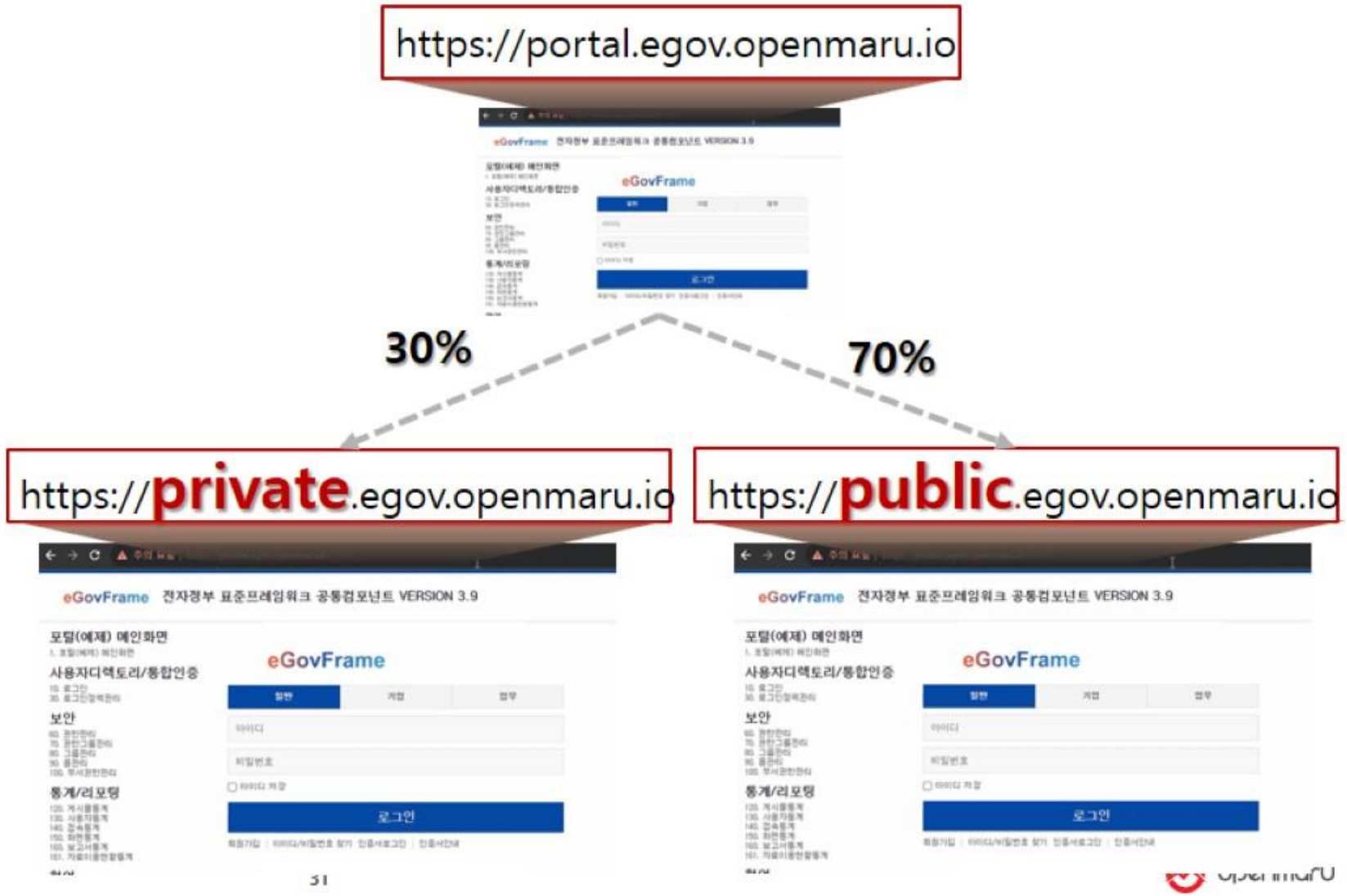


전자정부 F/W 포탈에 대한 글로벌 서버 부하 분산

GSLB (Global Server Load Balancing)을 통한 업무 부하 분산 데모

☑	portal.egov.openmaru.io	가중치 기반
☐	portal.egov.openmaru.io	가중치 기반
☐	private.egov.openmaru.io	단순
☐	public.egov.openmaru.io	단순

전자정부 F/W 포탈 서비스를 내부와 외부 클라우드에서 동시 운영



하이브리드 클라우드 데모-개발환경 데모

Private Cloud

Public Cloud

① Source 변경

② Source Commit

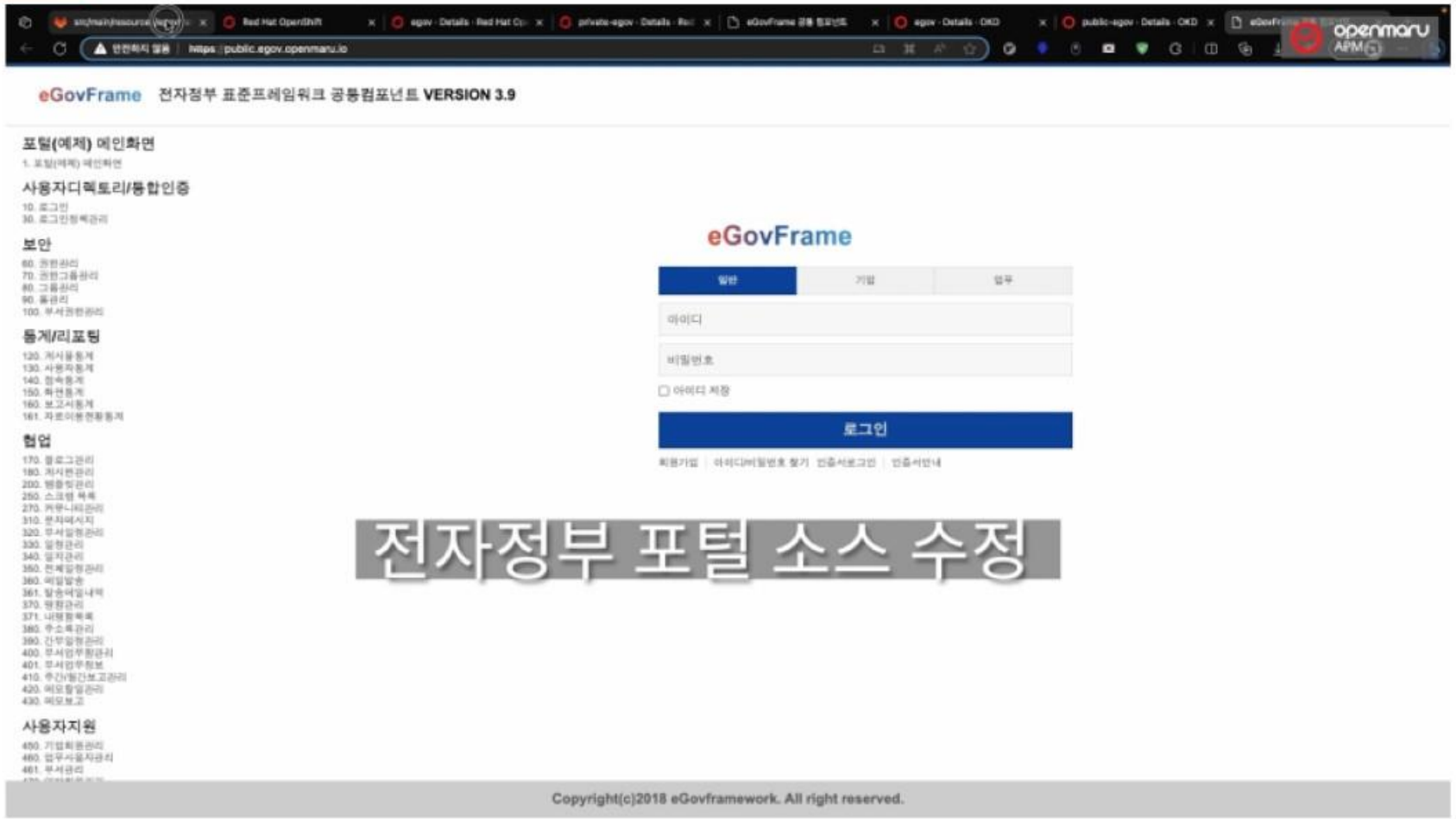
③ Source To Image Build

④ Private Cloud Image Push

⑤ Private Cloud Upgrade

⑥ Public Cloud Image Push

⑦ Public Cloud Upgrade



openmaru

조달청 디지털서비스몰 등록

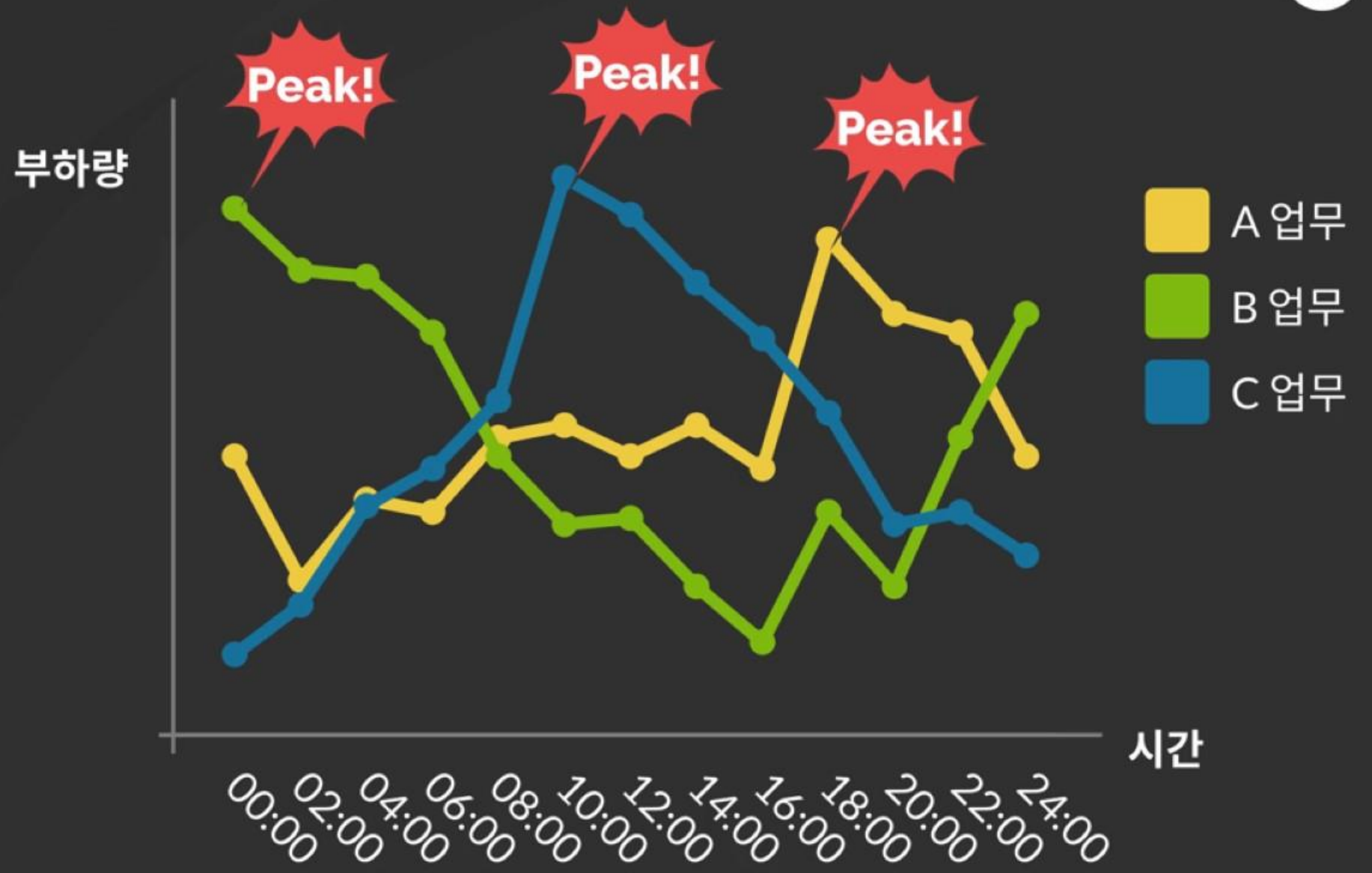
오픈마루

디지털서비스몰에서 **오픈마루** 를 검색하세요.

통계청 차세대 나라통계
멀티 애플리케이션에서 자동 확장

openmaru

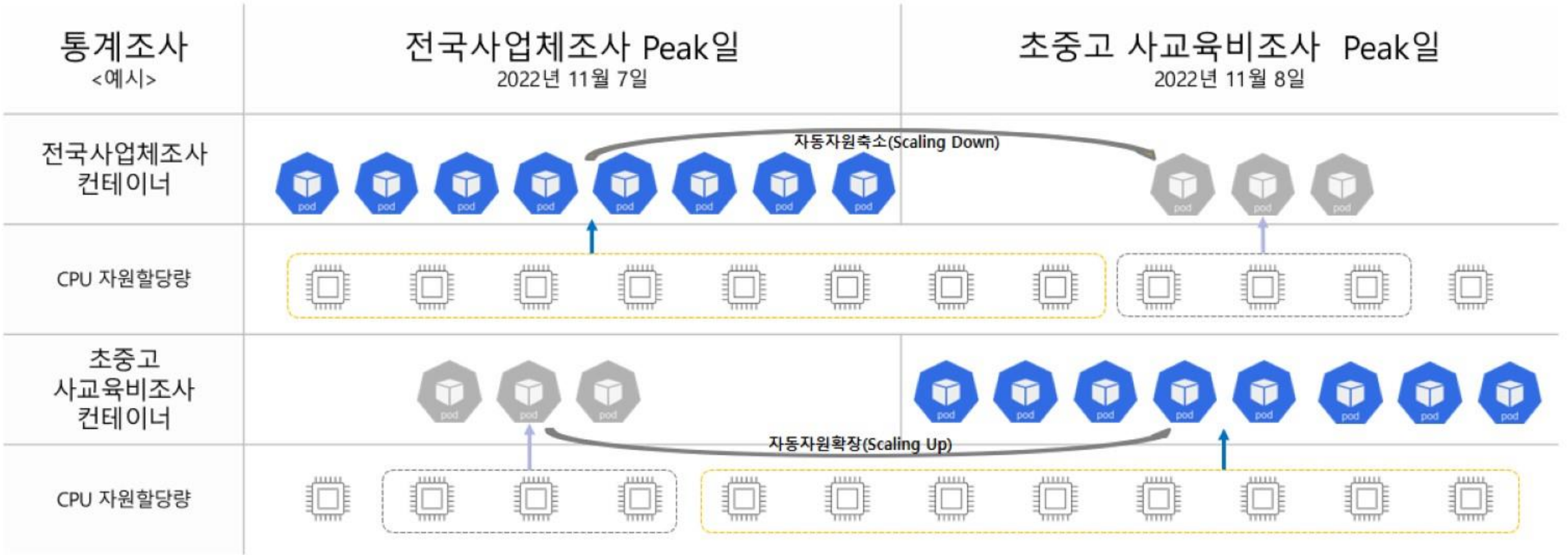
openmaru


Peak!
Peak!
Peak!
부하량
A 업무
B 업무
C 업무
시간
00:00
02:00
04:00
06:00
08:00
10:00
12:00
14:00
16:00
18:00
20:00
22:00
24:00

