


Open cloud Platform Summit 2024

(K-PaaS)

클라우드 네이티브 시대를 위한
업무중심의 모니터링

오픈마루

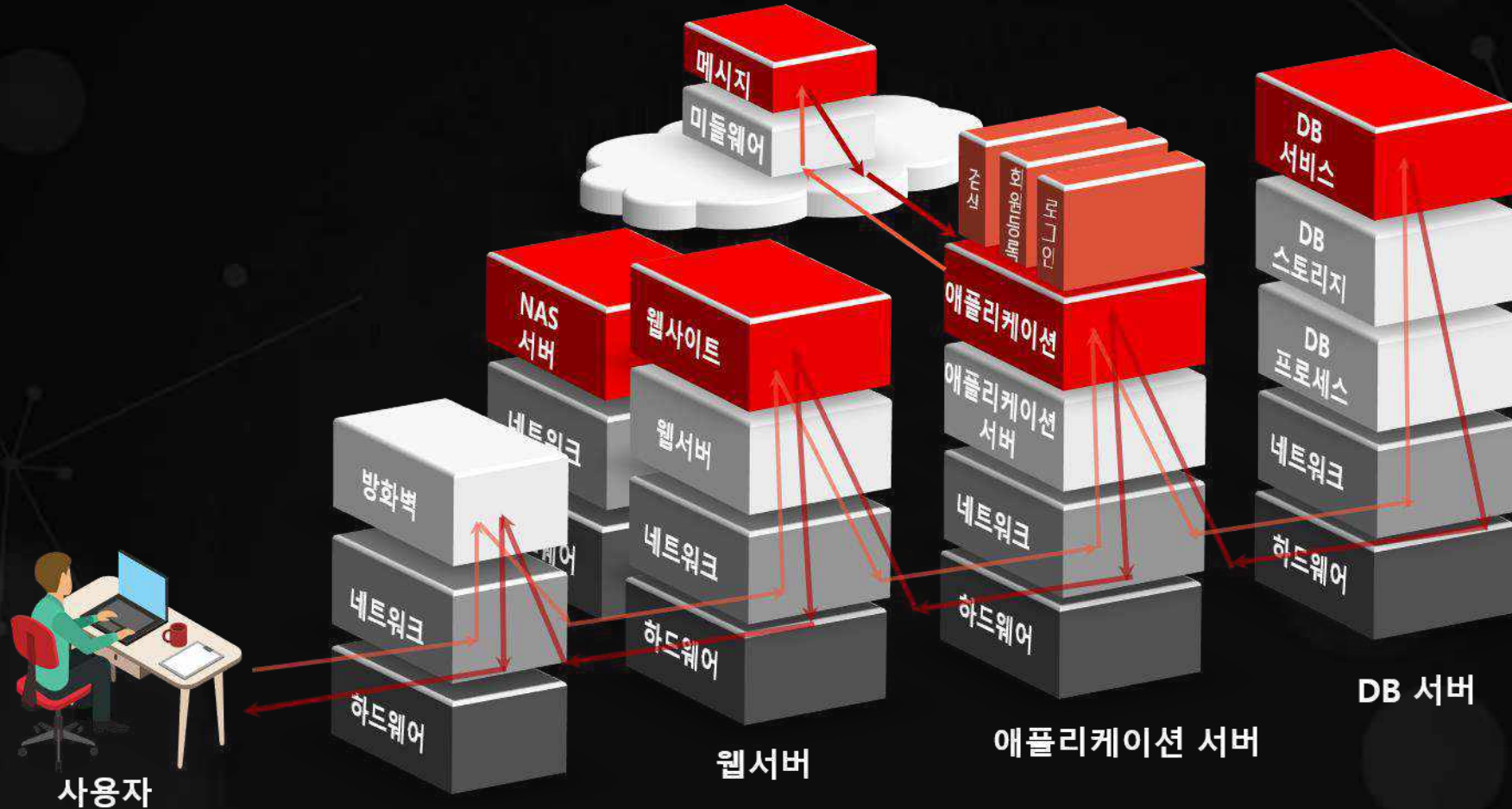
클라우드 네이티브 시대를 위한 업무중심의 모니터링



클라우드 네이티브 시대의 APM (Application Performance Management)

미들웨어는 시스템 장애의 관문이자 시작점

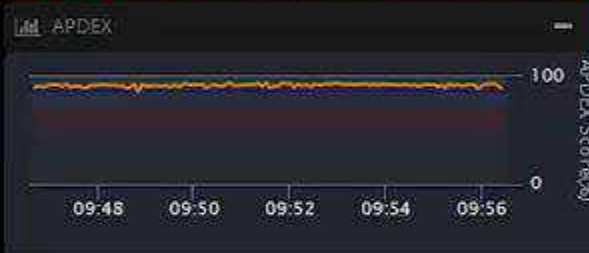
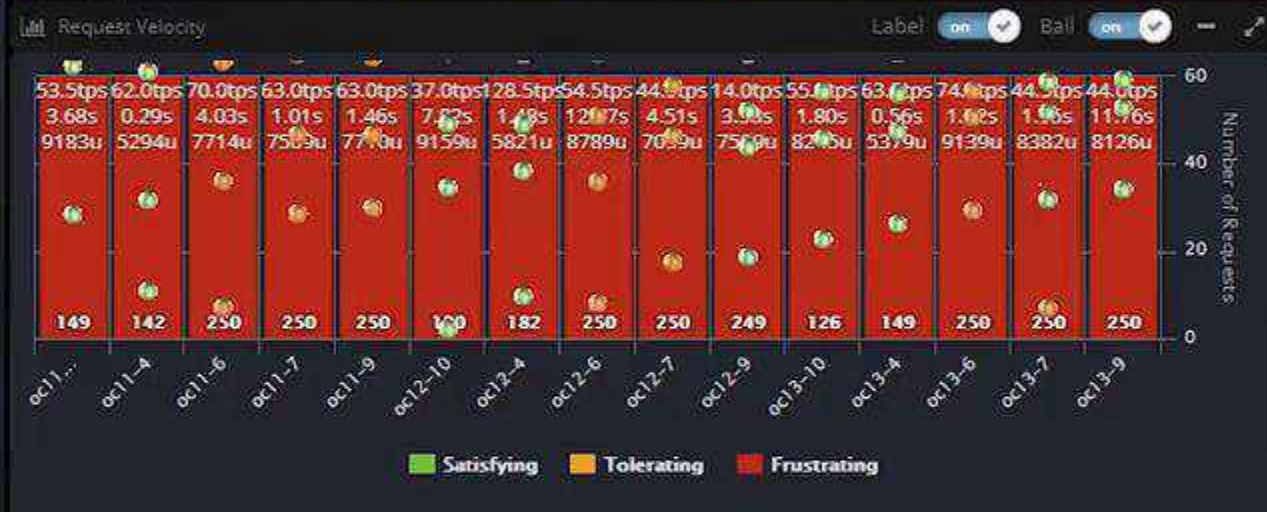
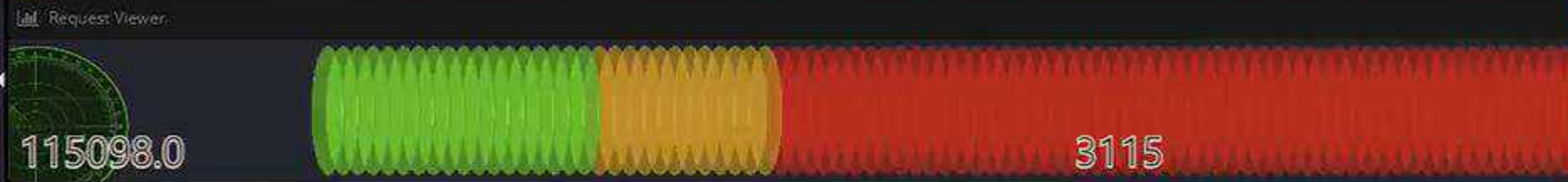
- 데이터베이스가 50% 느려진다면 사용자 응답시간은 어떻게 될까요?
- DB 테이블 변경으로 SQL 에서 오류로 인하여 페이지가 오류가 난다면?



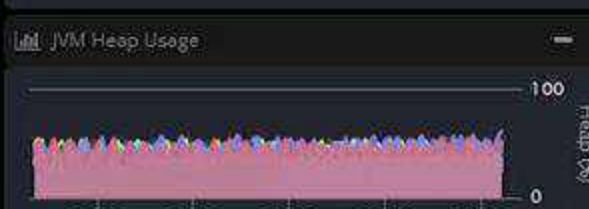
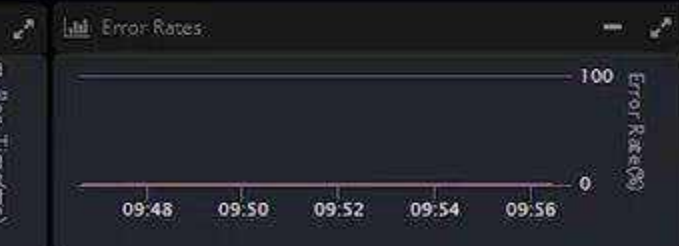
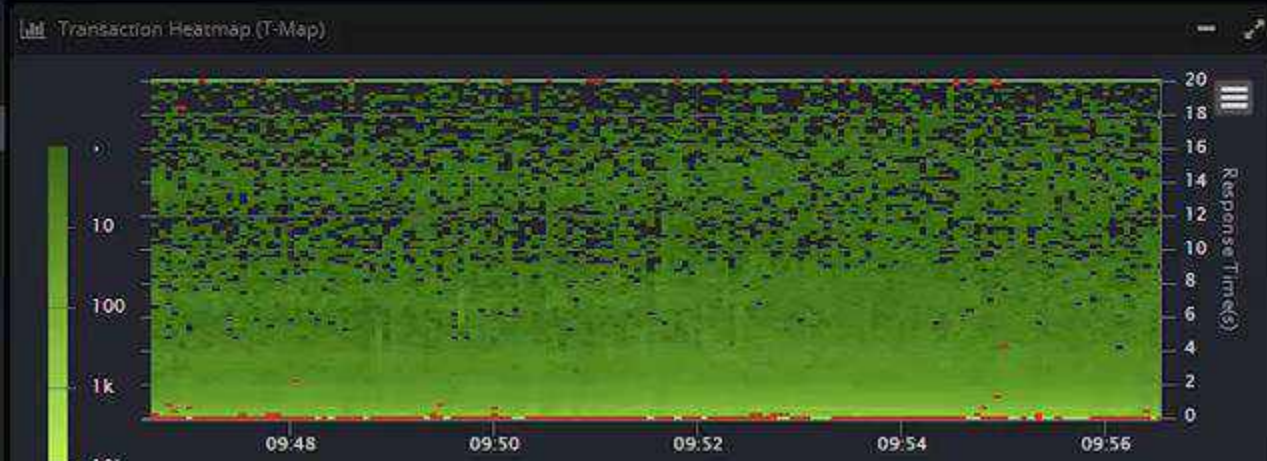
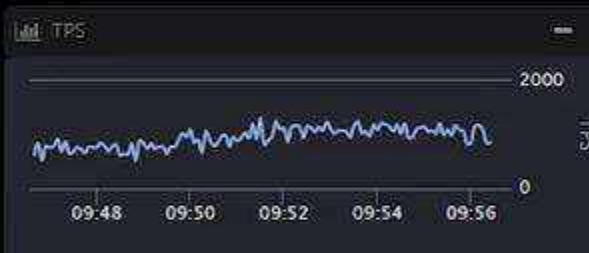
[긴급] 비정상 상황 발생 - 대시보드

[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



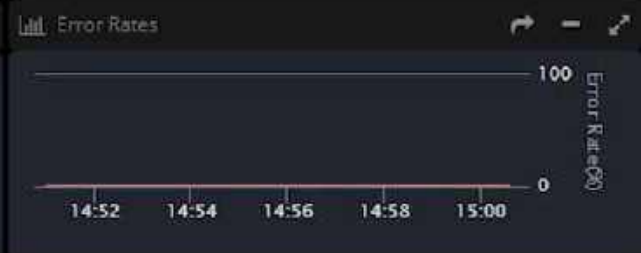
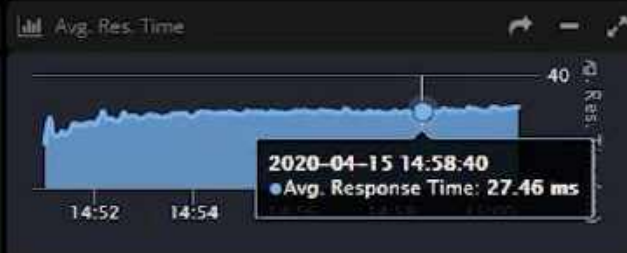
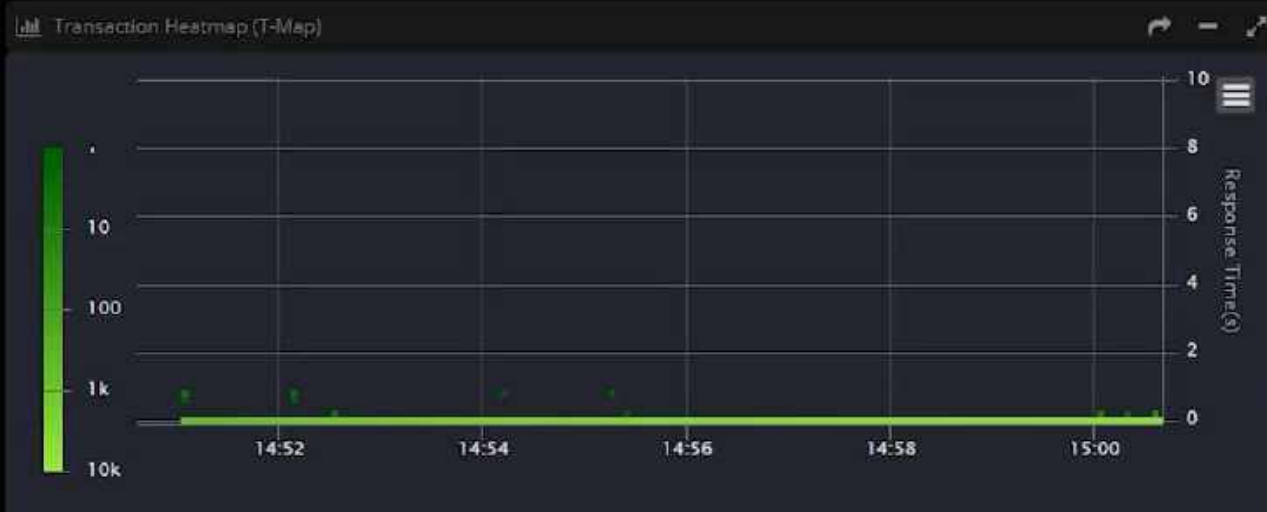
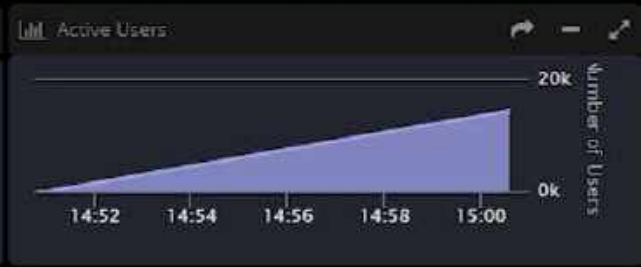
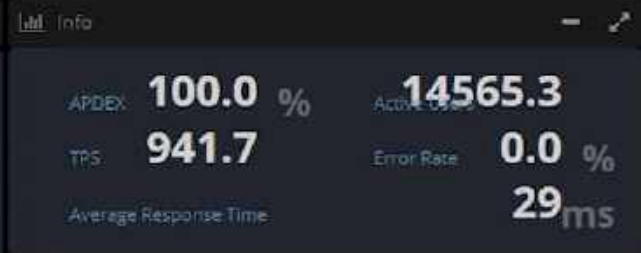
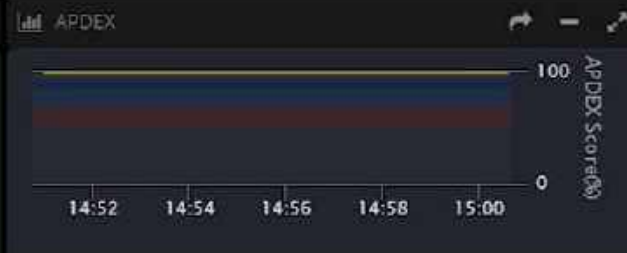
APDEX: **93.3 %**
 TPS: **1071.8**
 Average Response Time: **2,884 ms**
 Error Rate: **0.1 %**
 Total Requests: **115277.2**



[정상] 정상 서비스 상황



Off Sum EBS-STRESS oc-7-1 oc-7-2 oc-7-3 oc-9-1 oc-9-2 oc-9-3 oc-10-1 oc-10-2 oc-10-3



Application Performance Management



Cloud Native 시대

Development Process



WATERFALL



AGILE



DEVOPS



Application Architecture



MONOLITHIC



N-TIER



MICROSERVICES



Deployment & Packaging



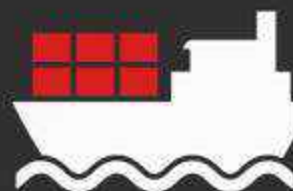
PHYSICAL SERVERS



VIRTUAL SERVERS



CONTAINERS



Application Infrastructure



DATA CENTER



HOSTED

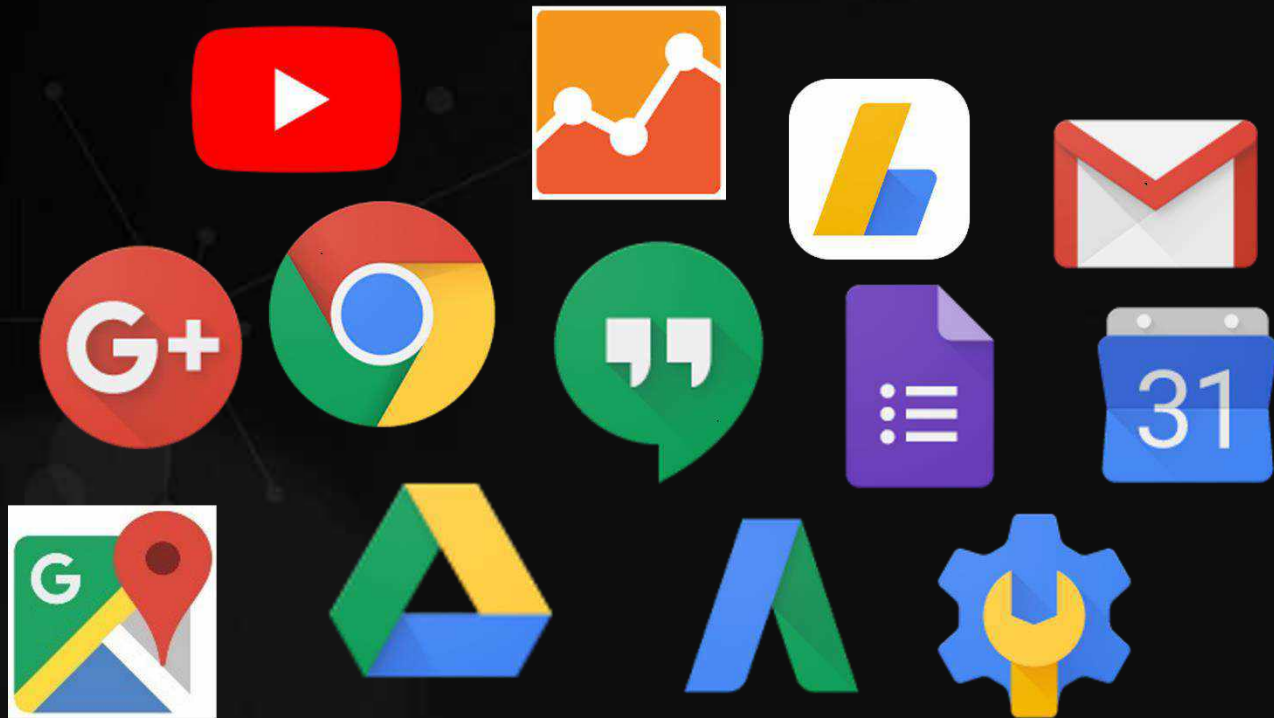


CLOUD



Google 의 모든 서비스는 **컨테이너** 에서 실행

- Gmail , 검색, 지도 ...
- MapReduce , GFS , Colossus ...
- Google Compute Engine의 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.