차세대 나라통계 - Peak 시점 별 자동 자원할당 활용 예시

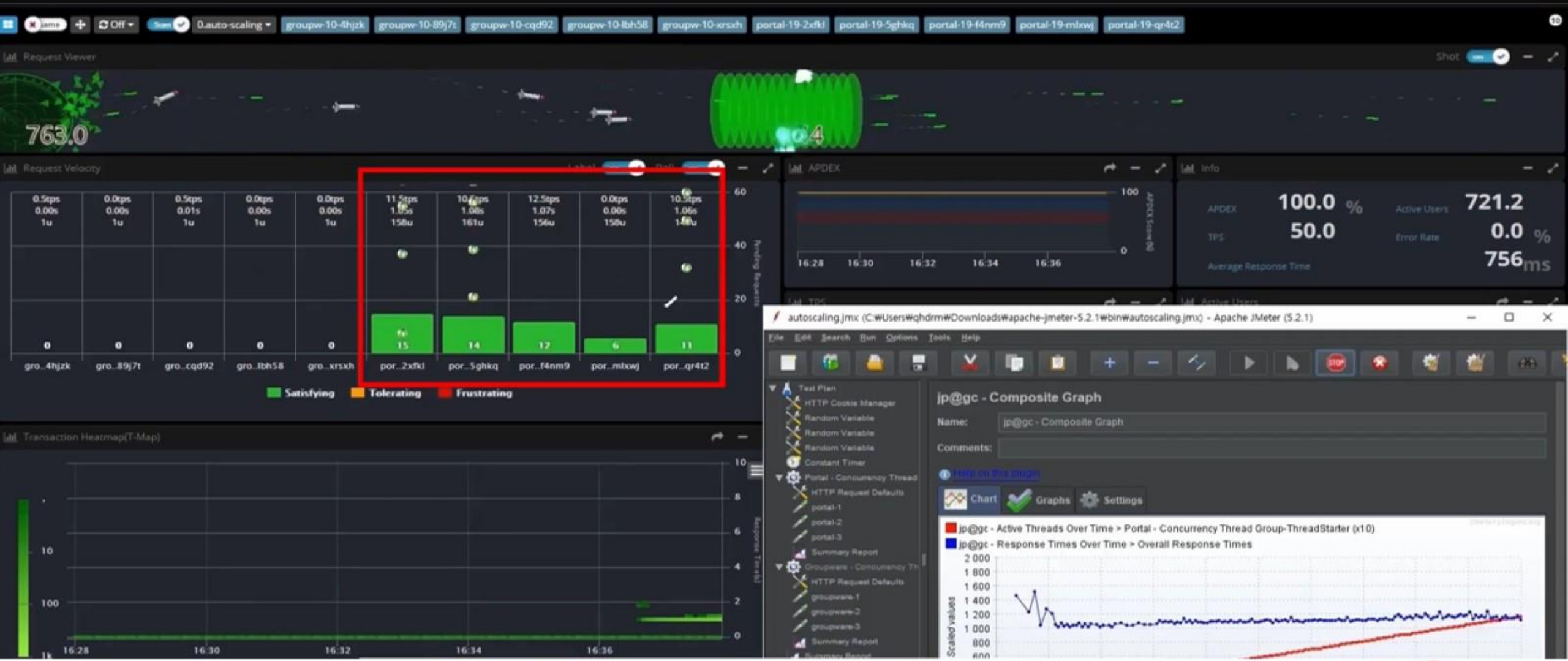
관리자의 개입없이 시스템 사용량에 따라 자동으로 확장/축소

- 한정된 서버자원을 Peak 시점에 따라 효율적으로 배분하여 사용 가능
- Peak 시점이 다른 통계조사에 대해 사람의 개입없이 자동자원할당이 가능



T-01 : 인스턴스 개수 한정된 환경(기존 나라통계)

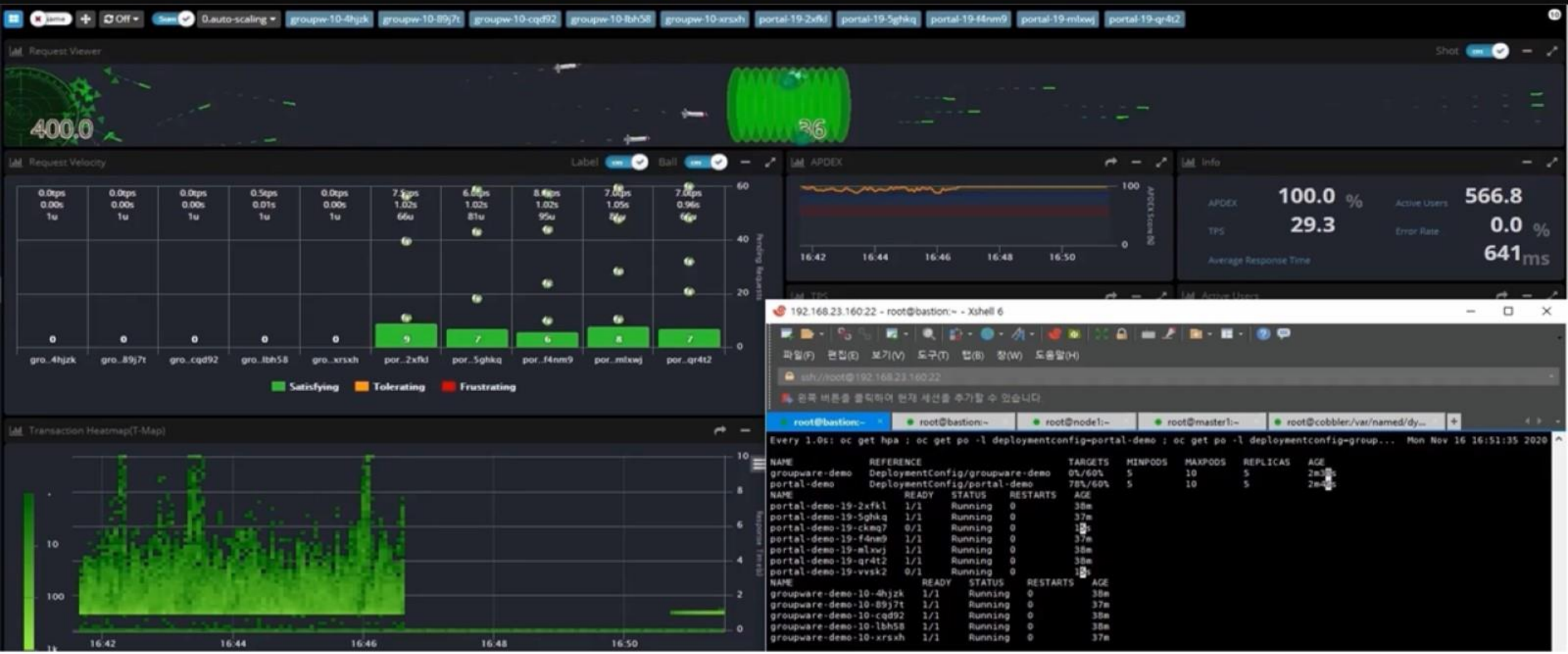
부하에 따른 응답시간 지연



openmaru

T-02 : 전국사업체조사 Peak일 때 자동부하분산 환경 부하테스트

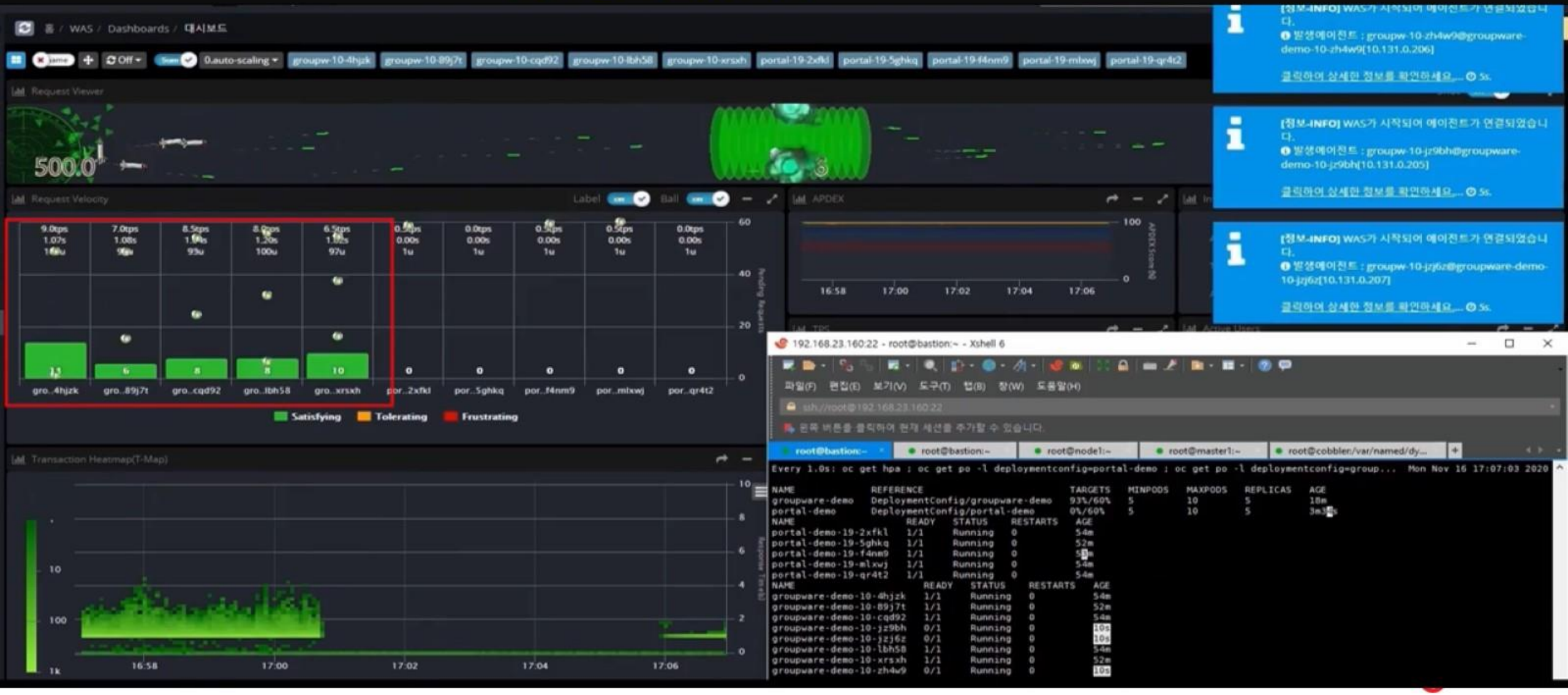
부하에 따른 컨테이너 자동확장으로 **응답시간 보장과 TPS 증가**



openmaru

T-03 : 초중고 사교육비조사 Peak일 때 자동부하분산 환경 부하테스트

부하에 따른 컨테이너 자동확장으로 **응답시간 보장과 TPS 증가**



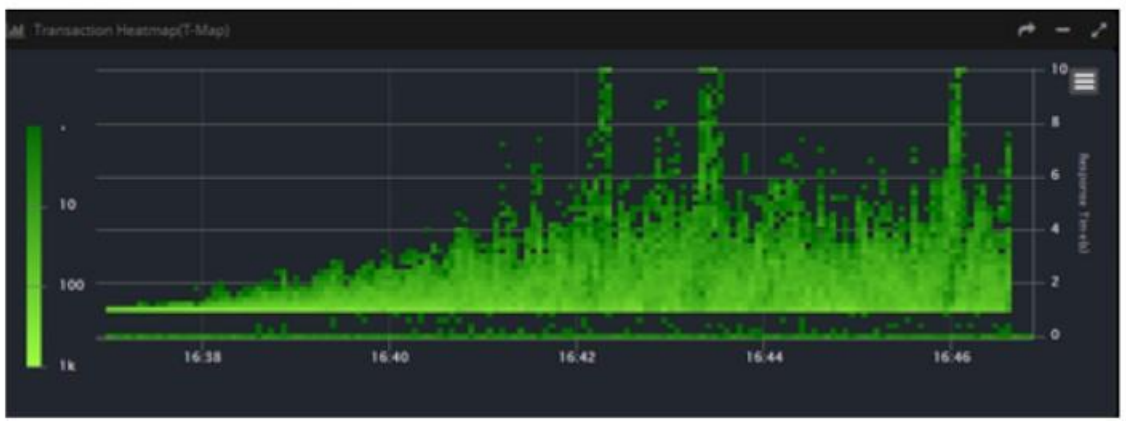
DEMO - 부하 테스트 결과 비교

Peak 시점이 다른 통계조사에 대해 별도 개입없이 자동자원할당이 가능함을 확인

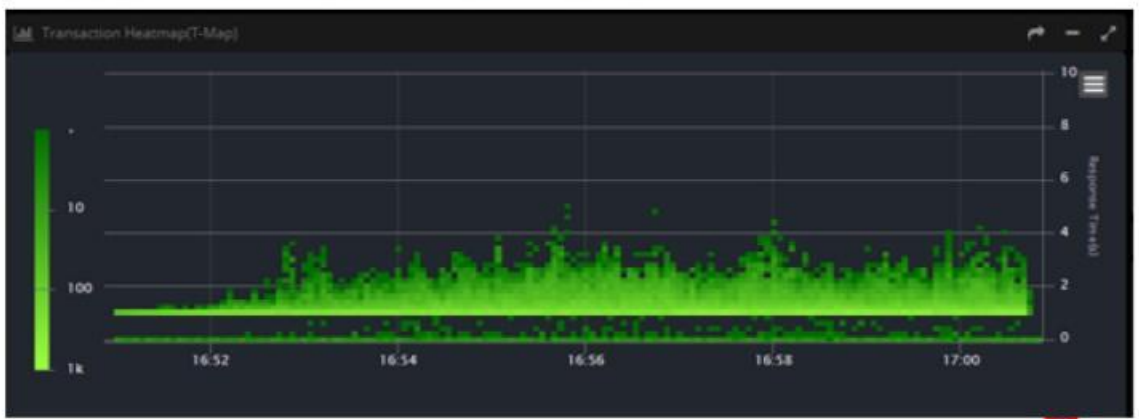
- **기존환경과 비교하여 자동자원할당이 되는 환경이 1.7 배 많은 양을 처리하며, 평균응답시간이 2.5 배 빠름**

테스트 케이스	테스트 내용	시뮬레이션 테스트 환경	처리량	**처리량 비교**	평균응답시간	**응답시간비교**	최소응답시간	최대응답시간	TPS
T-01	자동자원할당이 안 되는 환경에서 부하테스트	기존 나라통계 환경	**53,582**	100%	**2,374**	100%	1,010	19,979	**88**
T-02	Peak 시점이 다른 자동 자원할당이 되는 환경에서 부하테스트	전국사업체 조사 Peak 일 때 환경	**93,869**	169%	**919**	258%	10	4,589	**155**
T-0 3	Peak 시점이 다른 자동자원할당이 되는 환경에서 부하테스트	초중고 사교육비 조사 Peak일 때 환경	**91,394**	164%	**971**	244%	10	5,134	**151**

T-01 자동자원할당 안되는 기존 환경 응답시간분포



T-02/03 자동자원할당이 되는 환경 응답시간분포

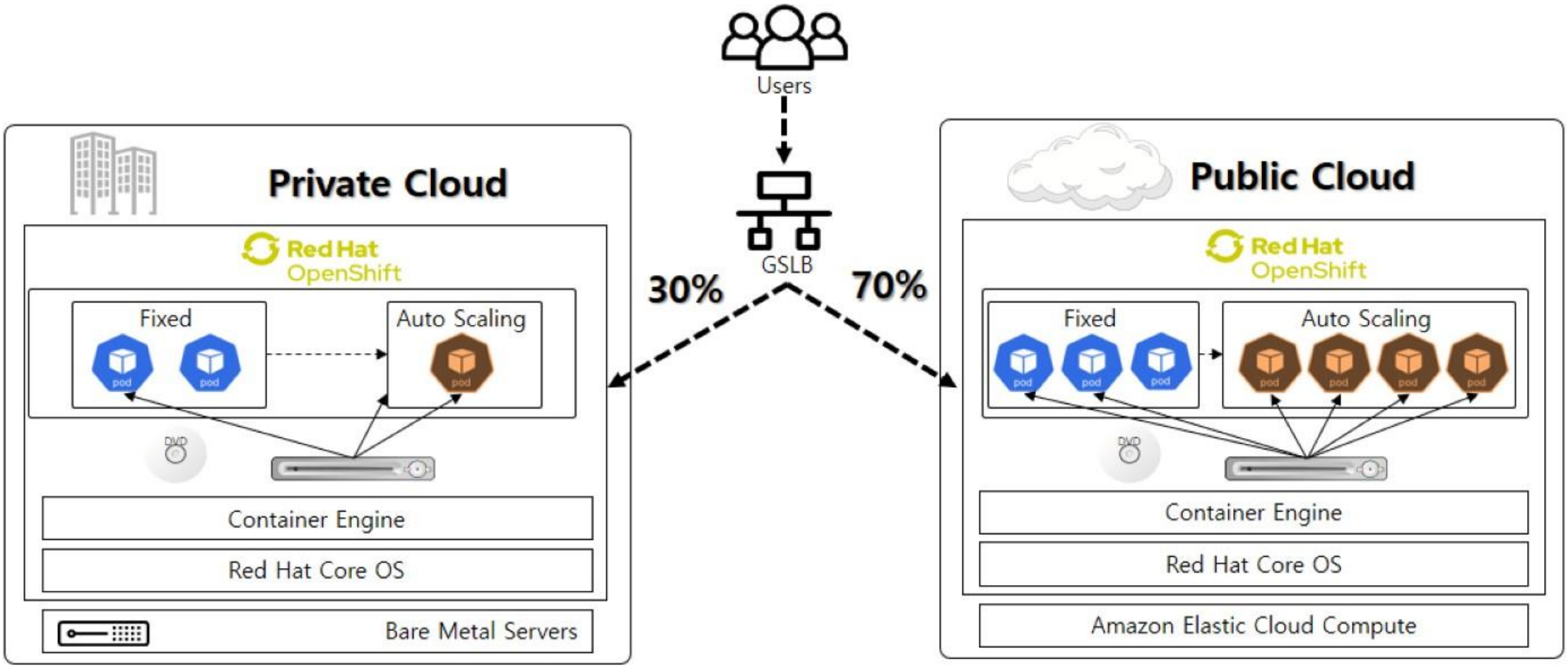



openmaru
APM
조달청 디지털서비스몰 등록
오픈마루
디지털서비스몰에서오픈마루를 검색하세요.
하이브리드 클라우드 데모
– 사용자 증가 자동 확장
openmaru

하이브리드 클라우드 데모 – 사용자 증가 자동 확장

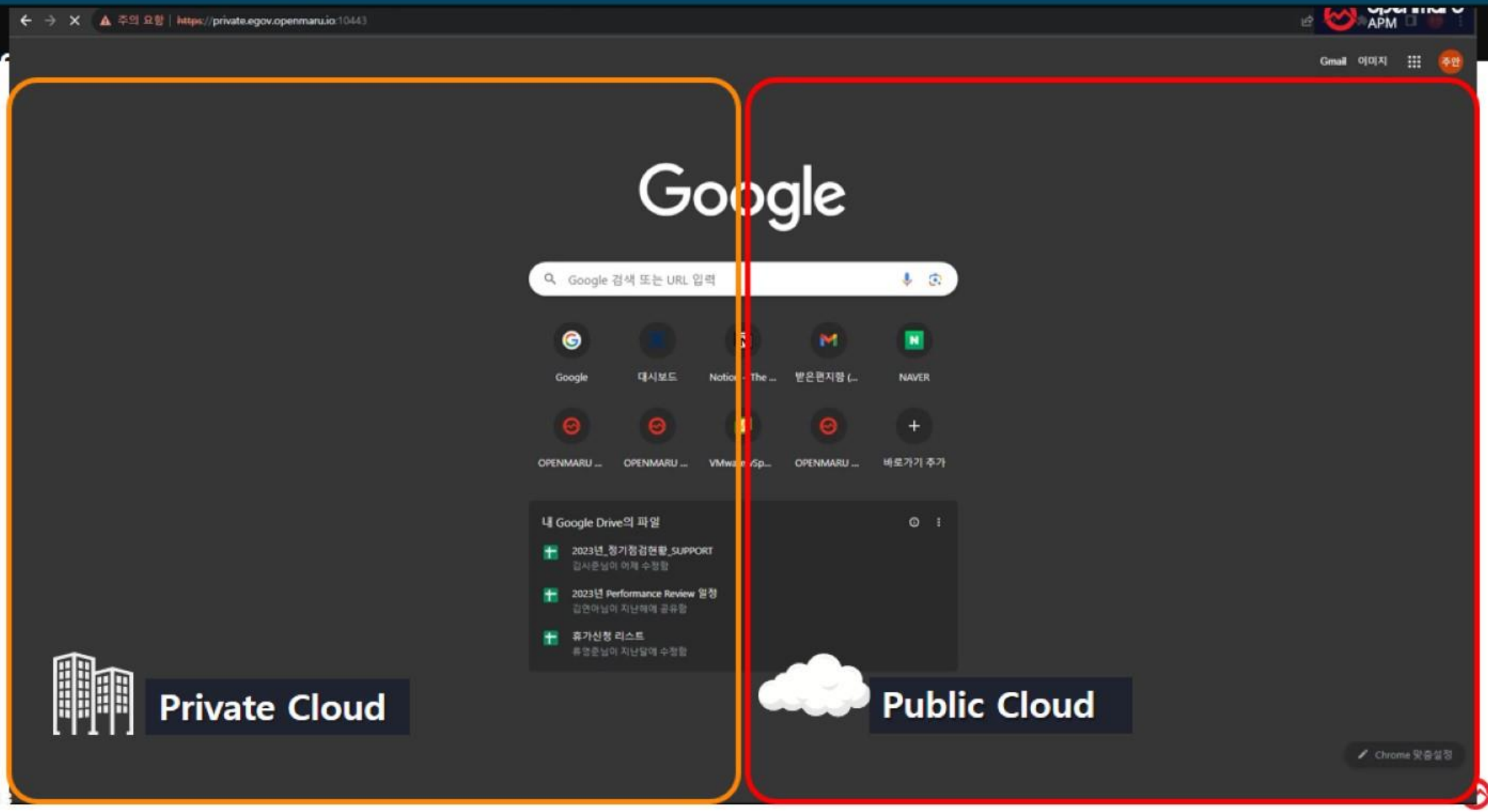
하이브리드 클라우드로 운영되는 포탈을 접속하는데 자동확장이 필요한 사용자 폭주상태를 가정

- 내부 클라우드와 외부 클라우드에 모두 동일한 홈페이지 서비스를 하고 프라이빗 30% vs. 퍼블릿 70% 로 운영 중



하이브리드 클라우드 데모 – 자동 확장 이전

하이브리드 클라우드 데모 – 자동 확장 이후

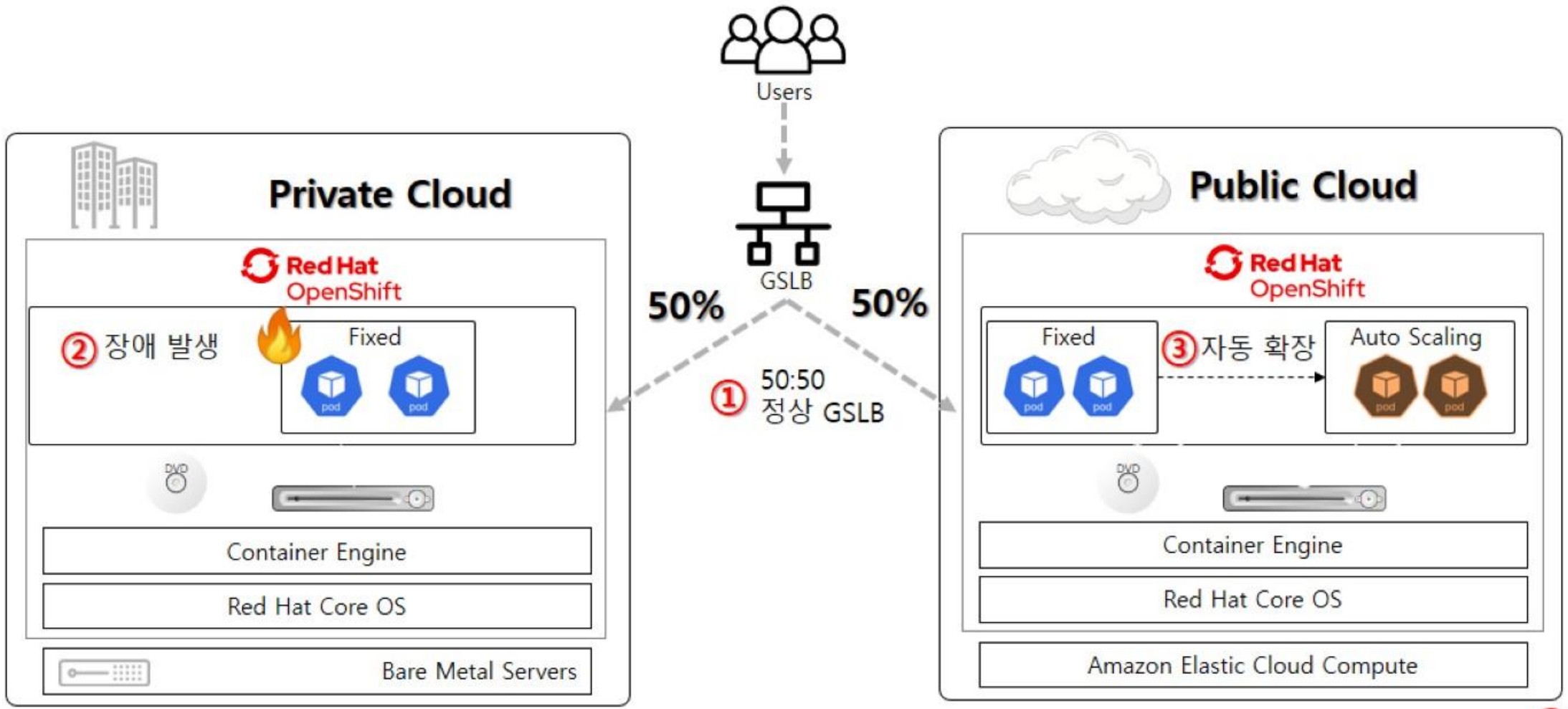



openmaru
APM
조달청 디지털서비스몰 등록
오픈마루
디지털서비스몰에서오픈마루를 검색하세요.
하이브리드 클라우드 데모
– 장애시 자동 확장 데모
openmaru

하이브리드 클라우드 데모 – Active Active DR

하이브리드 클라우드로 운영되는 포탈을 접속하는데 프라이빗 클라우드 장애 발생을 가정

- 내부 클라우드와 외부 클라우드에 모두 동일한 홈페이지 서비스를 하고 프라이빗 50% vs. 퍼블릿 50% 로 운영 중



하이브리드 클라우드 데모 – 장애 발생 이전 Active-Active GSLB

하이브리드 클라우드 데모 – Private Cloud 서비스 장애 발생

하이브리드 클라우드 데모 – Public Cloud 에서 부하에 따른 자동확장

조달청 디지털서비스몰 등록

오픈마루

디지털서비스몰에서**오픈마루**를 검색하세요.

오픈마루 회사 소개

openmaru

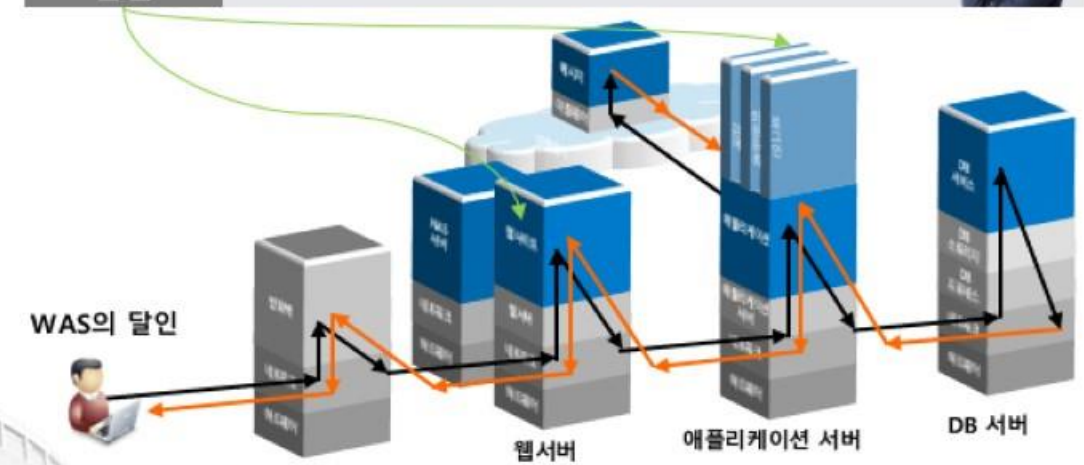
오픈소스와 WAS 전문가 집단 - 애플리케이션 중심 기술 지원

오픈소스 이지만 안전성, 보안, 성능은 WAS 달인을 통해서 해결

시스템 장애는 미들웨어에서부터 시작

WEB/ WAS는 장애의 관문

- ☑ DB가 느리면 WEB/WAS에 응답 시간이 지연
- ☑ 파일서버가 느려도 WEB/WAS 응답 시간 지연
- ☑ 연계서버가 느려도 WEB/WAS 응답시간 지연



의사소통 능력 + 인프라 경험

- ☑ PaaS 구축 / 운영에 대한 경험
- ☑ 데이터베이스 연결과 사용시 이슈에 대한 경험
- ☑ 연계솔루션과 네트워크 관련 이슈에 대한 경험
- ☑ 파일서버 등 Storage 관련 이슈에 대한 경험

▫ 레드햇 PaaS 와 WAS 만 공급과 설치하는 오픈마루 달인들

오픈마루 인력 구성

▫ 본 사업에 납품되는 버전
WAS 온라인 북

※ 저자 : 오픈마루 임직원들

애플리케이션 성능관리 전문 기업

어렵고 힘든 클라우드 사업, WEB/WAS 부분은 오픈마루가 책임지고 수행했습니다.