DevOps → 다양한 'OOO'Ops의 등장



'OOO'Ops에서 제일 중요한 것은 서비스



머신 중심에서 애플리케이션 중심 인프라로 변화

- 컨테이너화는 데이터 센터를 머신 중심에서 애플리케이션 중심으로 전환
 - 개발자와 운영팀에게 서버와 운영 환경에 대한 세부 사항을 추상화
 - 운영 중인 애플리케이션과 개발자에 미치는 영향을 최소화하면서 새로운 하드웨어 또는 운영 환경을 업그레이드하여 인프라팀에게 유연성을 제공
 - 서버의 CPU와 메모리 정보 뿐만 아니라 애플리케이션과 관련한 매트릭을 연결하여 오토 스케일링



Machine Centric Infrastructure

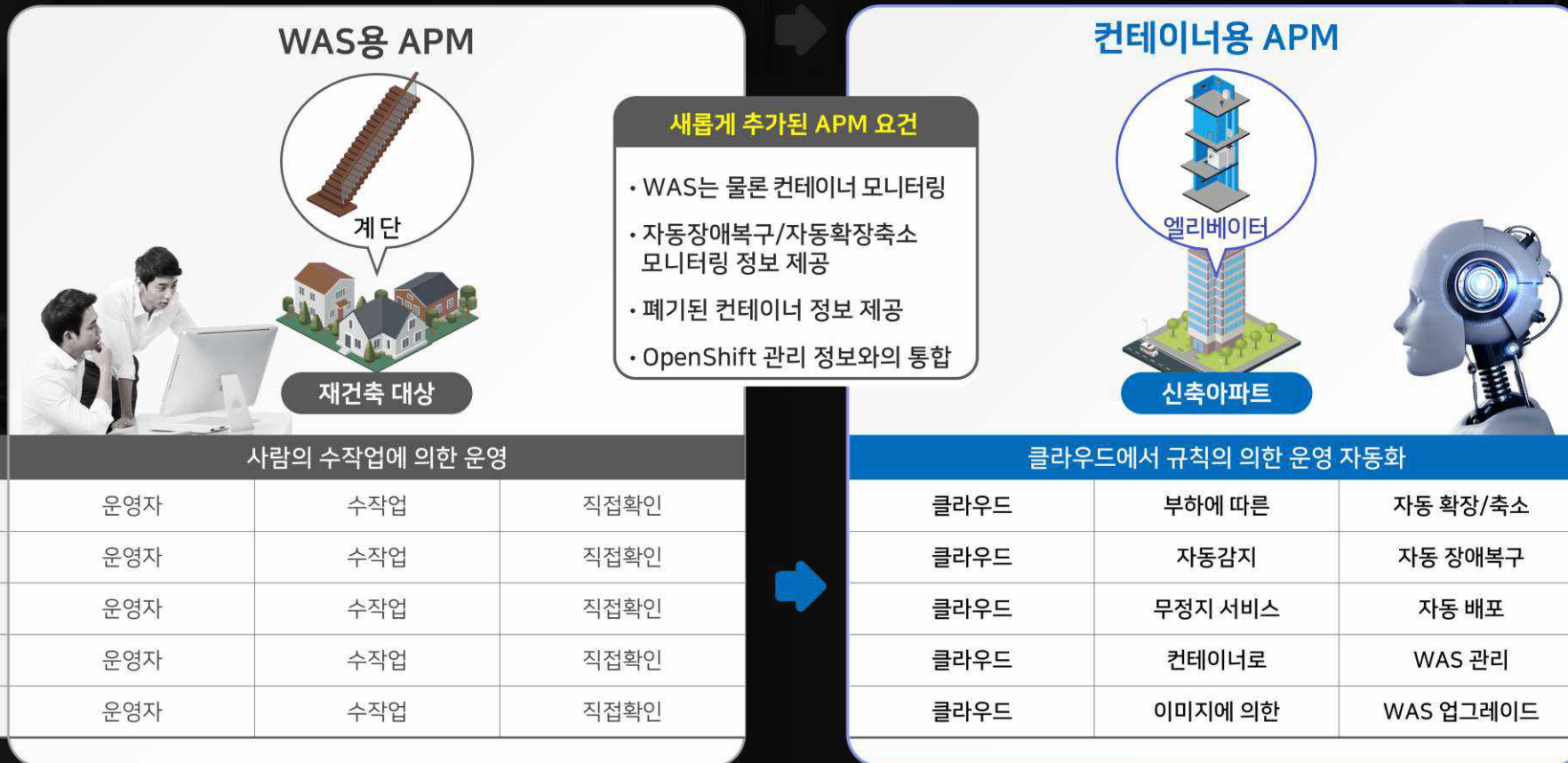


Application Centric Infrastructure



클라우드 전환에 따른 APM 요구사항의 변화

- 클라우드에서는 APM 모니터링 대상이 WAS에서 컨테이너로 변경
 - 컨테이너 단위로 WAS에 대한 확장/축소, 장애 복구, 업그레이드, 패치 작업을 모니터링
 - 기존 물리서버나 가상서버와는 달리 컨테이너는 휘발성으로 상태를 가지고 있지 않음

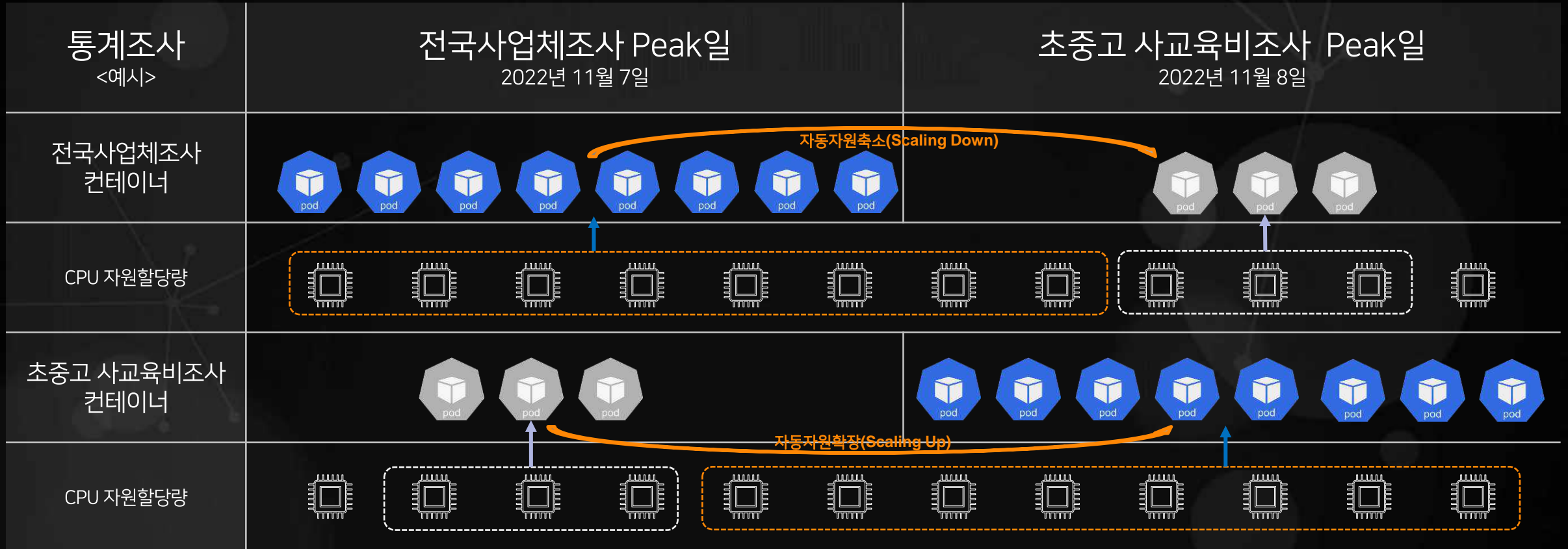


Application Performance Management

Peak 시점이 다른 애플리케이션의
Auto Scaling 모니터링 데모

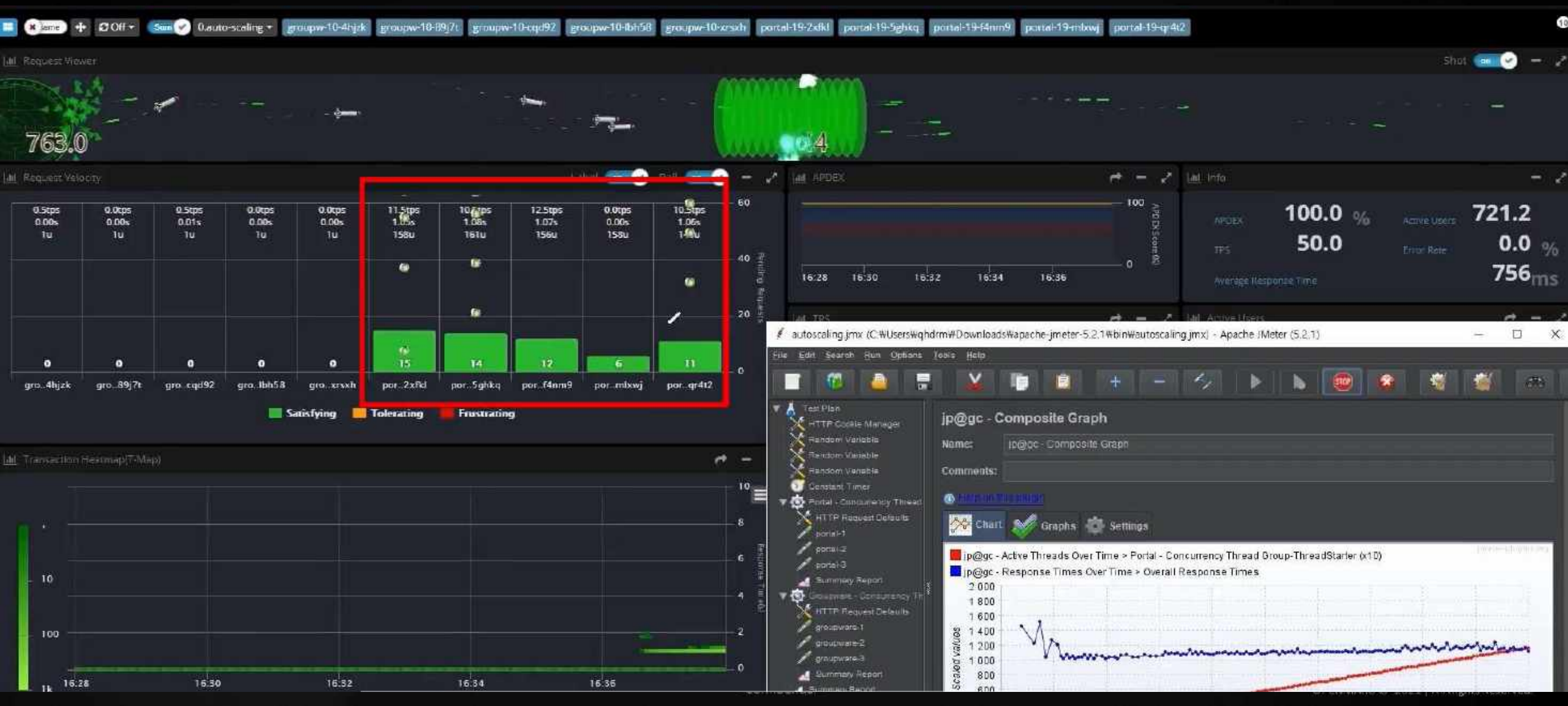
차세대 나라통계 - Peak 시점 별 자동 자원할당 활용 예시

- 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



인스턴스 개수 한정된 환경(기존 시스템)

기존에는 가상화 환경으로 부하 상황에서 리소스를 효율적으로 활용할 수 없음



T-02 전국사업체조사 Peak일 때 자동부하분산 환경 부하테스트

부하에 따른 컨테이너 자동확장으로 응답시간 보장과 TPS 증가



T-03 초중고 사교육비조사 Peak일 때 자동부하분산 환경 부하테스트

부하에 따른 컨테이너 자동확장으로 응답시간 보장과 TPS 증가



WAS / Dashboards / 대시보드

0 auto-scaling

Request Viewer

Request Velocity

APDEX

TPS

Transaction Heatmap(T-Map)