오픈마루는 레드햇 스페셜 파트너

Red Hat OpenShift | Red Hat JBoss Enterprise Application Platform — PaaS와 WAS 고객

온-나라 클라우드	채널계, 계정계	클라우드 플랫폼
올크레딧	PaaS 시스템	PaaS 시스템

Red Hat JBoss Enterprise Application Platform — WAS 고객

핀테크	이커머스 사이트	Crew Manager
온라인 교육서비스	코로나 대비 온라인클래스	디지털교과서

Apache Tomcat 고객

ERP 시스템	상품권 서비스	이커머스 사이트
결제 서비스	모바일 서비스	학사행정,홈페이지

공개SW에 대한 우려

- 설치/구성
- 튜닝
- 보안
- 성능
- 안정성
- 고가용성
- 기술지원

고객

WAS에 대한 제품 개발 및 공급

- openmaru Installer — WAS에 대한 설치/구성, 튜닝솔루션
- openmaru APM — WAS에 대한 성능 튜닝, 안정화, 장애감지 솔루션
- openmaru Cluster — WAS에 대한 고가용성 보장 솔루션
- openmaru Carepack — WAS에 대한 설치/구성, 튜닝, 안정화 기술지원 및 유지보수 서비스

오픈마루 WAS 달인들

- 공개SW WEB/WAS에 기술지원
- WEB/WAS 에 대한 설치, 구성, 튜닝, 오픈지원, 안정화, 유지보수 업무 담당

애플리케이션 마이그레이션 전문 - U2L / U2V / U2C

오픈 소스 WAS 인 WAS 로 전환하더라도 성능, 안정성, 보안에 책임 질 수 있어야 합니다.

U2L/U2C 전환 사업의 핵심은?

단순한 **사업 규모**를 고려하면 **중요도와 난이도**를 고려하면

WAS 설치/구성/튜닝 시 95% 단축

- 전문가 수준 구축/튜닝
- 설치보고서 제공
- 대규모 적합
- 웹취약점 조치
- 휴먼에러 제거

| OPENMARU Installer 구축 사례

온라인 개학	대상	서버 수
2020년 4월 9일	중3, 고3	60 대
2020년 4월 16일	초4~6, 중1~2, 고1~2	240대
2020년 4월 20일	초1~3	300 대

- 3 주 만에 600대 서버 WAS 설치 튜닝 – EBS 온라인클래스
- 코로나로 인한 학습 공백 – 준비되지 않은 클라우드 환경 튜닝 및 안정화

강동경희대학교병원 (2020)

통합의료정보시스템 고도화 사업

Unix2Linux
- HCI 장비 기반의 가상화
- JEUS 에서 WAS 교체 , 제니퍼에서 OPENMARU APM

건국대학교병원 (2017)

전산기기 교체 및 운영환경 고도화 사

Unix2Linux
- Bare Metal서버 기반
- JEUS 에서 WAS 교체 , 제니퍼에서 OPENMARU APM

Adidas Korea (2019)

E-Commerce/POS AWS 전환

Unix2Cloud
- Oracle Database To RDS
- WebLogic To WAS
- OPENMARU APM

예금보험공사(2015)

IT인프라최적화 사업 (2015)

Unix2Linux
- Oracle DB Upgrade
- WebSphere To WAS

포장이사 전문 - WAS 전환 핵심은 애플리케이션 마이그레이션

클라우드를 넘어 SaaS 서비스 까지 하는 APM 전문기술기업

설치형 APM 제품에서 부터 SaaS 클라우드 서비스까지 구축하고 운영중인 클라우드 전문 기술 기업입니다.

- 국내 최초로 OpenShift 와 Kubernetes 지원, 2021년에는 AWS Cloud 에 OPENMARU Cloud APM SaaS서비스를 오픈
- SaaS 서비스 구축은 클라우드 네이티브 기술과 보안, 멀티테넌시, 미터링, 데이터 관리 등 높은 수준의 기술들을 구현

Iaas,PaaS 부터 ▸▸▸ **SaaS 까지**

삼성SDS 클라우드에 OPENMARU APM 등록

삼성이 검증한 **APM** 제품

컨테이너 환경에 최적화된 **APM**

클라우드 적합한 가격 정책

openmaru APM

AWS Cloud 에서 SaaS 클라우드로 서비스 중인 OPENMARU Cloud APM

- 2021년 9월 정식 오픈
- 2022년 2월 현재 – 등록 사용자 :28명 / 모니터링 프로젝트: 76 개

완벽한 멀티테넌시 구현 | 3분 안에 모니터링 개시 | 가장 저렴한 가격 | 검증된 성능과 확장성 | 컨테이너 기반 미터링

홈페이지
https://www.openmaru.io

클라우드 콘솔
https://console.openmaru.io

OPENMARU APM
https://<user>.apm.a-apne2.openmaru.io/

Red Hat OpenShift

aws

openmaru

다양한 운영 환경에서 검증된 APM 제품

성능과 장애 관리 도구인 APM은 물리/가상서버, 클라우드 등 다양한 WAS운영 환경에서 검증되어야 합니다.

- OPENMARU APM은 국내 최초로 PaaS 와 Kubernetes 에서 WAS와 클라우드 인프라 모니터링을 제공하였습니다.
- 중앙부처와 지방자치단체를 비롯한 금융권과 주요 국내 기업에서 OPENMARU APM 을 사용하고 있습니다.

OPENMARU APM & PaaS 9개 이상

openmaru APM + Red Hat OpenShift

KLID 한국지식정보기술원 / 행정안전부 / 통계청
국가정보자원관리원 / SAMSUNG / KCB
KAI 한국항공우주산업(주) / 롯데카드 / DOOSAN

주요 지방자체단체 구축 사례 외 다수

서울특별시 / 인천광역시 / 세종특별자치시 / 충청북도
Jeju / 횡성군 / 안성
울산광역시 / 광주광역시 / 보령시 / Smart Asan
PAJU 파주시 / GUNPO 군포시 / 시민행복 / 포천시
남양주시 / 창원특례시 / 구로구

주요 기업 구축 사례외 다수 10개 이상

koscom / 한화생명 / Finnq
MEDIAWILL / DAEKYO 대교 / SPC
PN풍년 / Happymoney
Johnson&Johnson

주요 공공기관 구축 사례 외 다수 30개 이상

청와대 / 외교부 / 국방전산정보원
중앙선거관리위원회 / 법무부 / 방위사업청
EBS 한국교육방송공사 / KERIS 한국교육학술정보원 / 국립환경과학원

학교 / 병원 외 다수 5개 이상

강동경희대학교병원 / 건국대학교병원 / 한국기술교육대학교
한국기술교육대학교 직업능력심사평가원 / KRIVET 한국직업능력개발원 / SDU 서울디지털대학교

대규모 PaaS 운영환경에서 검증된 OPENMARU APM

국내 공공기관 및 대기업에 도입된 OPENMARU APM 주요 구축사례 입니다.

발주처	KLID 한국지역정보개발원	KAI 한국항공우주산업주식회사	KCB 코리아크레딧뷰로	LOTTE CARD
				
프로젝트 명	온-나라 클라우드 문서 2.0	스마트 플랫폼 구축	MSA 기반 개인신용정보	롯데카드 채널계 클라우드 구축
당면과제	▪ 1차 사업 - 성능 및 안정화 이슈 ▪ 2차 사업 - 오픈마루가 성능/장애 해결	▪ 한국항공우주산업의 글로벌 경쟁력확보에 필요한 스마트플랫폼 구축	▪ 개인신용정보서비스와 업신용정보서비스 같은 핵심적인 서비스 운용 중 ▪ MSA 기반으로 핵심 비즈니스 서비스 구축	▪ 금융권 최초로 채널계 시스템 도입으로 대고객 접점 서비스 ▪ 국내최초로 OPENMARU APM으로 안정화 지원 및 성능 튜닝
오픈마루 역할	PaaS/WAS 성능관리 사업자	PaaS BMT 수행/구축/유지보수	PaaS 구축 및 안정화 지원	APM 공급 및 안정화 지원

openmaru APM

▪ PaaS 문제 발생!! 소방관
▪ WAS 문제 발생!! 해결사

Red Hat JBoss Enterprise Application Platform

Red Hat OpenShift

코리아 크레딧 뷰로(KCB) - 클라우드 네이티브 전환

코리아 크레딧 뷰로(KCB) MSA 기반 개인신용정보, 기업신용정보, 마이데이터 서비스 - PaaS & OPENMARU APM

프로젝트 명	올크레딧 개인신용정보 서비스
얼마나 중요한 업무인가?	신용관리, 명의보호, 금융관리를 통합하여 관리하는 국가대표기관 시스템
규모는 어떻게 되는가?	개인신용정보서비스와 기업신용정보서비스 같은 핵심적인 서비스 운용중
오픈마루 역할은?	오픈소스 WAS 마이그레이션 컨설팅, 클라우드 네이티브 환경 도입 컨설팅

☑ **OPENNARU**와 함께 **PaaS** 환경을 도입/구축한 서비스

카드본인확인서비스
상용 WAS를 오픈 소스로 전환

개인신용정보서비스
PaaS 도입

기업CB 구축사업
MSA 기반으로 재개발

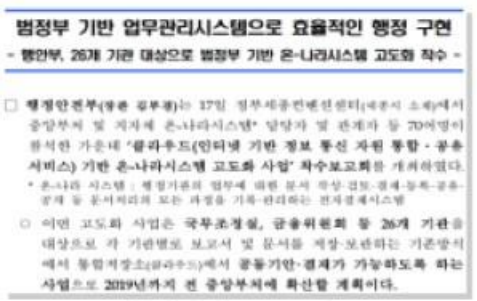
마이데이터 신규 구축 사업
API G/W 도입

행정안전부 - 국가 80 여개 중앙부처 그룹웨어(온-나라) 시스템

한국지역정보개발원 온-나라 클라우드 문서 2.0 사업(2018 년) - Red Hat PaaS & OPENMARU APM

프로젝트 명	온-나라 클라우드 문서 2.0
얼마나 중요한 업무인가?	**정부부처 20군데와 지방자치단체 약 80군데** 등 100만 공무원이 사용하는 문서 결재 시스템
규모는 어떻게 되는가?	**Wokrer Node 80대** 규모
오픈마루 역할은?	**PaaS 전문인력 상주**와 **OPENMARU APM으로 성능관리를 담당**하여 안정화 지원

☑ **OPENMARU APM**은 국내 최초, 최다 컨테이너 **&** 클라우드 환경을 지원


범정부 기반 업무관리시스템으로 효율적인 행정 구현

- 행안부, 26개 기관 대상으로 범정부 기반 온-나라시스템 고도화 착수 -

행정안전부 보도 자료
범정부 기반 온-나라 시스템 고도화 착수

클라우드 전환 시 노후 장비 교체 대비
2.6배의 비용 효과 발행

클라우드 온-나라 문서 2.0
개념도

온나라 클라우드 APM
성능 테스트

한국교육방송공사 - 코로나 대응을 위한 온라인 클래스 시스템

EBS 온라인 클래스에 도입되어 **동시접속자 130만명** , VM 600 대 , **WAS 인스턴스 1200** 여 개 모니터링

프로젝트 명	EBS 온라인 클래스
얼마나 중요한 업무인가?	코로나로 인하여 초,중,고 300백만명의 학생이 접속하는 온라인 클래스 구축 업무
규모는 어떻게 되는가?	WEB/WAS 600대 / 인스턴스1,200개 규모
오픈마루 역할은?	1주일 내 600대 머신에 1,200개 인스턴스 구축 및 APM으로 오픈지원, 성능 튜닝 및 안정화 지원

☑ **OPENMARU APM**으로 안정화를 실현한 실제 **EBS** 방송 화면

교육 차관에게 OPENMARU APM으로 보고하는 화면

600여대의 서버의 총 1,228개 WAS 모니터링

실제 인스턴스 1,200개 그룹대시보드 모니터링 화면 -2

실시간 Active User를 확인할 수 있는 모니터링 화면


openmaru
APM
조달청 디지털서비스몰 등록
오픈마루
디지털서비스몰에서오픈마루를 검색하세요.
OPENMARU APM with Cloud Native
openmaru

클라우드 전환에 따른 APM 요구사항의 변화

클라우드에서는 APM 모니터링 대상이 WAS에서 컨테이너로 변경

- 컨테이너 단위로 WAS에 대한 확장/축소, 장애 복구 , 업그레이드, 패치 작업하여 WAS와 함께 컨테이너를 모니터링해야 합니다.
- 기존 물리서버나 가상서버와는 달리 컨테이너는 휘발성으로 상태를 가지고 있지 않습니다.