192.168.23.160:22 - root@bastion:~ - Xshell 6

```

ssh://root@192.168.23.160:22
root@bastion:~
root@bastion:~
root@node1:~
root@master:~
root@cobbler/var/named/dy...
Every 1.0s: oc get hpa ; oc get po -l deploymentconfig=portal-demo ; oc get po -l deploymentconfig=group... Mon Nov 16 17:07:03 2020
NAME                REFERENCE                               TARGETS  MINPODS  MAXPODS  REPLICAS  AGE
groupware-demo      DeploymentConfig/groupware-demo         93%/60%   5         10        5          18m
portal-demo         DeploymentConfig/portal-demo            0%/60%   5         10        5          3m35s

NAME                READY   STATUS    RESTARTS  AGE
portal-demo-19-2xfkl  1/1     Running   0          54m
portal-demo-19-5ghkq  1/1     Running   0          52m
portal-demo-19-f4nm9  1/1     Running   0          58m
portal-demo-19-nlxwj  1/1     Running   0          54m
portal-demo-19-qr4t2  1/1     Running   0          54m

NAME                READY   STATUS    RESTARTS  AGE
groupware-demo-18-4hjzk  1/1     Running   0          54m
groupware-demo-10-89j7t  1/1     Running   0          52m
groupware-demo-10-cqd92  1/1     Running   0          54m
groupware-demo-10-jz9bh  0/1     Running   0          18m
groupware-demo-10-jzj6z  0/1     Running   0          20m
groupware-demo-10-lbh58  1/1     Running   0          54m
groupware-demo-10-xrsxh  1/1     Running   0          52m
groupware-demo-18-zh4w9  0/1     Running   0          35m
    
```

9.0tps 1.07s 148u | 7.0tps 1.08s 99u | 8.5tps 1.09s 93u | 8.0tps 1.20s 100u | 6.5tps 1.09s 97u | 0.5tps 0.00s 1u | 0.0tps 0.00s 1u | 0.5tps 0.00s 1u | 0.5tps 0.00s 1u | 0.0tps 0.00s 1u

gro-4hjzk | gro-89j7t | gro-cqd92 | gro-lbh58 | gro-xrsxh | por-2xfkl | por-5ghkq | por-f4nm9 | por-nlxwj | por-qr4t2

Satisfying | Tolcrating | Frustrating

[정보-INFO] WAS가 시작되어 메인즈가 연결되었습니다.
 발생에이전트 : groupw-10-zh4w9@groupware-demo-10-zh4w9[10.131.0.206]
 클릭하여 상세한 정보를 확인하세요... 5s

[정보-INFO] WAS가 시작되어 메인즈가 연결되었습니다.
 발생에이전트 : groupw-10-jz9bh@groupware-demo-10-jz9bh[10.131.0.205]
 클릭하여 상세한 정보를 확인하세요... 5s

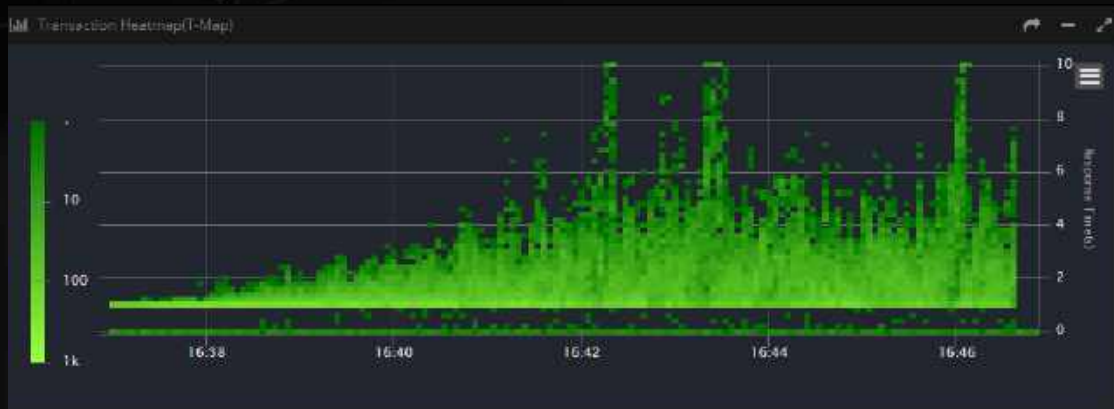
[정보-INFO] WAS가 시작되어 메인즈가 연결되었습니다.
 발생에이전트 : groupw-10-jzj6z@groupware-demo-10-jzj6z[10.131.0.207]
 클릭하여 상세한 정보를 확인하세요... 5s

DEMO - 부하 테스트 결과 비교

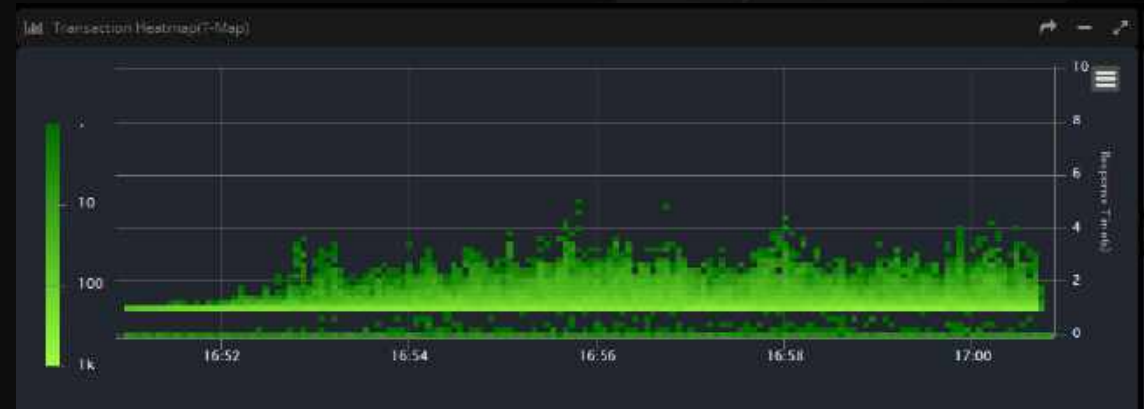
- 기존환경과 비교하여 **자동자원할당이 되는 환경이 1.7 배 많은 양을 처리하며, 평균응답시간이 2.5 배 빠름**
- **Peak 시점이 다른 통계조사에 대해 별도 개입없이 자동자원할당이 가능함을 확인**

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-03	Peak 시점이 다른 자동 자원할당이 되는 환경에서 부하테스트	초중고 사고육비 조사 Peak 일 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



Application Performance Management

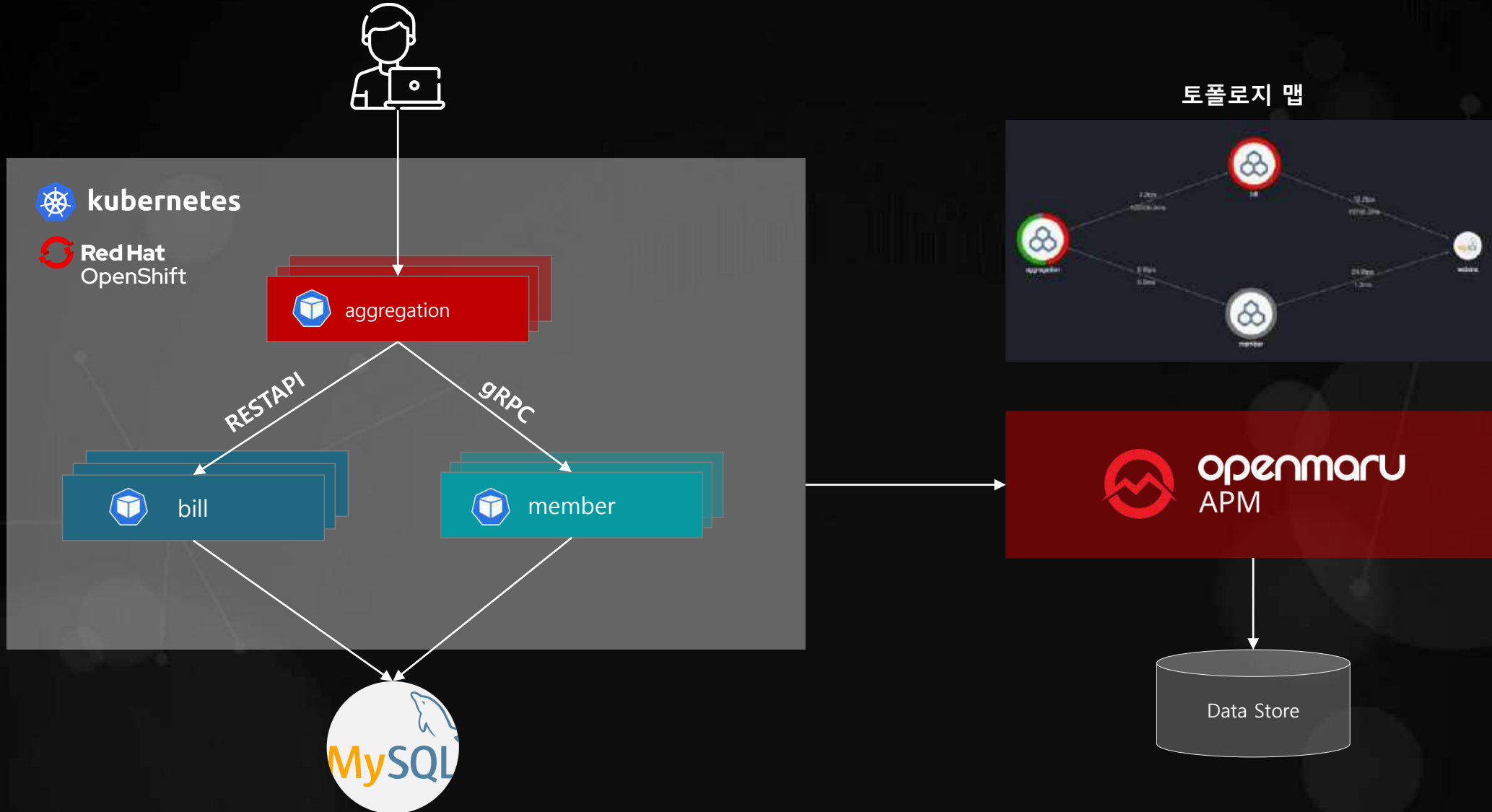
MSA(Micro Service Architecture)
모니터링



MSA(Micro Service Architecture) 모니터링 기능



MSA 모니터링 데모 구성



openmaru AFM Administrator

WAS / Dashboards / 그룹 토폴로지맵

Topology

애플리케이션 그룹 토폴로지 맵

```

    graph LR
      aggregation((aggregation)) -- "3.2tps  
120039.9ms" --> bill((bill))
      aggregation -- "6.0tps  
0.0ms" --> member((member))
      bill -- "12.8tps  
13755.2ms" --> webina((webina))
      member -- "24.0tps  
1.3ms" --> webina
  
```

aggregation, bill, member, webina

3.2tps, 120039.9ms, 12.8tps, 13755.2ms, 6.0tps, 0.0ms, 24.0tps, 1.3ms

OPENMARU, Inc. © 2016, All Rights Reserved. Version : 5.1.0, Build : 35e0f, Time : 2023-06-08 19:43:06 [User Guide] [Quick Service]



Kubernetes 환경의 장애분석

Kubernetes(OpenShift) 장애분석 방법

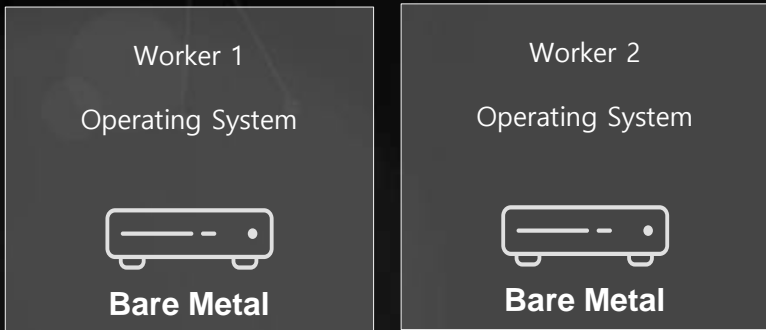
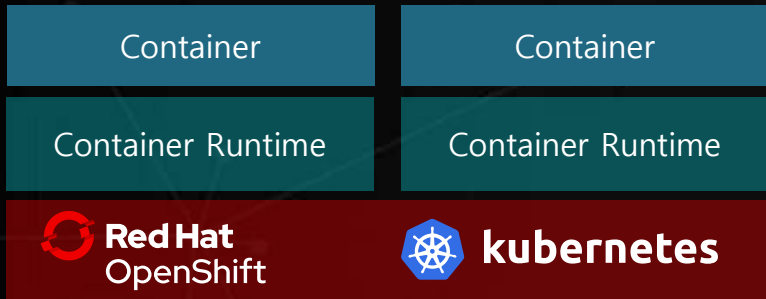
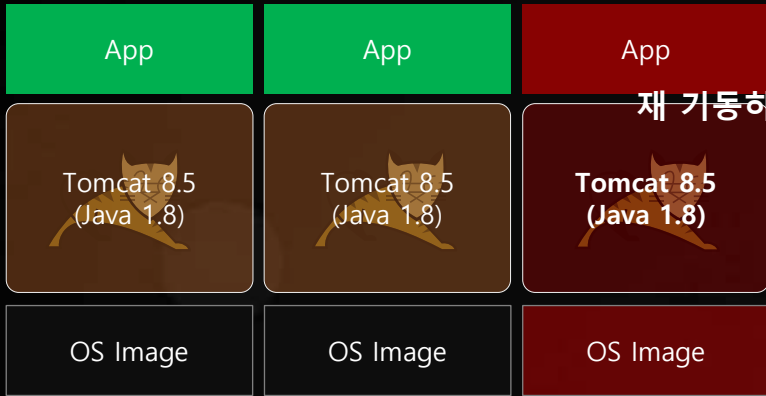
장애원인을 찾아려고, 웹 터미널에서 스크린샷을 받았는데, 복사하려다 재기동되어 사라져 버렸어요.

메모리가 부족하여 1주일에 한번씩 재기동되네요. 원인을 찾아야 해요.



Kubernetes(OpenShift) 장애분석 방법 - 느린 애플리케이션 원인 분석

Immutable Infrastructure

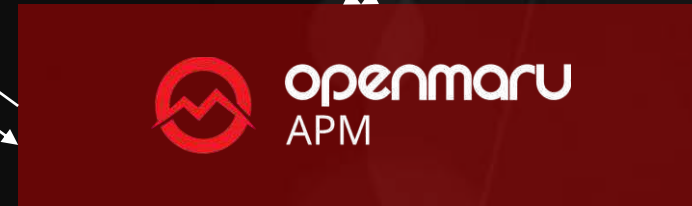


① 느린 서비스 확인

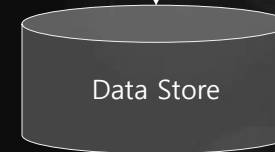
⑤ 느린 애플리케이션 원인분석



② 분석을 위한 데이터 요청
③ 원인 분석 데이터 전송



④ 분석 데이터 저장



- Administrator
- 나의 대시보드
- WAS 2/27
- Dashboards
- Hosts 2/27
- 애플리케이션 그룹 2/27
- 대시보드
- (0)ALL 2/27
- aggregation-client 2/4
- aggregation-grpc-server
- aggregation-http-server
- eap74
- homepage-23
- Instances
- 대시보드
- 만족도 지수(APDEX)
- 실시간 요청 모니터링
- 트랜잭션 맵(T-Map)
- 스레드맵 분석
- 초당처리량(TPS)
- 액티브 사용자수
- 사용자 통계
- 자원 트랜잭션
- 평균응답시간
- 응답시간
- JVM 메모리 비교



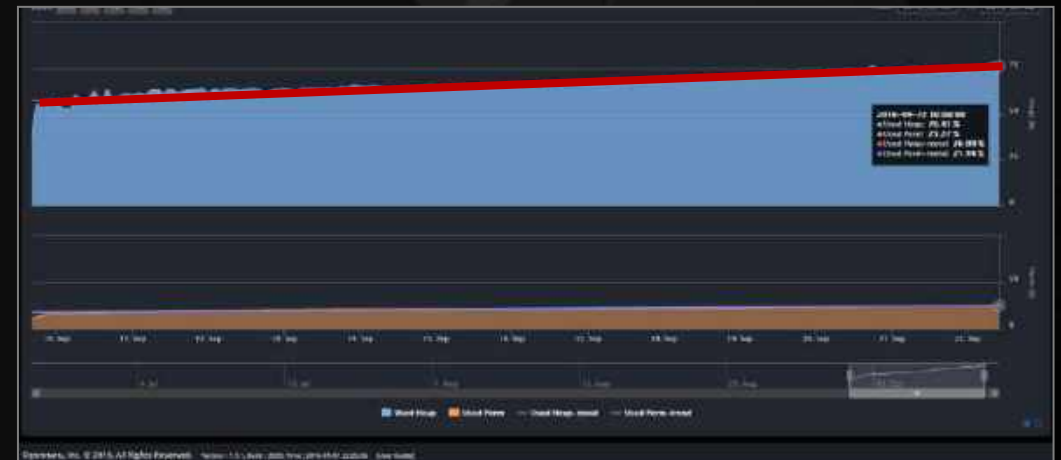
JVM의 OutOfMemory의 유형

- **메모리 부족 현상**
 - **Heap 메모리 OutOfMemory 오류**
 - 특정시각에 급격한 메모리 증가
 - **Native 메모리의 OutOfMemory 오류**
 - RES 영역 등 OS에서 사용하는 영역에서 발생
- **Memory Leak**
 - **메모리 사용량이 계속 증가해 결과적으로 메모리 부족한 상태**
 - **HTTP Session 객체의 증가, JDBC 관련 객체의 Leak등이 원인**
- **Java VM 버그**
 - **매우 드문 경우지만 JVM 버그의 가능성**

JVM 메모리의 급격한 증가



JVM 메모리가 점차 증가(Memory Leak)



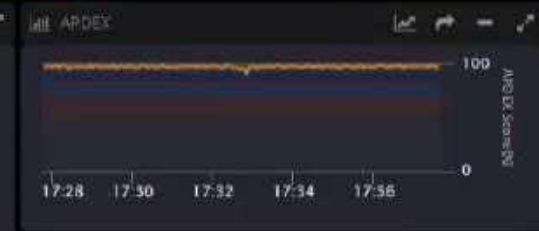
- Administrator
- 나의 대시보드
- WAS
 - Dashboards
 - 대시보드
 - 대시보드-리소스
 - 대시보드-그룹
 - 인스턴스 토폴로지맵
 - 그룹 토폴로지맵
- Hosts
- 애플리케이션 그룹
- 실시간 요청 모니터링
- 액티브 사용자수
- JVM 메모리 비교
- Web
- DBMS
- 시스템
- SLA 모니터링
- 이벤트
- 보고서
- 설정

WAS / Dashboards / 대시보드

[0]GAUS-INFRA gaus-infra-homepage-25-58829 gaus-infra-homepage-25-dv162 gaus-infra-homepage-25-hjrnq gaus-infra-homepage-25-pxr36 gaus-infra-homepage-25-tdqmw



서비스에 부하가 많은 상황

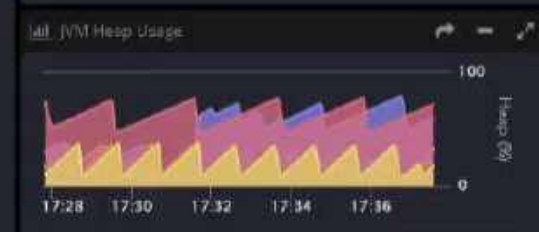
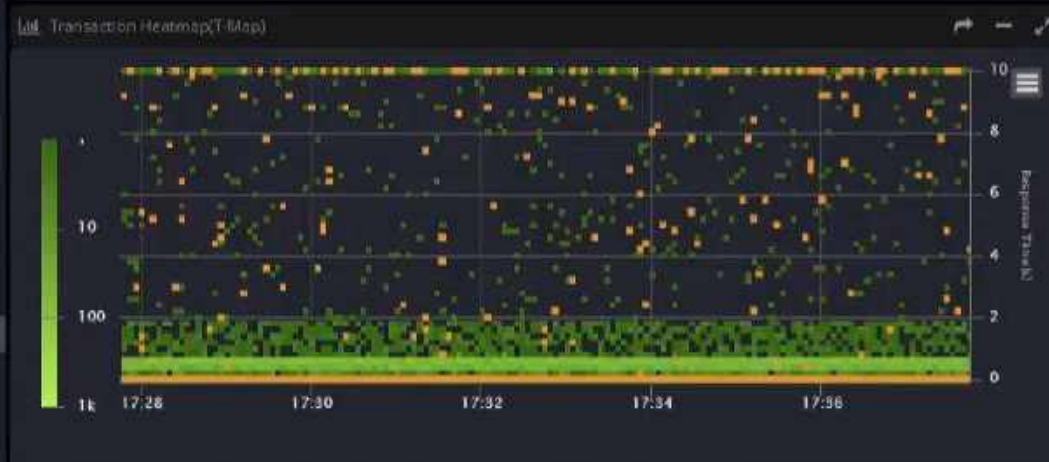
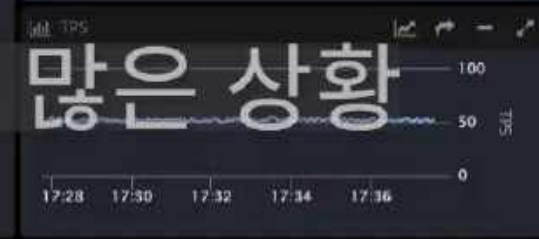