새롭게 추가된 APM 요건

- WAS는 물론 컨테이너 모니터링
- 자동장애복구/자동확장축소 모니터링 정보 제공
- 폐기된 컨테이너 정보 제공
- OpenShift 관리 정보와의 통합

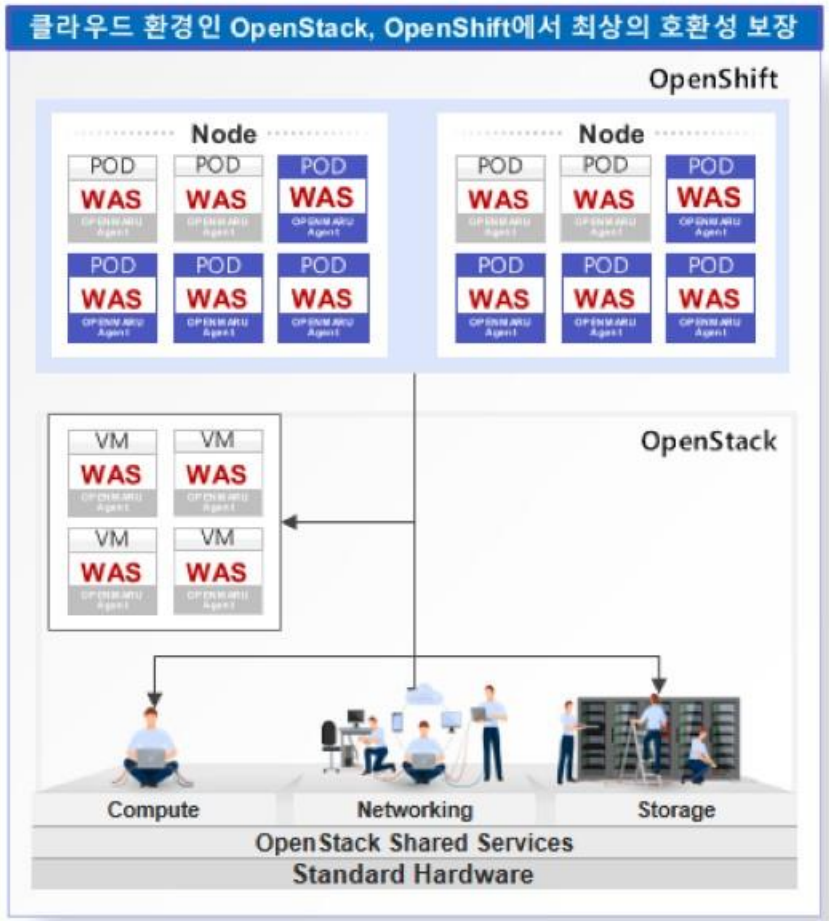
운영 요구 사항	사람의 수작업에 의한 운영		
WAS증설과 축소	운영자	수작업	직접확인
WAS 장애	운영자	수작업	직접확인
애플케이션 배포	운영자	수작업	직접확인
WAS 운영관리	운영자	수작업	직접확인
WAS 업그레이드	운영자	수작업	직접확인

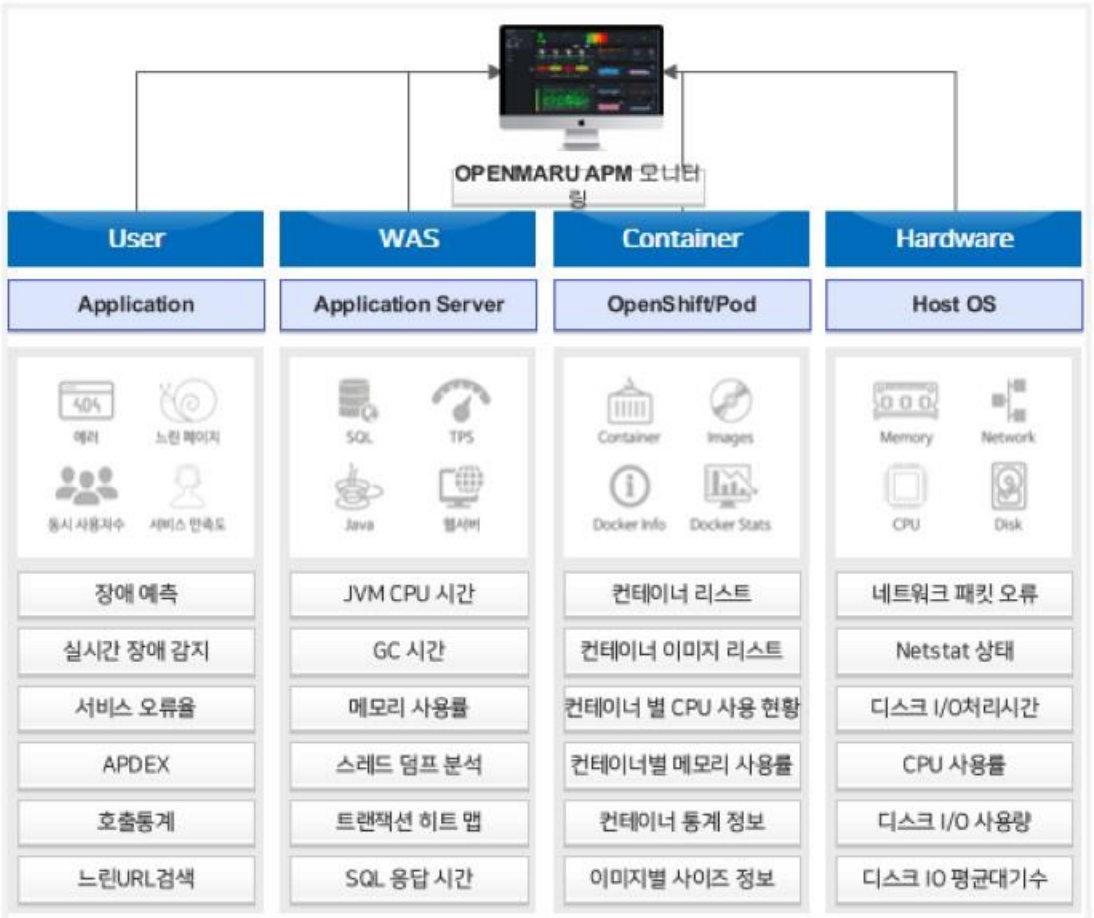
클라우드에서 규칙의 의한 운영 자동화		
클라우드	부하에 따른	자동 확장/축소
클라우드	자동감지	자동 장애복구
클라우드	무정지 서비스	자동 배포
클라우드	컨테이너로	WAS 관리
클라우드	이미지에 의한	WAS 업그레이드

클라우드 환경에 최적화된 APM

클라우드 운영환경, 개발,검증 환경에 최적화된 기능 제공

- 가상화(OpenStack 등), Kubernetes(OpenShift)에 최적화된 APM 입니다. 다수의 대규모 컨테이너 환경에 적용되어 적합성을 이미 검증 받았습니다.
- OPENMARU APM은 클라우드 환경(VM/Container)에 유리한 라이선스 정책을 적용하고 있습니다.

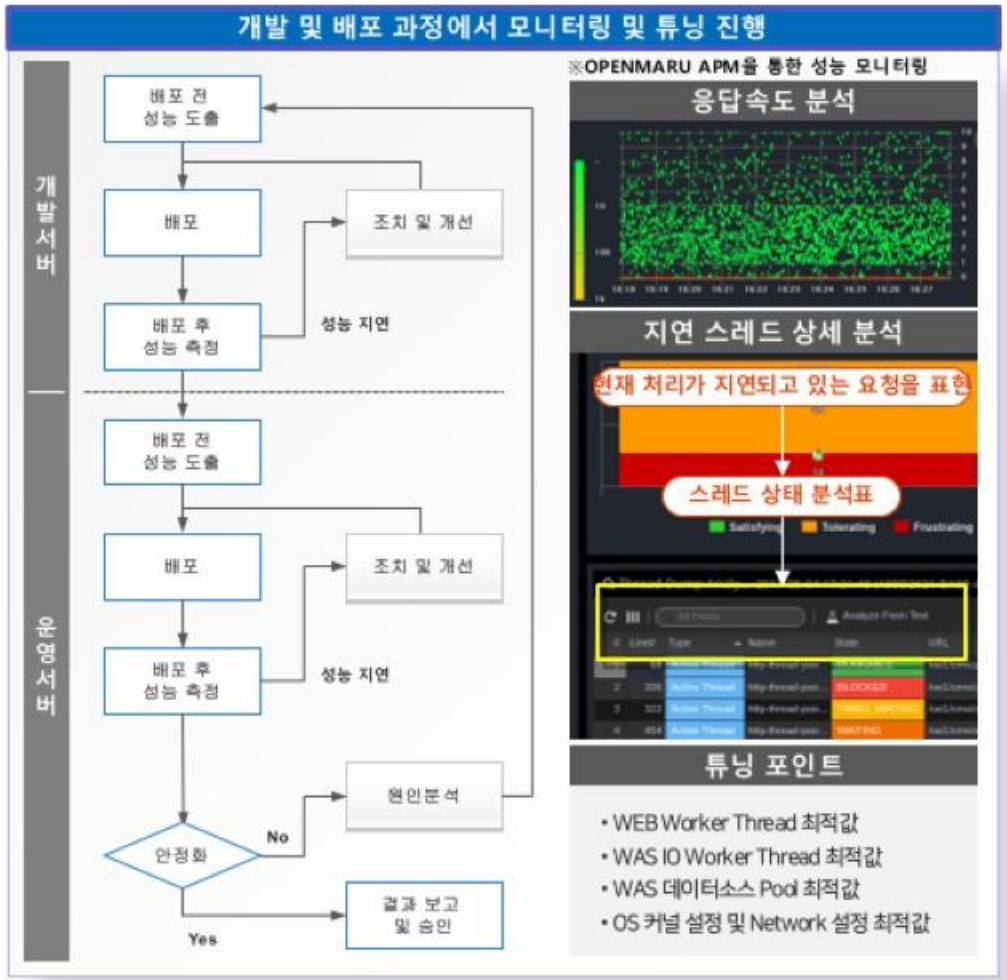


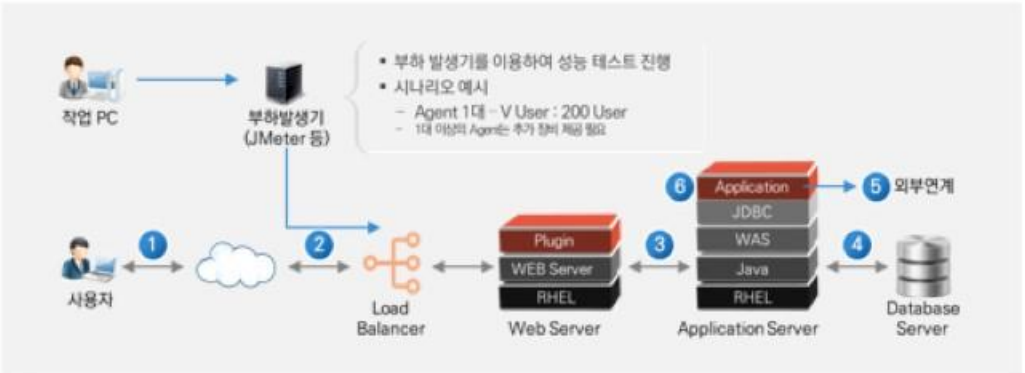


서비스 안정화 및 기술지원

APM은 서비스 개발 및 오픈이후 안정화까지 모니터링과 튜닝이 필요합니다.

● OPENMARU APM을 활용하여 개발 시점부터 오픈 시점까지 지속적인 모니터링 및 튜닝을 시행하여 시스템의 안정적 오픈과 운영을 지원합니다.





주요 구간별 확인 및 튜닝 포인트

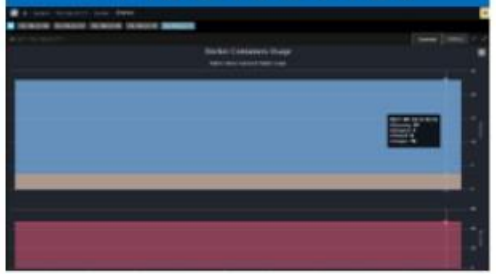
1	사용자 – 방화벽 및 L4(LB)	▪ 방화벽 ▪ 대역폭 확인
2	방화벽 및 L4(LB) – WEB	▪ 목적지 연결 설정 ▪ 로드 밸런싱 옵션 (Client IP Hash)
3	WEB – WAS	▪ MEM Worker Thread 튜닝 ▪ ModJk 연결 옵션 튜닝 ▪ ModJk Sticky 옵션
4	WAS – DBMS	▪ IO Worker Thread 튜닝 ▪ DB Connection Pool 튜닝 ▪ DB Statement Cache 튜닝 ▪ JVM Heap Memory 튜닝
5	WEB/WAS – NAS	▪ IOPS, 지연 시간 확인
6	WAS – 외부 인터페이스	▪ HTTP API Connection Pool 튜닝
7	애플리케이션	▪ 오류 확인 ▪ 비즈니스 로직 성능 확인

컨테이너 기반 모니터링 기능

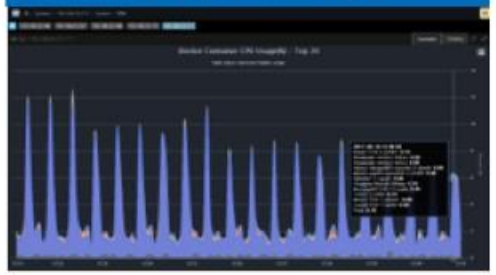
컨테이너(쿠버네티스) 기반 모니터링 기능 제공

- 컨테이너에서 작동하는 WAS의 모니터링 뿐만 아니라 컨테이너의 CPU, 메모리, 네트워크 트래픽/패킷 오류 를 확인 할 수 있습니다.
- 동작하는 컨테이너의 모니터링 뿐만 아니라 컨테이너 이미지의 메타 상세 정보도 확인 할 수 있습니다.

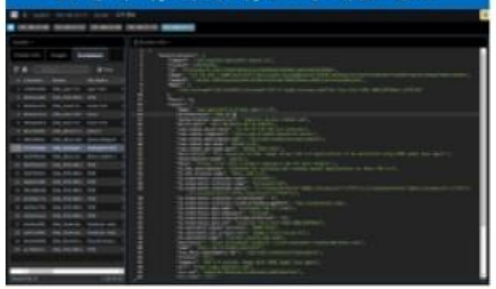
모니터링 항목별 임계치 설정



컨테이너의 CPU 사용률



컨테이너, 이미지, 도커 데몬 정보



컨테이너의 메모리 사용률

컨테이너 패킷 오류 정보

컨테이너 네트워크 트래픽 정보

컨테이너(PaaS) 지원 APM 과 기존 APM 기능 비교

구 분	세부 항목	OPENMARU APM (컨테이너 지원 APM)	기존 APM 제품
기존 시스템	Java	• 모니터링 지원	모니터링 지원
	WAS	• 모니터링 지원	모니터링 지원
클라우드 시스템	PaaS 환경	• Docker Daemon 과 관련 이미지 정보 제공 • POD 에 대한 리소스 정보 제공 (CPU/Memory/Disk/Network)	모니터링 지원 불가
	오토스케일링	• 오토스케일링 관련 리소스 정보 제공 • WAS 상태 정보 제공	모니터링 일부 지원
	PaaS 에서 Java 정보	• POD 상에서 실행되는 Java 가상 머신 정보 제공 (Heap , Java 상태 정보 등)	모니터링 일부 지원
	PaaS 에서 WAS 정보	• POD 상에서 실행된 WAS 에 대한 정보	모니터링 일부 지원
	POD 상태 정보 제공	• 폐기된 POD 에 대한 검색 지원 • 과거 POD 에 대한 정보 제공	모니터링 일부 지원

컨테이너 환경 에이전트 관리방안

컨테이너 환경에서 WAS 인스턴스 관리 기능 제공

- 오토 스케일 인/아웃 상황에 따라 에이전트가 자동으로 추가되고, 자동으로 숨겨집니다.
- 컨테이너 중지로 인해 삭제된 인스턴스의 모니터링 정보는 삭제되지 않고 숨겨져 있으므로 필요 시 다시 분석 가능합니다.