Info

APDEX: 98.7 % Active Users: 32.0

TPS: 52.5 Error Rate: 0.0 %

Average Response Time: 618 ms



클라우드 네이티브 장애관리와 성능상의 특화기능



- Immutable Infrastructure 로 시스템에 접속이 어려움
- 가상화 대비 컨테이너 개수가 수 배 이상 많음

Java 스레드 덤프 분석기

Lock을 추적가능, URL 정보표시

Java 메모리 누수 분석기

Java 메모리를 점유한 객체 분석/비교

네트워크 상태 분석기

Java 프로세스,시스템이 사용중인 네트워크 분석

오픈파일 분석기

Java 프로세스가 오픈한 파일 분석

시스템 프로세스 분석기

시스템의 프로세스 CPU, 메모리 사용량 분석/비교

데이터 추세 분석

과거 데이터의 증감 추세를 분석하는 기능

Cloud Native



Kubernetes 모니터링

Pod가 계속 재기동되었는데
알지 못했어요.
Pod가 재기동되면
알림을 받고 싶어요

부하가 많아 Pod Auto Scaling이
발생하면 알림을 받고 싶어요

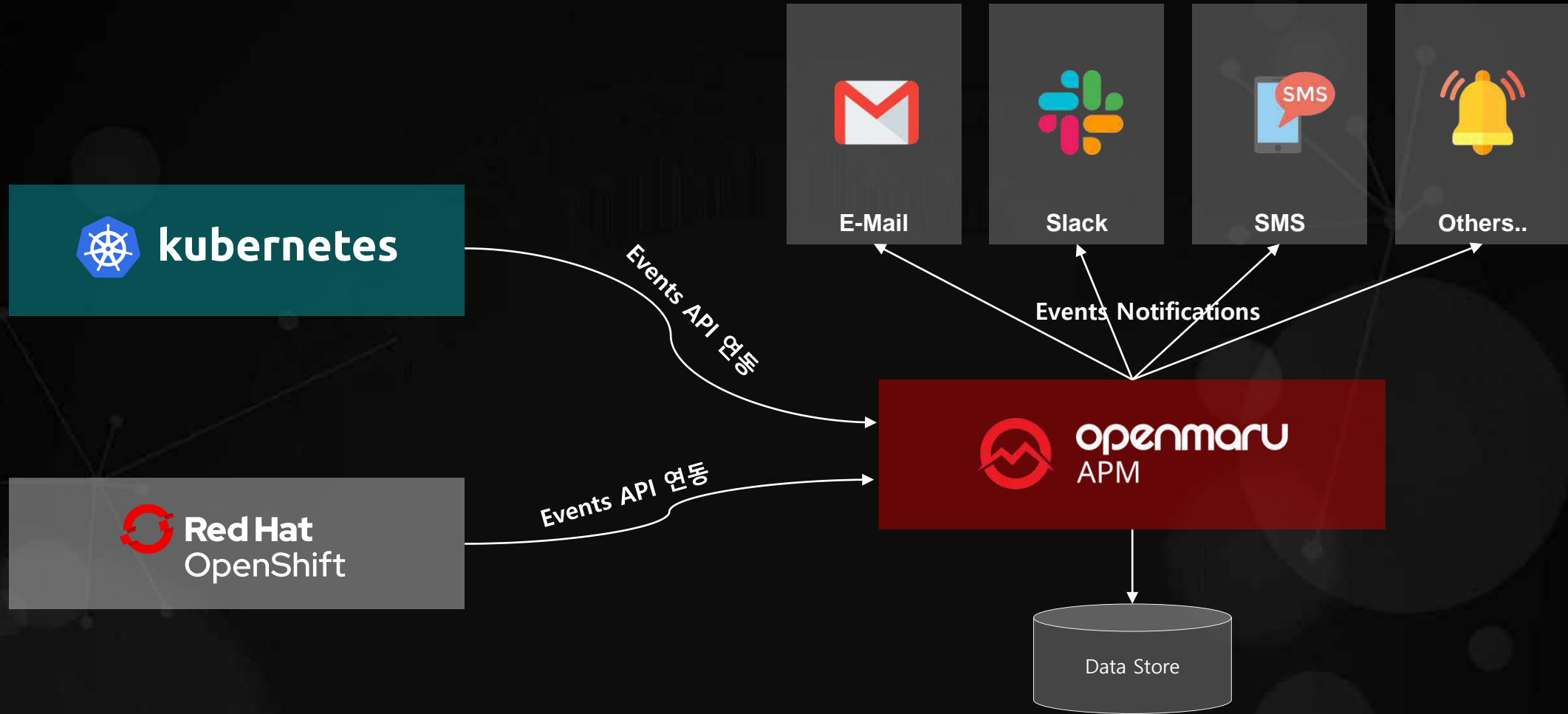


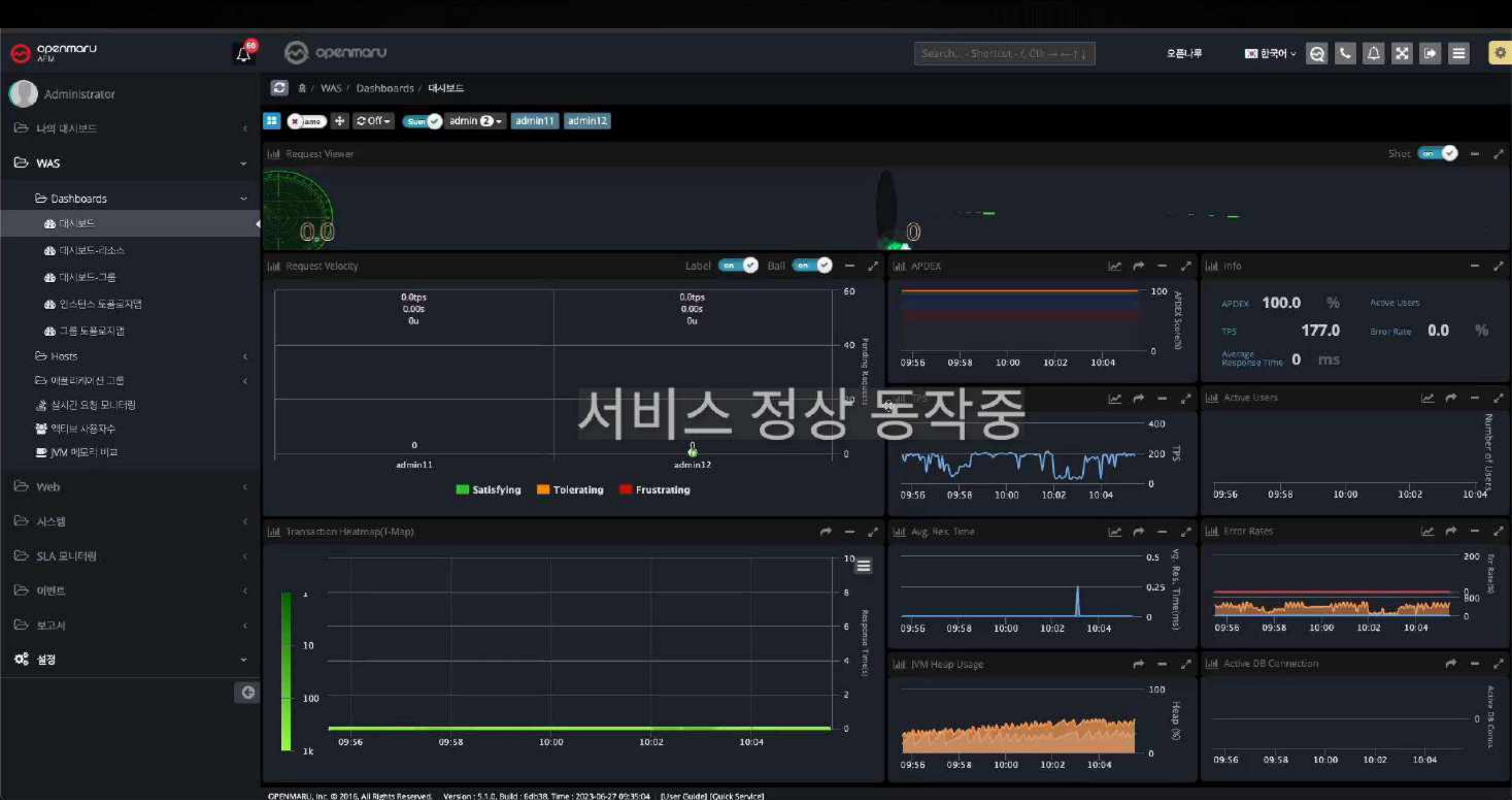
Kubernetes(OpenShift) 주요 이벤트의 종류



Events	주요 상황	비고
CrashLookBackOff	Pod가 시작되었지만 Crash가 발생한 후 재기동을 반복하는 것을 의미함	
ImagePullBackOff	노드에서 Image Registry에서 이미지를 가져오지 못할 때 발생	
Evicted	CPU, 메모리가 부족하여 리소스 확보를 위해 Pod를 제거할 때 발생하는 이벤트	
FailedMount, FailedAttachVolume	Persistence Volume에 접근할 수 없어 Volume을 연결할 수 없을 때 발생	
FailedSchedulingEvents	스케줄러가 Pod를 실행할 노드를 찾을 수 없는 경우	
NodeNotReady	Pod를 실행하는 데 필요한 노드를 사용할 수 없는 경우	
Rebooted	노드가 재부팅 될 때	
HostPort conflict	호스트 포트가 충돌할 때	

Kubernetes(OpenShift) 이벤트 알림







서비스 정상 동작중

Kubernetes(OpenShift) 이벤트 알림 - CrashLoopBackOff

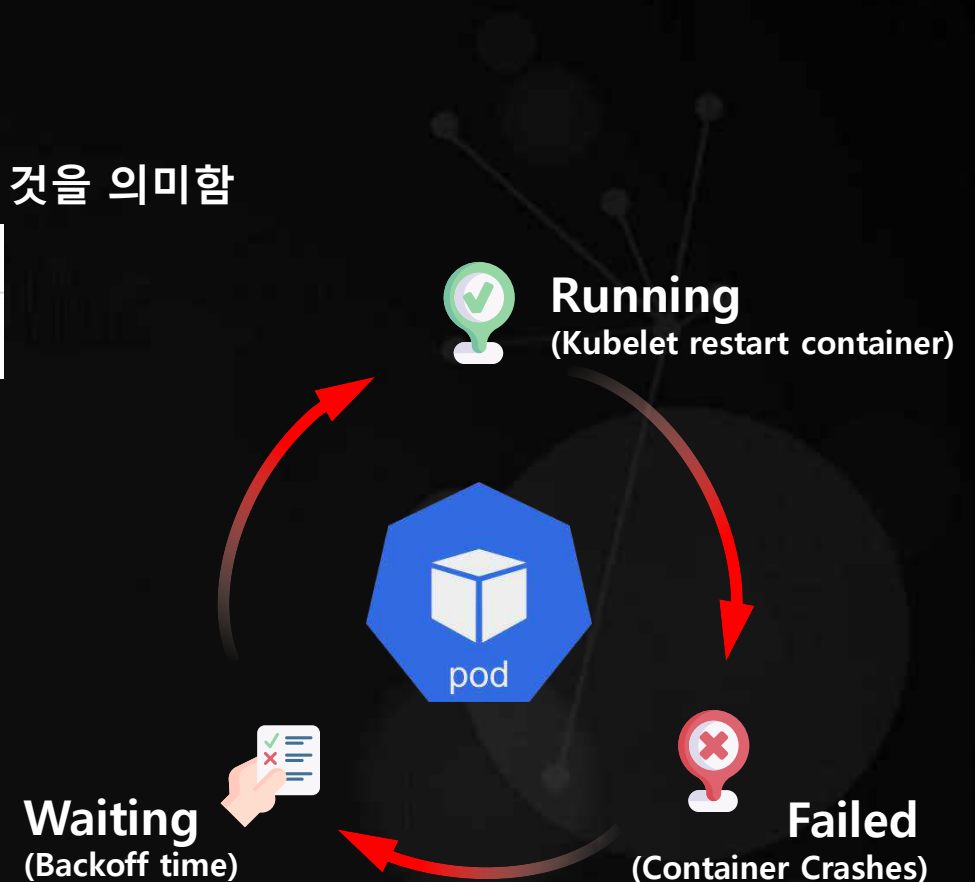
- CrashLoopBackOff란 ?

- Kubernetes의 Pod에서 발생하는 재시작 루프
- Pod가 시작되었지만 Crash가 발생한 후 재기동을 반복하는 것을 의미함

Name ↑	Status ↓	Ready ↓	Restarts ↓	Memory ↓
 echo-588c888c78-mfvqt	 CrashLoopBackOff	0/1	2212	8.2 MiB

- CrashLoopBackOff가 발생하는 주요 원인

- 설정파일 오류(오타)
- PersistentVolume과 같은 필요한 리소스가 없는 경우
- Command Line Argument가 틀린 경우
- Bind 포트를 사용할 수 없는 경우
- 파일 Write 권한이 없는 경우
- Liveness Probe가 실패한 경우
- 메모리가 부족하여 OOM Kill 되는 경우

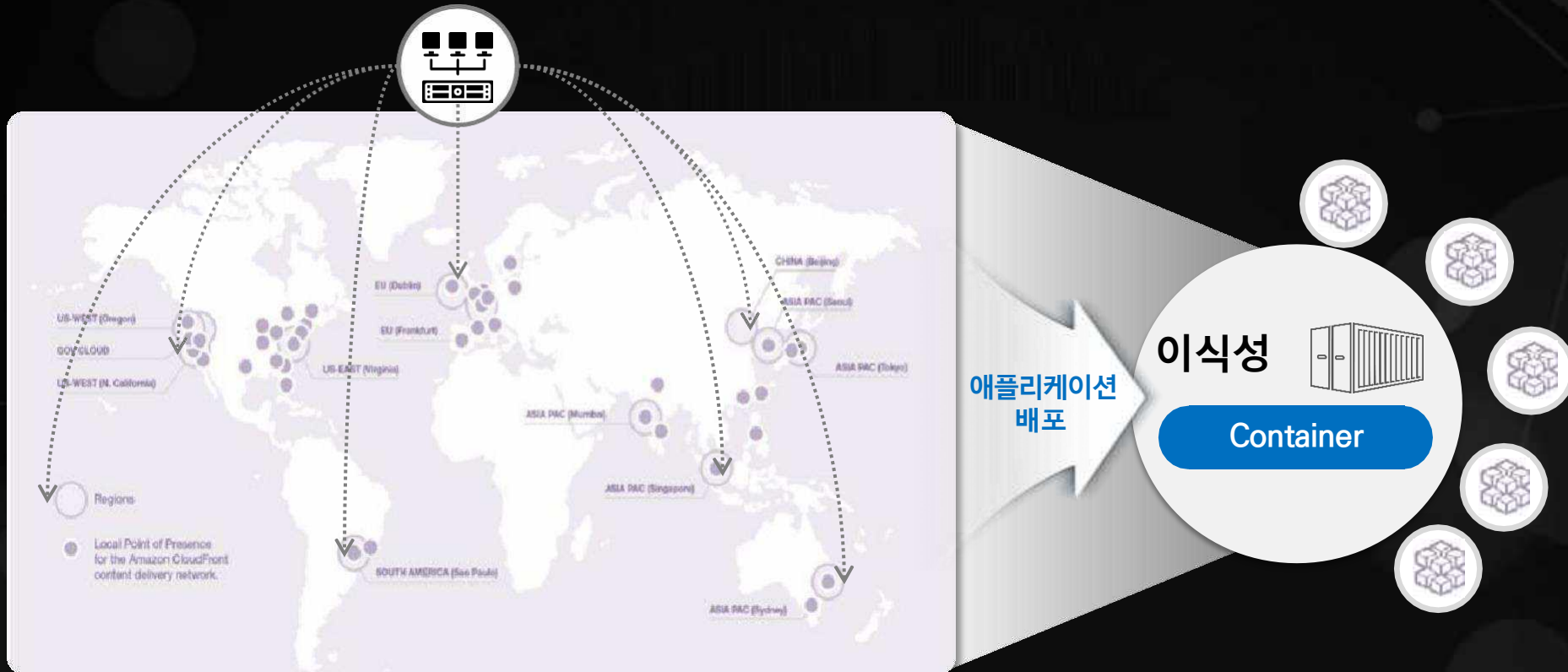


Application Performance Management

데모로 이해하는 클라우드 네이티브 - 하이브리드 클라우드 데모

클라우드 네이티브 특징

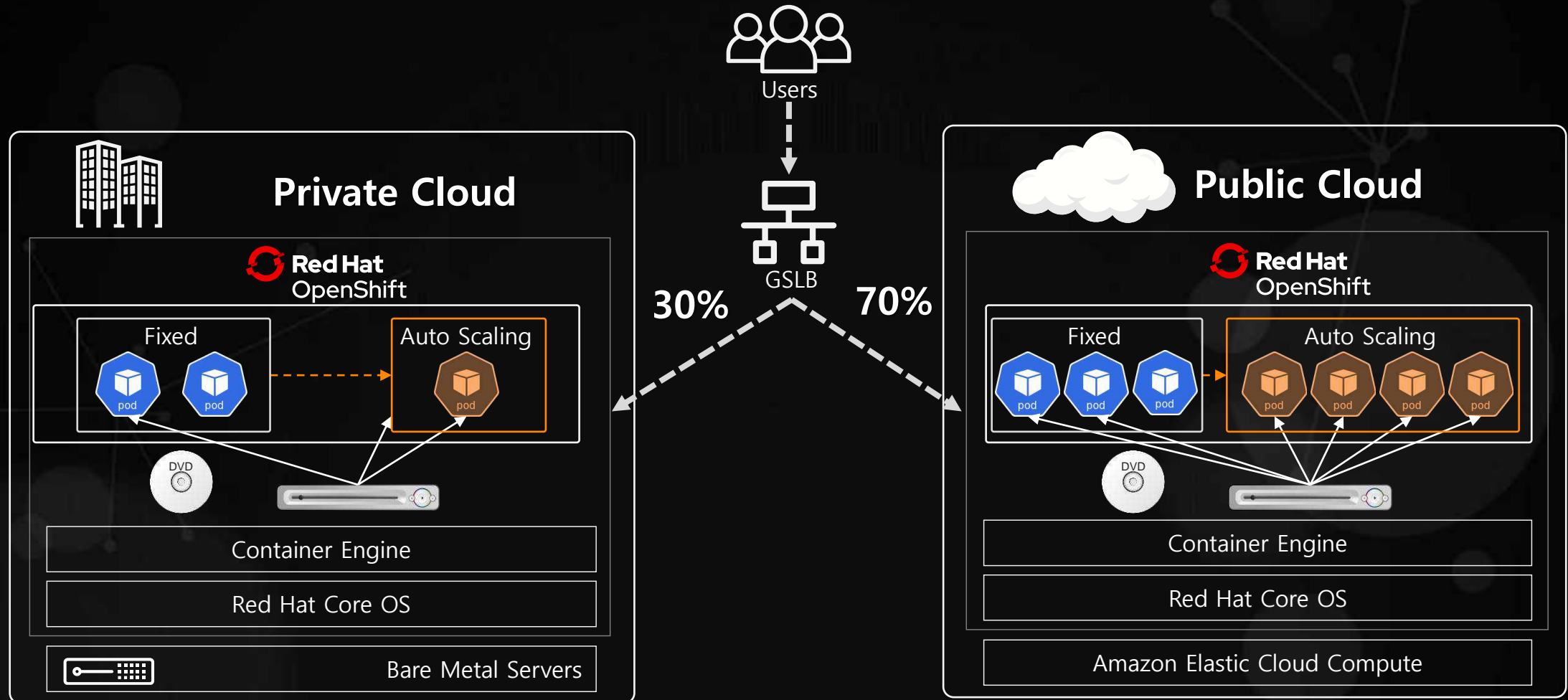
- 클라우드 네이티브는 작고, 가볍고 손쉽게 배포
- 클라우드 환경에서의 →서비스 배포는 전세계에 한번의 클릭으로 애플리케이션 배포 →전세계를 상대로 서비스 가능



Source : 클라우드 네이티브 추진 시 고려사항 (교육 교재)
클라우드 네이티브 기반 행정·공공 서비스 확산 지원 - 한국지능정보사회진흥원

하이브리드 클라우드 데모 - 사용자 증가 자동 확장

- 하이브리드 클라우드로 운영되는 포털을 접속하는데 자동확장이 필요한 사용자 폭주상태를 가정
- 내부 클라우드와 외부 클라우드에 모두 동일한 홈페이지 서비스를 하고 프라이빗 30% vs. 퍼블릭 70% 로 운영 중



하이브리드 클라우드 데모 - 전자정부 F/W 포탈에 대한 글로벌 서버 부하 분산

- GSLB (Global Server Load Balancing)을 통한 업무 부하 분산 데모
- 전자정부 F/W 포탈 서비스를 내부와 외부 클라우드에서 동시 운영

<https://portal.egov.openmaru.io>



<input checked="" type="checkbox"/>	portal.egov.openmaru.io	가중치 기반
<input checked="" type="checkbox"/>	portal.egov.openmaru.io	가중치 기반
<input type="checkbox"/>	private.egov.openmaru.io	단순
<input type="checkbox"/>	public.egov.openmaru.io	단순