Auto Scaling으로 증가된 인스턴스 자동 추가 및 모니터링

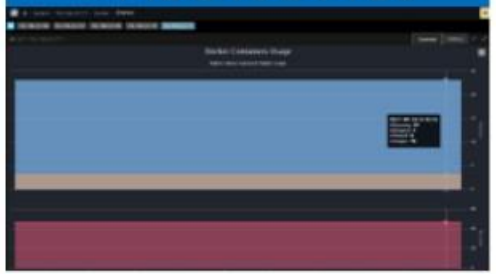
Auto Scaling으로 제거된 인스턴스 자동 삭제

컨테이너 기반 모니터링 기능

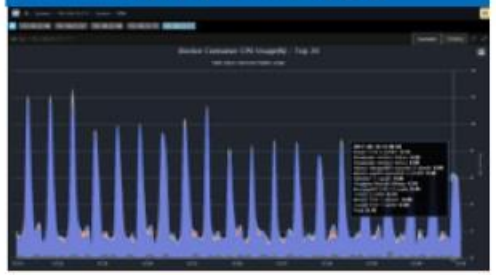
컨테이너(쿠버네티스) 기반 모니터링 기능 제공

- 컨테이너에서 작동하는 WAS의 모니터링 뿐만 아니라 컨테이너의 CPU, 메모리, 네트워크 트래픽/패킷 오류 를 확인 할 수 있습니다.
- 동작하는 컨테이너의 모니터링 뿐만 아니라 컨테이너 이미지의 메타 상세 정보도 확인 할 수 있습니다.

모니터링 항목별 임계치 설정



컨테이너의 CPU 사용률



컨테이너, 이미지, 도커 데몬 정보

컨테이너의 메모리 사용률

컨테이너 패킷 오류 정보

컨테이너 네트워크 트래픽 정보

컨테이너(PaaS) 지원 APM 과 기존 APM 기능 비교

구 분	세부 항목	OPENMARU APM (컨테이너 지원 APM)	기존 APM 제품
기존 시스템	Java	• 모니터링 지원	모니터링 지원
	WAS	• 모니터링 지원	모니터링 지원
클라우드 시스템	PaaS 환경	• Docker Daemon 과 관련 이미지 정보 제공 • POD 에 대한 리소스 정보 제공 (CPU/Memory/Disk/Network)	모니터링 지원 불가
	오토스케일링	• 오토스케일링 관련 리소스 정보 제공 • WAS 상태 정보 제공	모니터링 일부 지원
	PaaS 에서 Java 정보	• POD 상에서 실행되는 Java 가상 머신 정보 제공 (Heap , Java 상태 정보 등)	모니터링 일부 지원
	PaaS 에서 WAS 정보	• POD 상에서 실행된 WAS 에 대한 정보	모니터링 일부 지원
	POD 상태 정보 제공	• 폐기된 POD 에 대한 검색 지원 • 과거 POD 에 대한 정보 제공	모니터링 일부 지원

컨테이너 환경 에이전트 관리방안

컨테이너 환경에서 WAS 인스턴스 관리 기능 제공

- 오토 스케일 인/아웃 상황에 따라 에이전트가 자동으로 추가되고, 자동으로 숨겨집니다.
- 컨테이너 중지로 인해 삭제된 인스턴스의 모니터링 정보는 삭제되지 않고 숨겨져 있으므로 필요 시 다시 분석 가능합니다.

Auto Scaling으로 증가된 인스턴스 자동 추가 및 모니터링

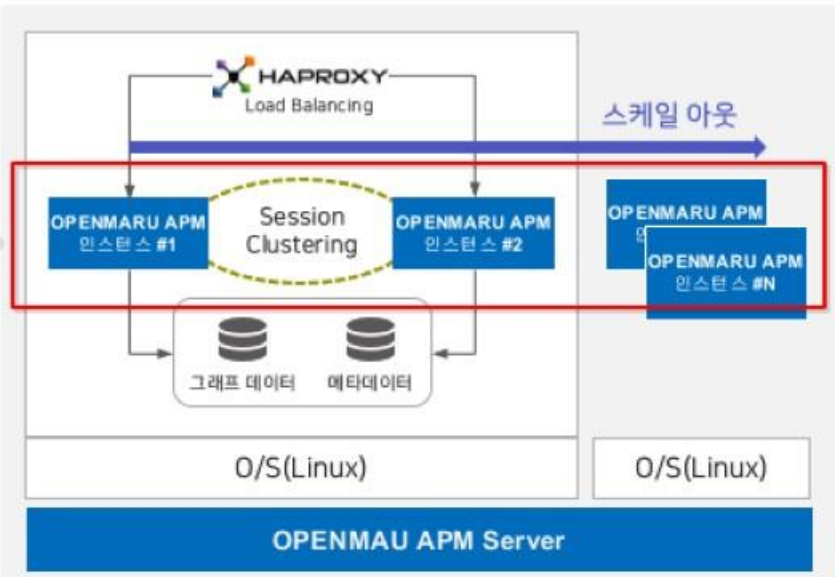
Auto Scaling으로 제거된 인스턴스 자동 삭제

확장가능한 수집서버 아키텍처

확장 가능한 아키텍처 지원 기능 제공

- 대용량 에이전트를 지원하기 위해 OPENMARU APM은 기본적으로 이중화로 구성되며, 필요한 경우 수집 인스턴스를 확장하여 부하를 분산할 수 있습니다.
- 이중화 된 OPENMARU APM 인스턴스는 HAProxy와 OPENMARU CLUSTER 세션 클러스터링을 통해 무 중단 서비스를 지원합니다.

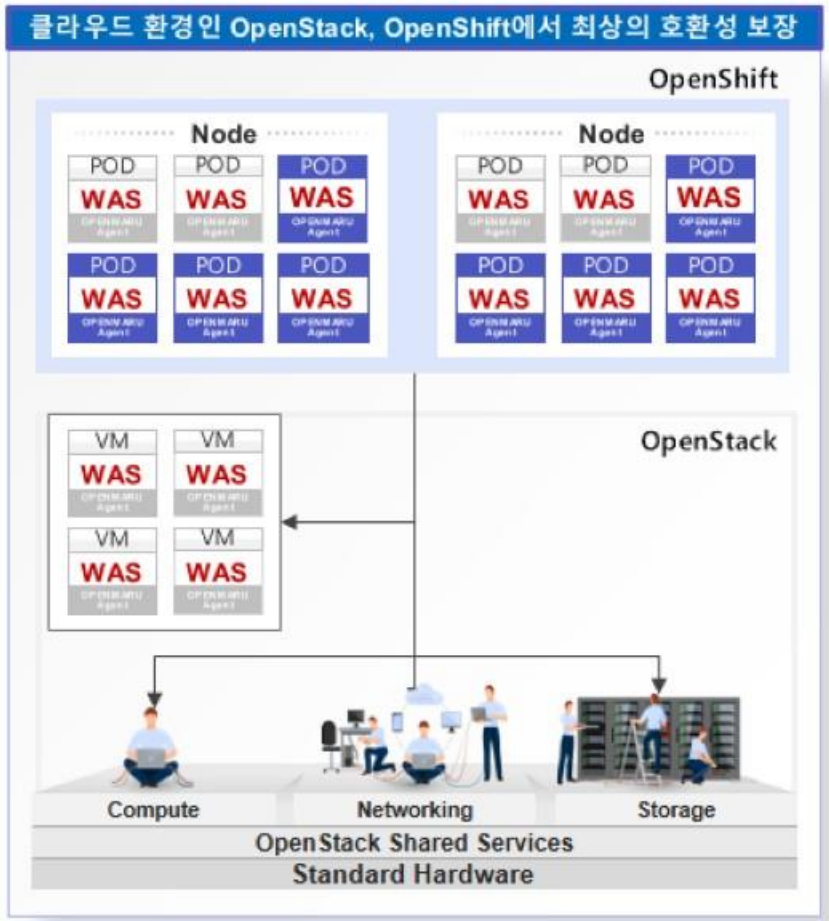
- ✓세션 유지를 위해 OPENMARU CLUSTER 세션 클러스터링 구성
- ✓자체적인 이중화를 통해서 한쪽 서버가 중단되더라도 서비스뿐만 아니라 모니터링도 정상적으로 유지
- ✓시스템 장애 발생 시 장애복구가 용이
- ✓패치 및 업그레이드 과정 시에도 365일 24시간 안정적인 무중단 서비스 작업을 위한 이중화 구조

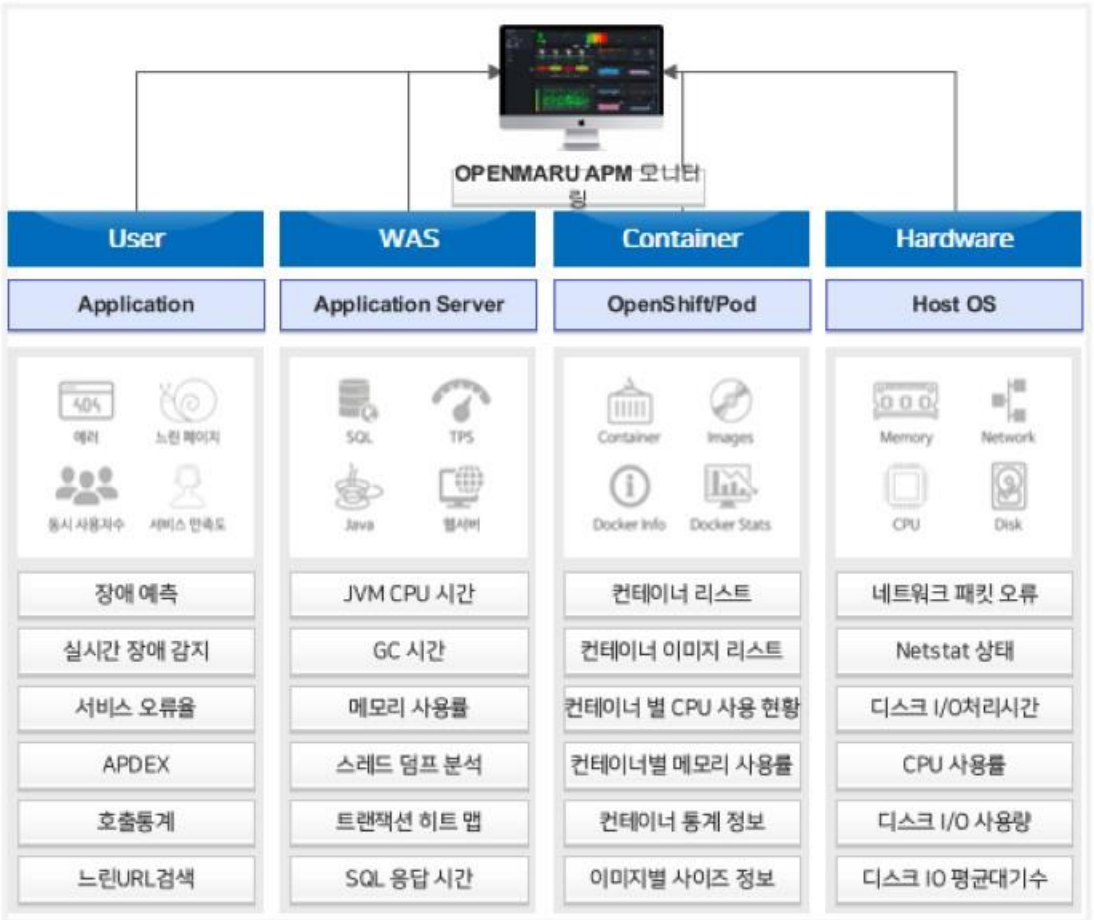


클라우드 환경에 최적화된 APM

클라우드 운영환경, 개발,검증 환경에 최적화된 기능 제공

- 가상화(OpenStack 등), Kubernetes(OpenShift)에 최적화된 APM 입니다. 다수의 대규모 컨테이너 환경에 적용되어 적합성을 이미 검증 받았습니다.
- OPENMARU APM은 클라우드 환경(VM/Container)에 유리한 라이선스 정책을 적용하고 있습니다.

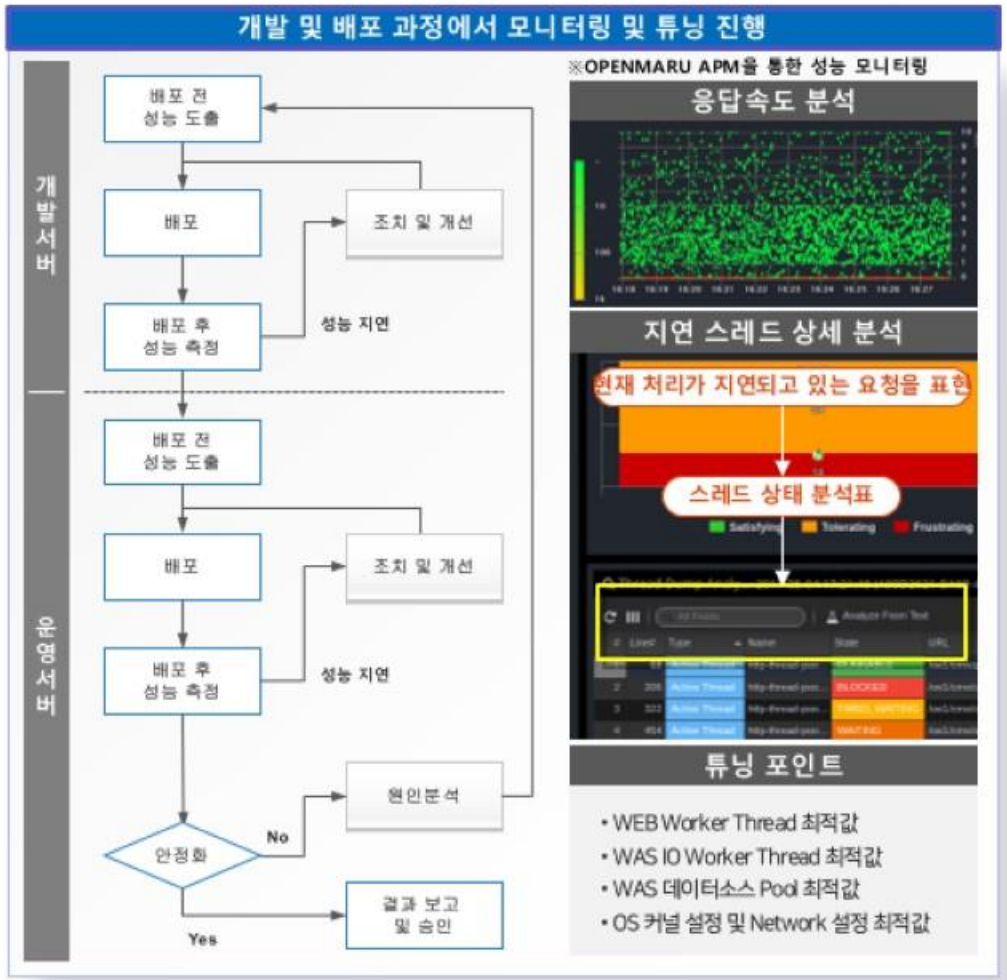


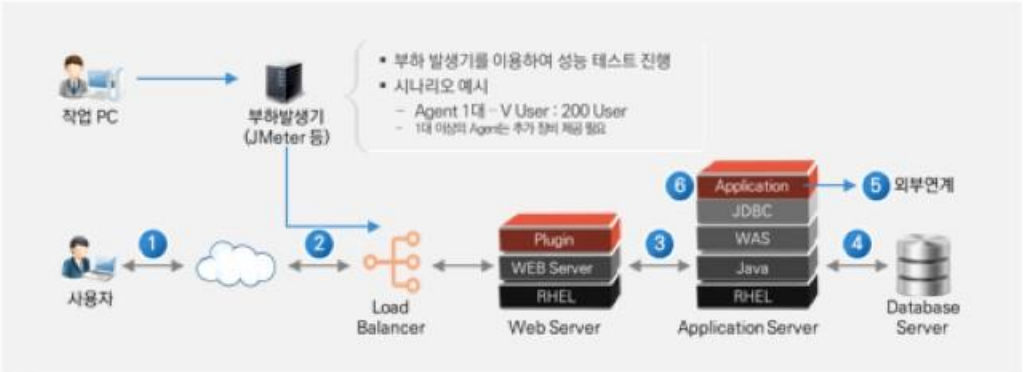


서비스 안정화 및 기술지원

APM은 서비스 개발 및 오픈이후 안정화까지 모니터링과 튜닝이 필요합니다.