30%

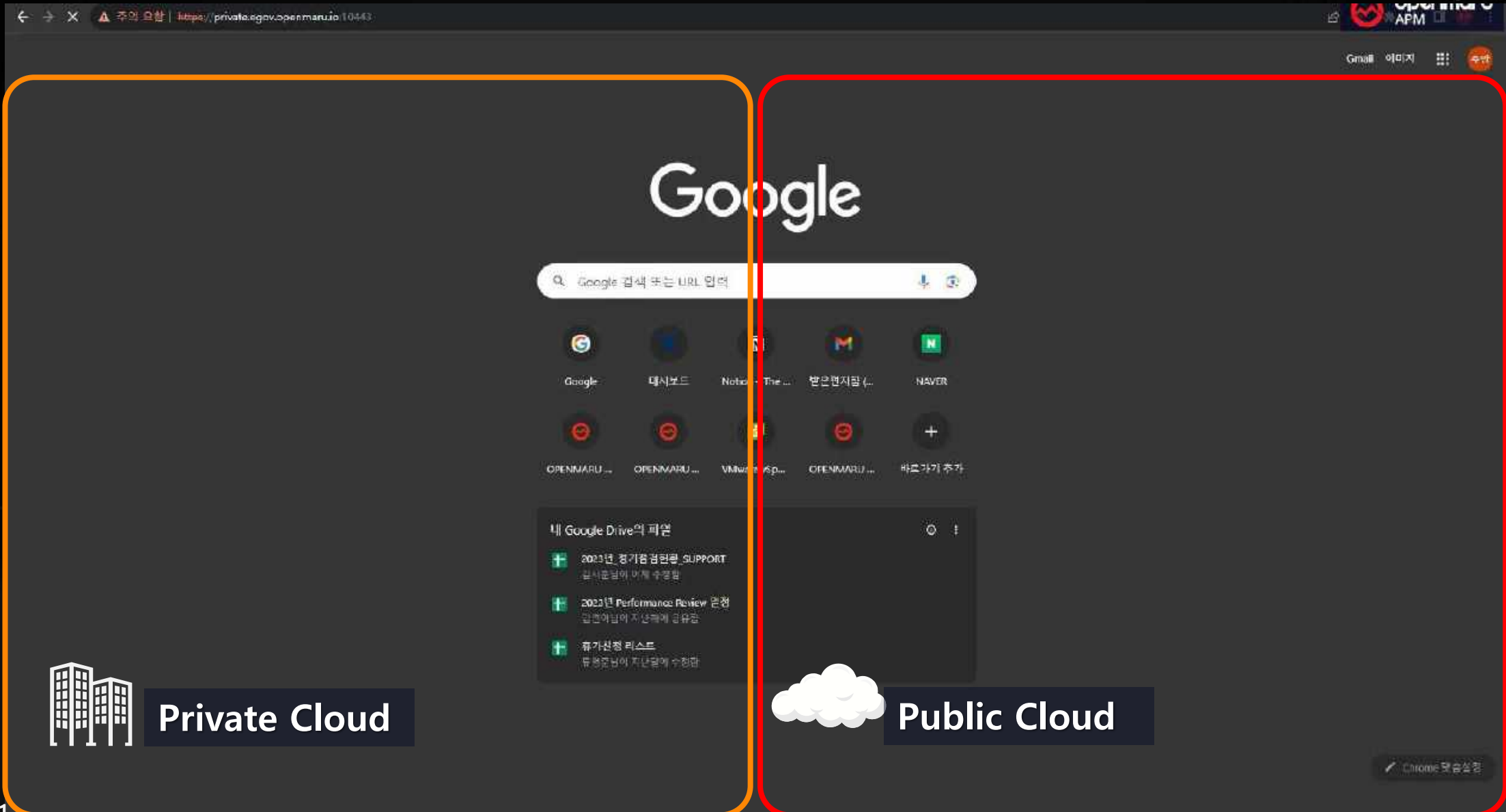
70%

<https://private.egov.openmaru.io>

<https://public.egov.openmaru.io>



하이브리드 클라우드 데모 - 자동 확장 이후

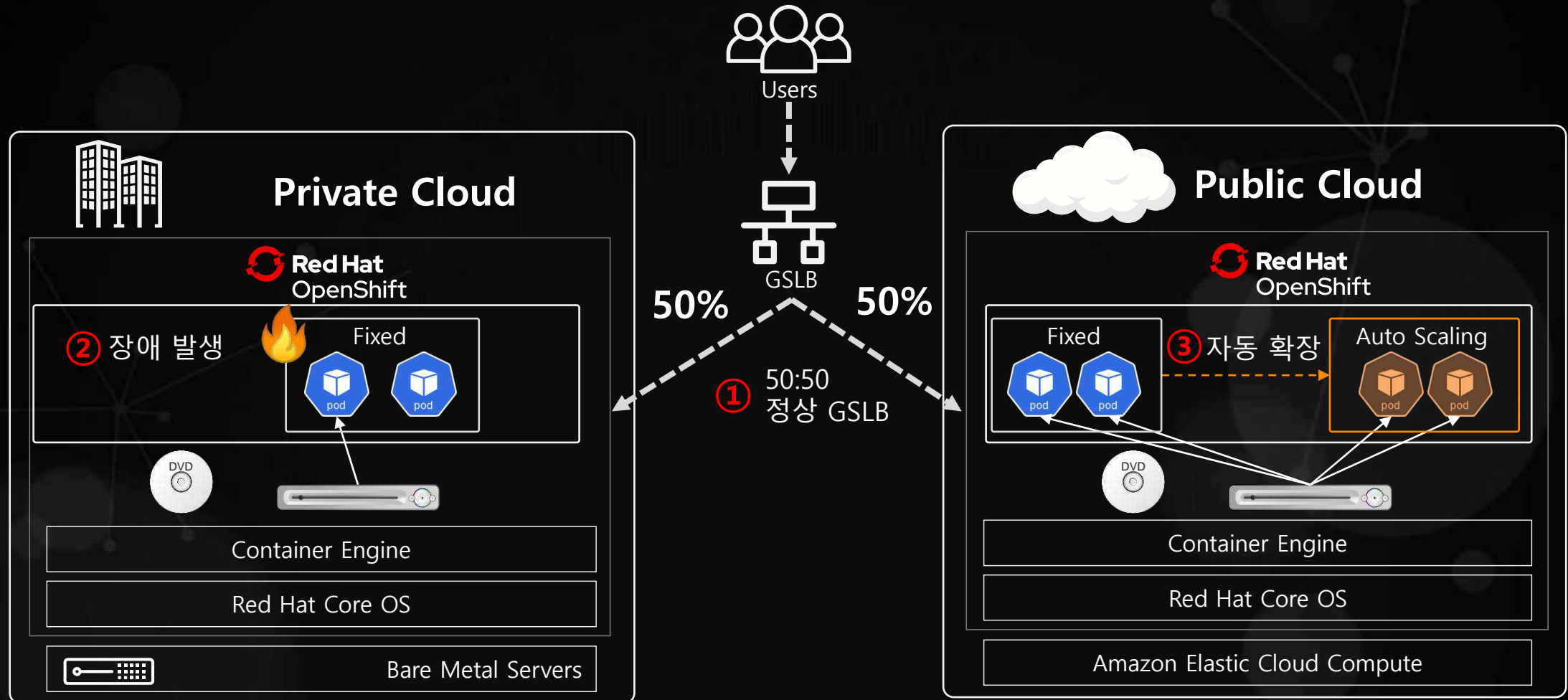


Private Cloud

Public Cloud

하이브리드 클라우드 데모 - Active Active DR

- 하이브리드 클라우드로 운영되는 포털을 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 - 장애 발생 이전 Active-Active GSLB



 Private Cloud

 Public Cloud

하이브리드 클라우드 데모 - Public Cloud 에서 부하에 따른 자동확장



Cloud Native



Cloud Native & Observability

Monitoring vs. Observability

- 클라우드 네이티브 환경은 복잡하기 때문에 클러스터의 문제 원인을 파악하는 것은 어렵습니다.
- 특정 노드에 문제 뿐만 아니라 컨테이너 이미지의 결함이나 비정상적인 POD 동작, 서비스 연결의 문제 등



서버나 대상의 상태를 지속적으로 감시,
관찰을 통해 예기치 못한 상황과 오류 대비

Monitoring



서버나 운영 환경에 직접 접속하지 않고,
시스템의 상태를 파악하는 것

Observability

Observability 3가지 요소

- 메트릭 , 트레이싱, 로깅 각 요소가 함께 동작
- 시스템 상태를 얻고 상황을 정확하게 파악



Metrics

무슨 일이 일어나고 있는지

특정 시간 간격으로 모니터링 정보를 측정하고 통계화



Tracing

어디서 일어났는지

서비스의 호출 관계를 흐름도로 보여, 호출의 시작부터 끝까지를 한눈에 볼 수 있음



Logging

무슨 일이 있었는지

발생한 이벤트를 타임스탬프로 기록한 것

Observability 주요 도구



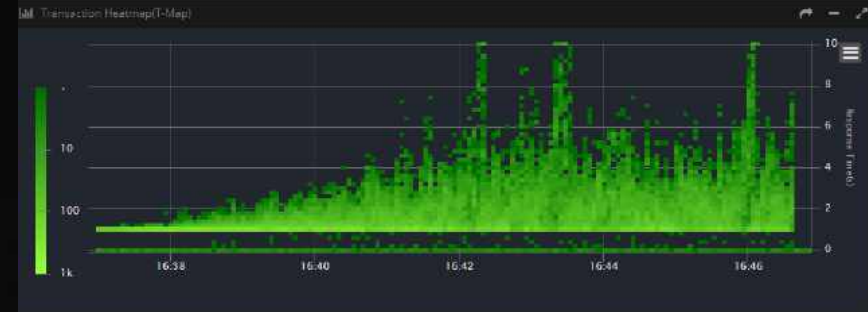
Metrics



Prometheus & Grafana



openmaru
APM



Tracing



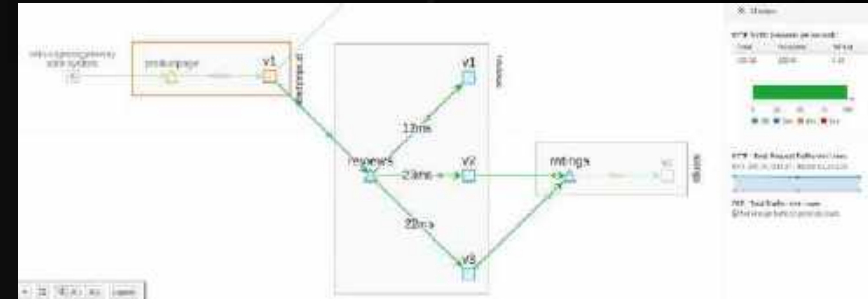
Jaeger



Zipkin



Open Telemetry



Logging



Grafana Loki



EFK



Application Performance Management

감사합니다.



openmaru
APM