● OPENMARU APM을 활용하여 개발 시점부터 오픈 시점까지 지속적인 모니터링 및 튜닝을 시행하여 시스템의 안정적 오픈과 운영을 지원합니다.





☑ 주요 구간별 확인 및 튜닝 포인트

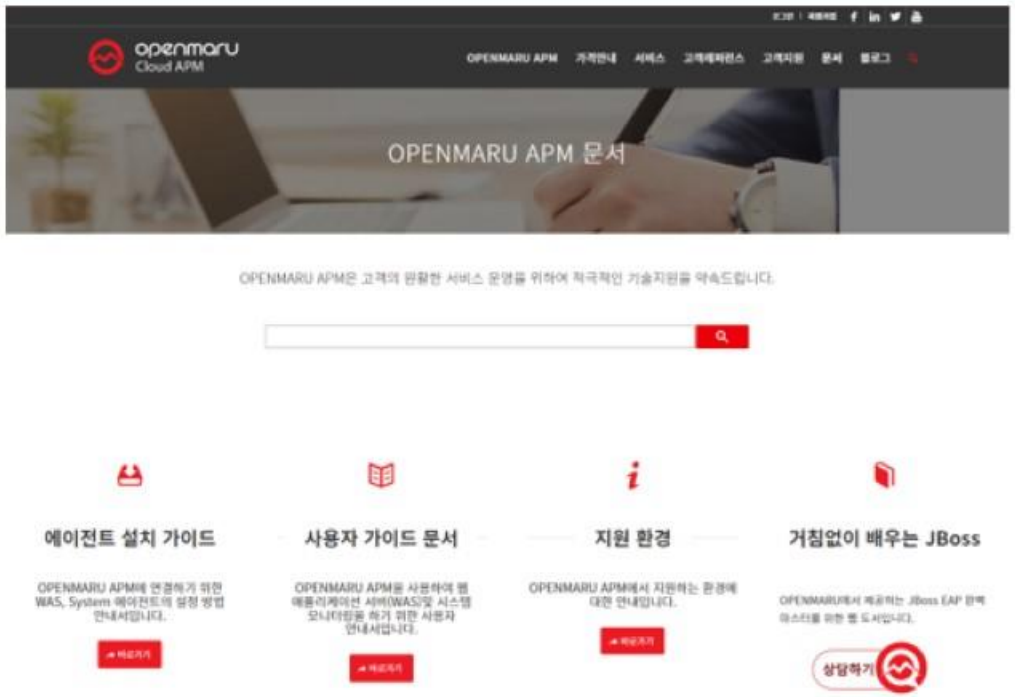
1	사용자 – 방화벽 및 L4(LB)	▪ 방화벽 ▪ 대역폭 확인
2	방화벽 및 L4(LB) – WEB	▪ 목적지 연결 설정 ▪ 로드 밸런싱 옵션 (Client IP Hash)
3	WEB – WAS	▪ MEM Worker Thread 튜닝 ▪ ModJk 연결 옵션 튜닝 ▪ ModJk Sticky 옵션
4	WAS – DBMS	▪ IO Worker Thread 튜닝 ▪ DB Connection Pool 튜닝 ▪ DB Statement Cache 튜닝 ▪ JVM Heap Memory 튜닝
5	WEB/WAS – NAS	▪ IOPS, 지연 시간 확인
6	WAS – 외부 인터페이스	▪ HTTP API Connection Pool 튜닝
7	애플리케이션	▪ 오류 확인 ▪ 비즈니스 로직 성능 확인

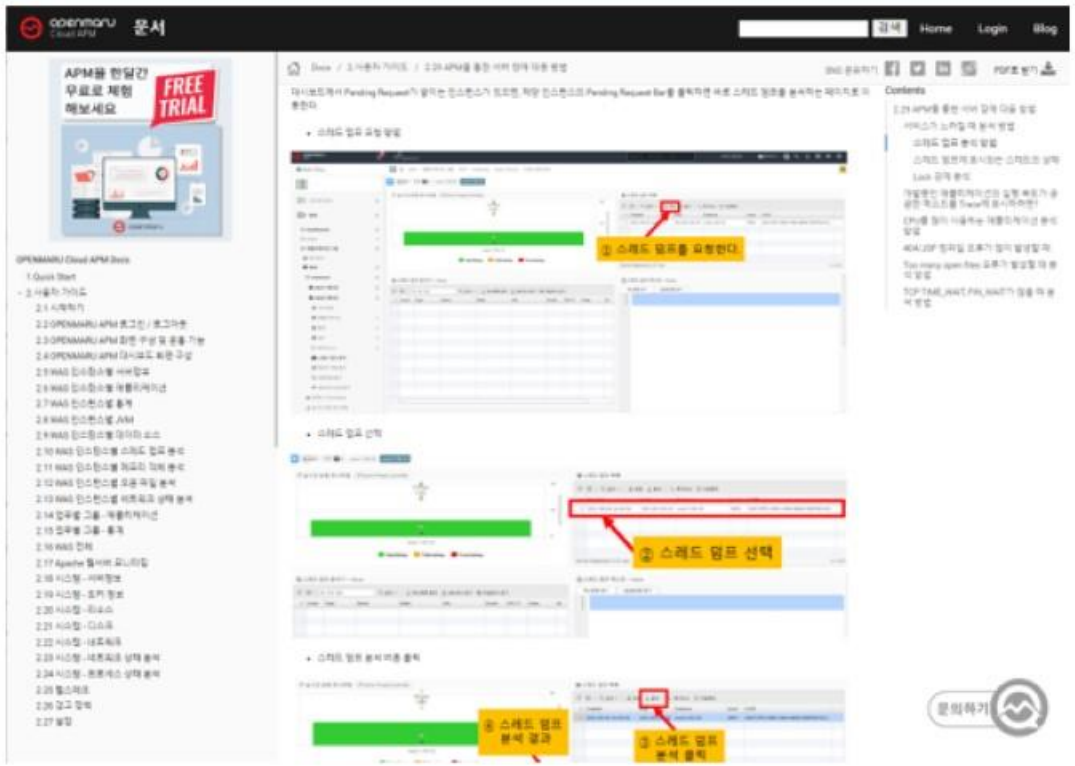
APM 사용자 온라인 매뉴얼

OPENMARU APM 사용자를 위한 온라인 매뉴얼 제공

- OPENMARU APM의 사용자 가이드 등 문서를 온라인 매뉴얼로 제공합니다.
- 또한, "거침없이 배우는 JBoss" 책을 웹 사이트에서 제공하고 있습니다.

온라인 매뉴얼 사이트 - https://www.openmaru.io/docs





OPENMARU APM 주요기능

Google Analytics 방식의 사용자 통계 적용

구글 애널리틱스와 동일한 사용자 측정 알고리즘을 이용한 방문자 수 계산 방법을 제공합니다.

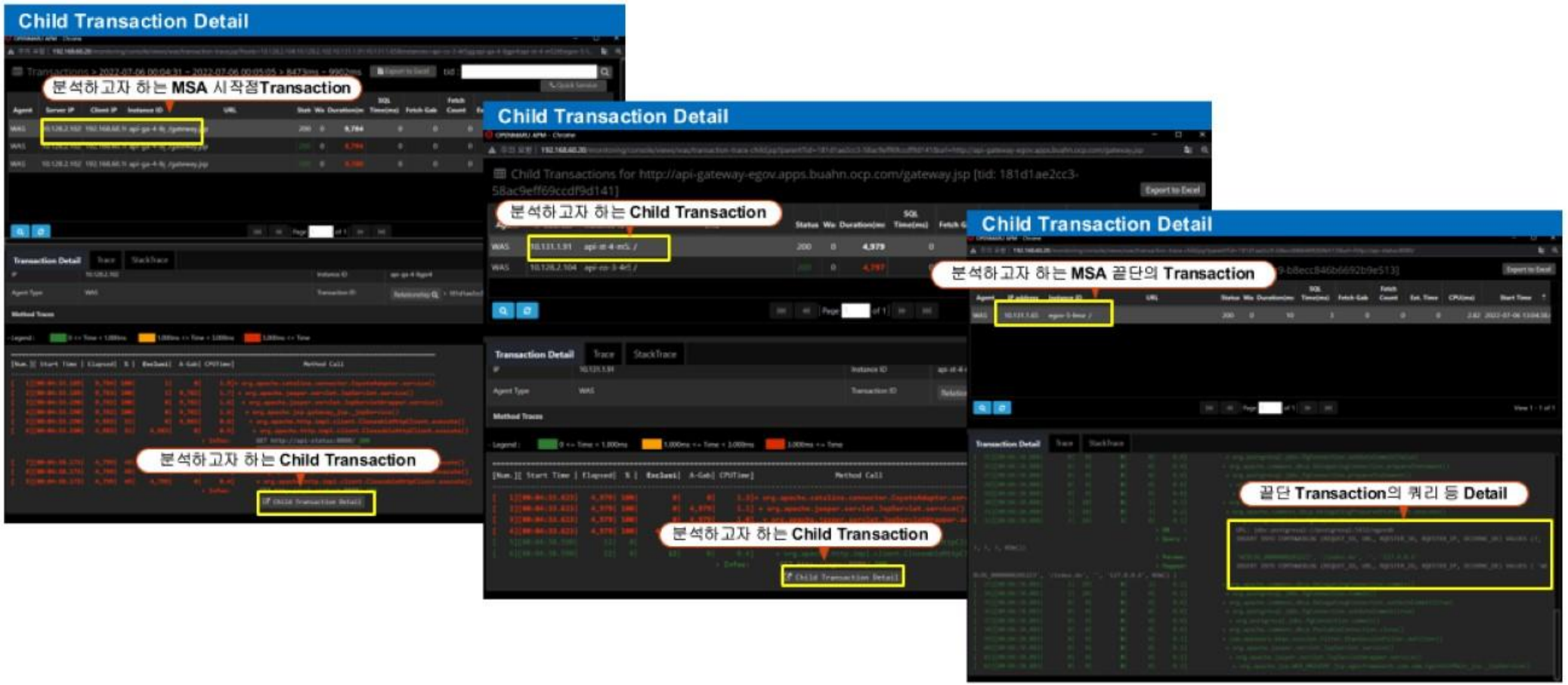
- 구글 애널리틱스에서 활용한 알고리즘을 활용한 액티브 사용자 수 기능 제공, 사용자가 여러 WAS 인스턴스에 동시 접속하더라도 동일 사용자로 계산함
- 일일(DAU)/주간(WAU)/월간(MAU) 사용자 분석 기능 제공
- 로그인 세션 아이디를 기반으로 사용자 수를 계산하여 제공

OPENMARU APM 주요기능

마이크로 서비스(MSA) 호출 추적 기능(1/2)

마이크로 서비스(MSA)와 같이 애플리케이션이 분산되어 있는 환경에서 호출된 API를 추적하는 기능 제공

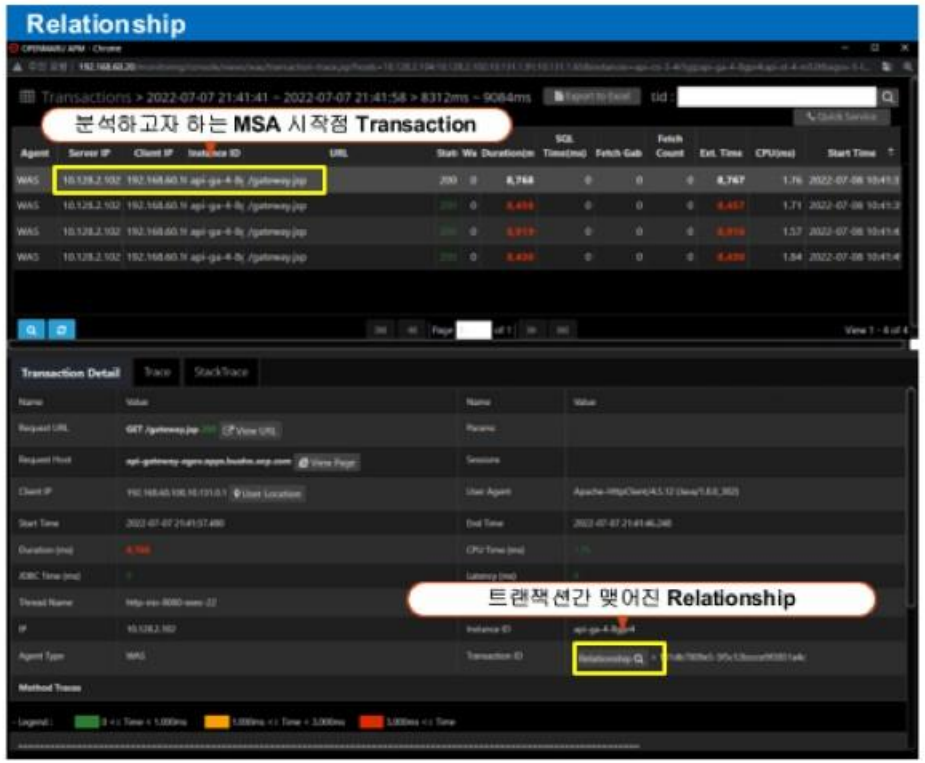
- MSA 환경에서 업무간 호출되는 REST API(HTTP) 호출에 대해 추적하는 기능 제공
- 호출관계를 추적하여 MSA 애플리케이션의 상세 수행을 확인할 수 있습니다.

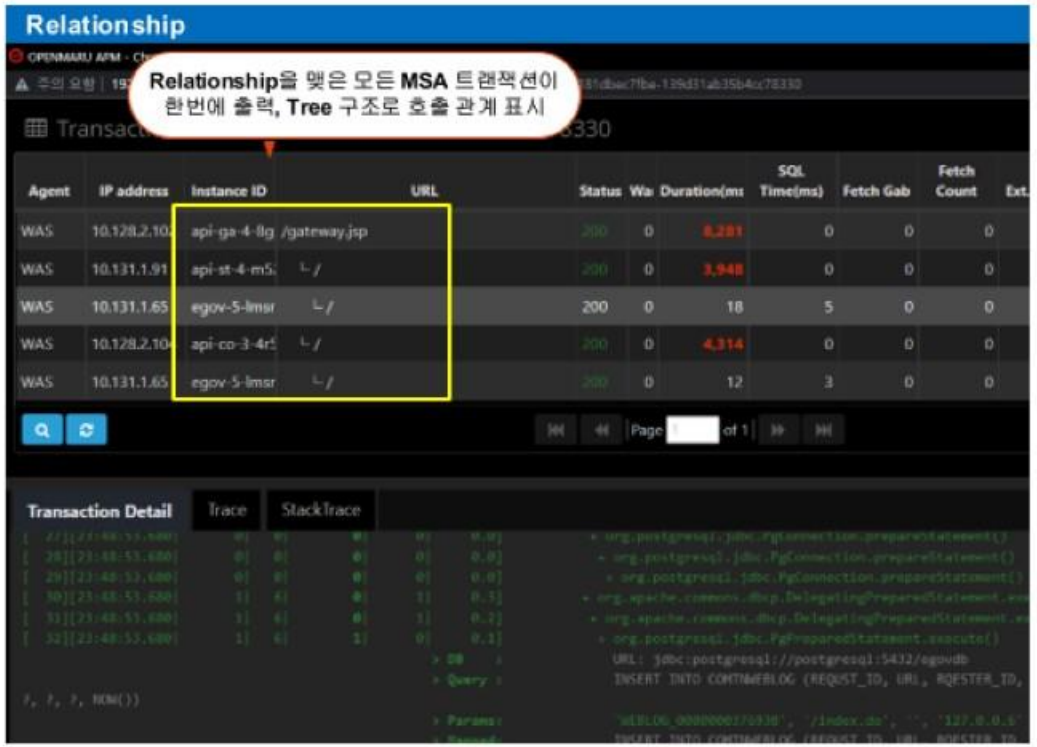


마이크로 서비스(MSA) 호출 추적 기능(2/2)

마이크로 서비스(MSA)와 같이 애플리케이션이 분산되어 있는 환경에서 호출된 API를 추적하는 기능 제공

- MSA 환경에서 REST API 호출 관계를 트리 형태로 보여주는 기능 제공
- REST API 호출관계를 트리형태로 표현합니다.

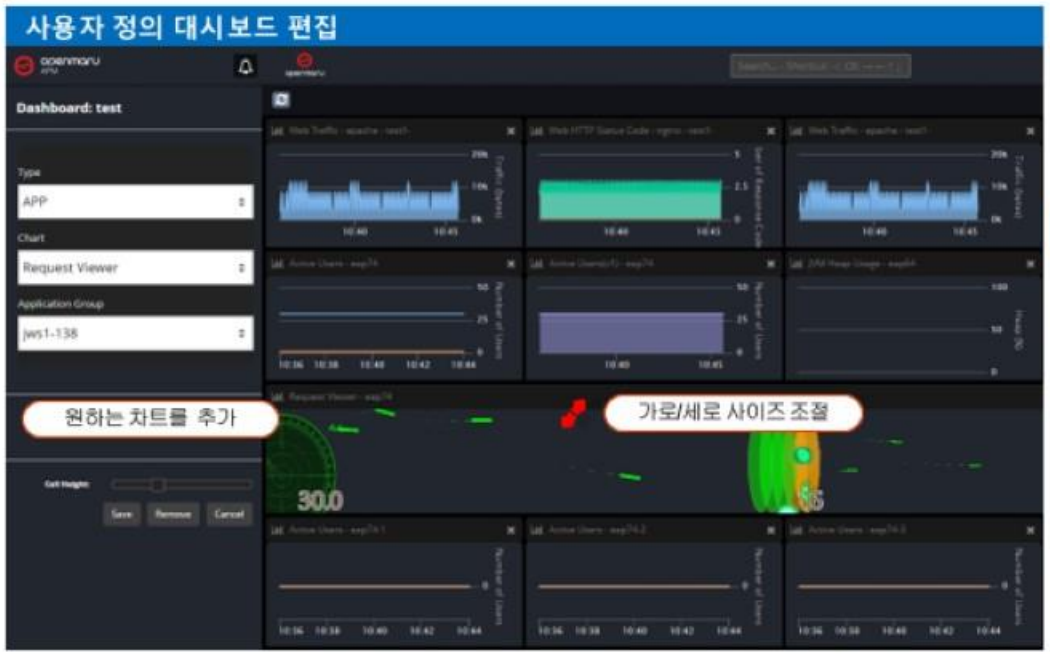




사용자 정의 대시보드 기능 지원

사용자 목적에 맞게 대시보드를 구성할 수 있는 기능제공

- N개의 애플리케이션 그룹 및 개별 인스턴스 차트를 별도로 구분하여 사용자가 원하는 그래프들을 대시보드로 정의하여 모니터링 할 수 있는 기능 제공
- 대시보드 모니터 환경에 맞게 세부적으로 차트 사이즈를 조절 할 수 있는 기능 제공



빠른 인스턴스 검색 기능 제공

모니터링 대상 인스턴스 이름 및 IP로 빠른 검색 기능 제공

- 인스턴스 이름 검색을 통해 빠르게 메뉴를 이동 할 수 있는 기능을 제공
- 컨테이너 환경에서 삭제되어 숨겨진 인스턴스를 검색하고 곧바로 활성화하여 분석 할 수 있는 기능 제공

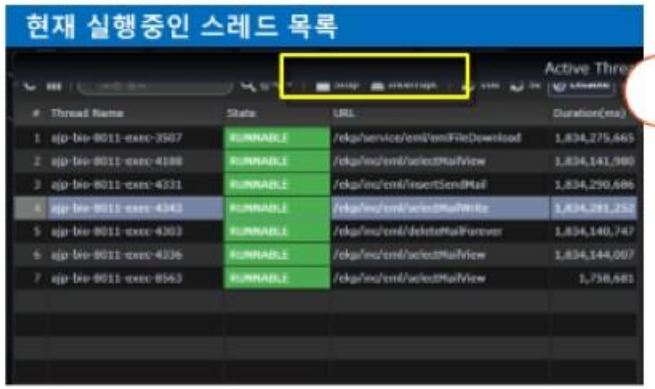
실행중인 트랜잭션에 대한 모니터링 및 제어기능

현재 실행중인 트랜잭션의 호출정보, 쿼리 정보를 모니터링하고 스레드를 중지하는 기능 제공

- 스레드 덤프를 생성하지 않고 실시간으로 트랜잭션을 모니터링 하는 기능 제공
- 실행중인 스레드 리스트, Current Stack(스택 트레이스), Transaction Trace(SQL, HTTP 등) 트랜잭션 기본 정보를 제공

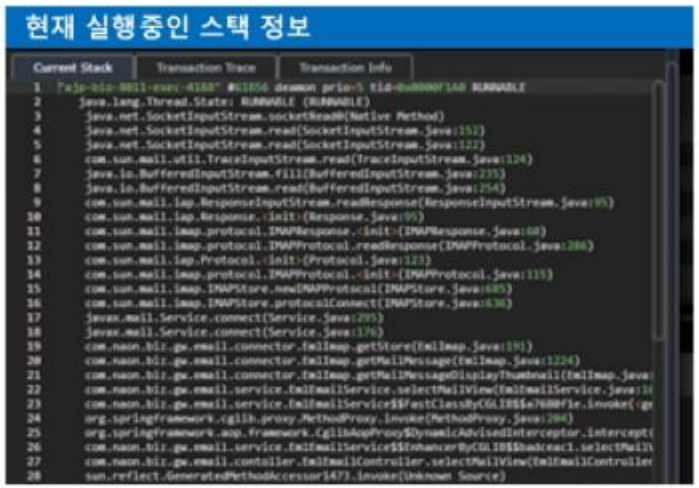
Active Thread Controller

현재 실행중인 스레드 목록

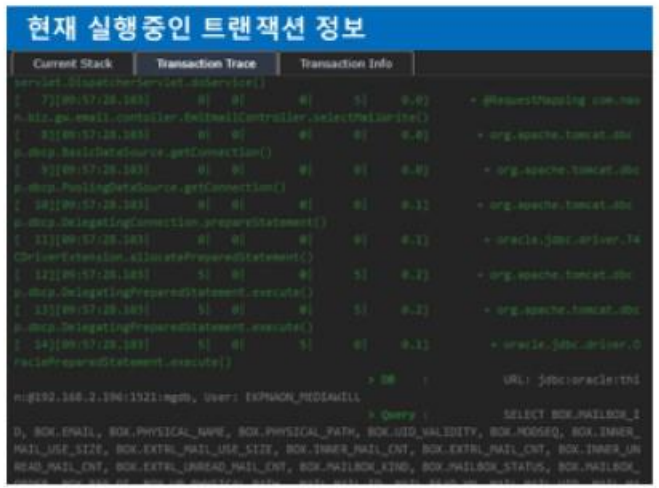


실행중인 스레드에 대한 중지, 인터럽트 기능

현재 실행중인 스택 정보



현재 실행중인 트랜잭션 정보



트랜잭션 URL 호출 정보

LDAP연동 로그인, Java 11, Spring Boot 지원

현재 실행중인 트랜잭션의 호출정보, 쿼리 정보를 모니터링하고 스레드를 중지하는 기능 제공

- 최신 Java 11 버전을 지원
- LDAP 및 Active Directory 를 이용한 APM 로그인 연동을 지원
- Spring Boot Embedded Dasource HikariCP 모니터링 지원

지원 Java 버전 업데이트

Java 11 OpenJDK 11

Spring Boot 2.x 버전 지원

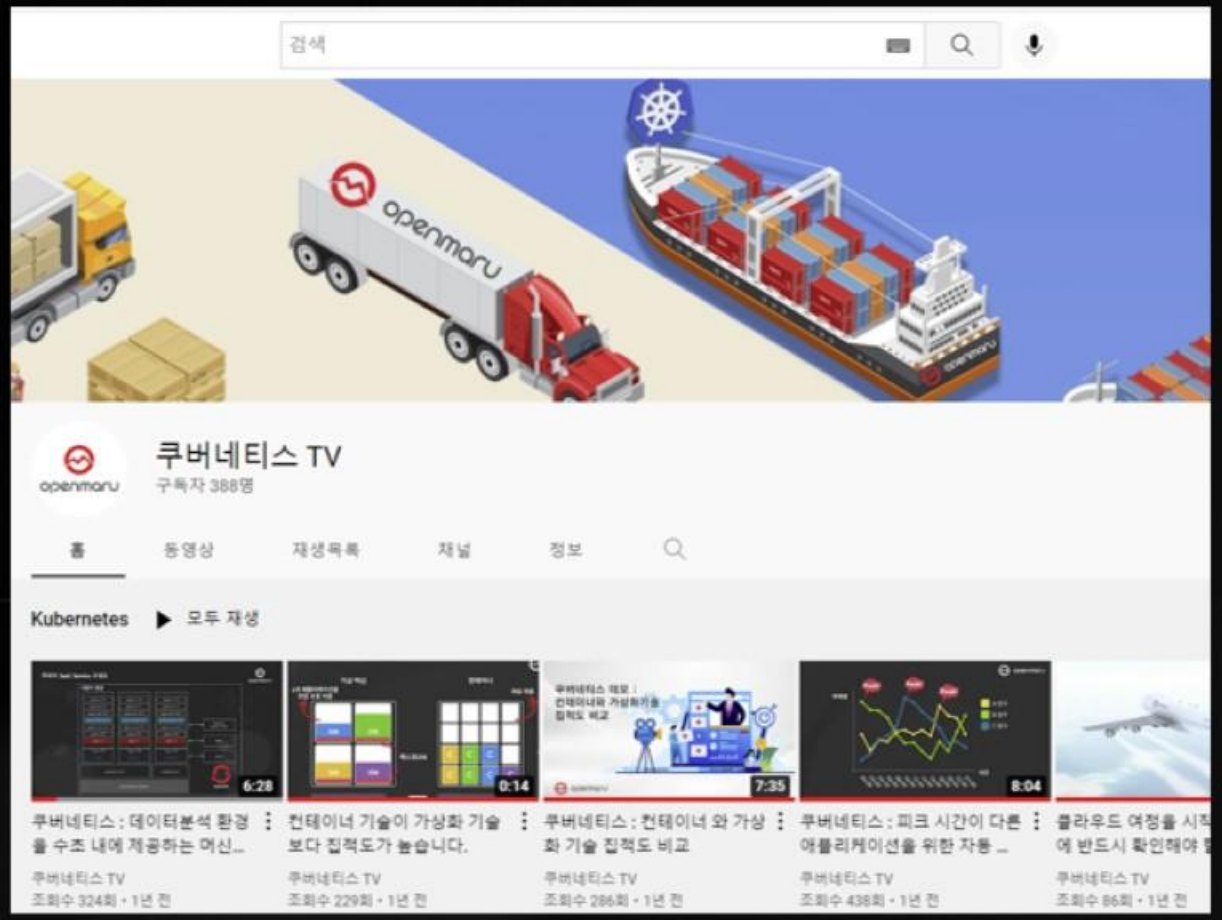
Spring Boot의 기본 Datasource인 HikariCP 모니터링

LDAP / Active Directory와 APM 로그인 연동 기능

LDAP/Active Directory과 연동한 로그인 기능 제공

Cloud Native Channel – Kubernetes TV by OPENMARU

- Cloud Native 에 진심인 기술 기업
- 2022년에만 10회 이상의 PaaS On line Seminar



Kubernetes TV on YOUTUBE

Cloud Native Seminar - TalkIT

Cloud Seminar for LGCNS

Cloud Native Seminar - Youtube

Cloud Seminar for 국민연금공단


openmaru


Application Performance Management
감사합니다.
openmaru
APM