

IaaS에서 PaaS로의 고도화 여정

OM사업본부 김호성 차장
hskim@mantech.co.kr

당신에게 1분은 어떤 의미입니까?



amazon

거래액 : 134,364 달러



Google™

Search 4,100,000 건

facebook.

CONTENT

330만 건



Message 전송
690만 건

Agenda

- I. 여행의 목적
- II. 사전 조사
- III. 준비물
- IV. IaaS에서 PaaS로의 고도화 여정
- V. 여행 후기...

- ✓ 가상화가 우리에게 안겨 준 혜택
- ✓ 여전히 해결되지 않은 현안과 화두
- ✓ 자동화의 필요성과 준비해야 할 것들
- ✓ 손쉽게 자율형 데이터센터로 고도화 할 수 있는 방안

운영 및 Process 측면

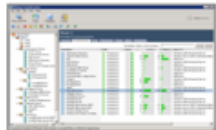
TCO 감소

ROI 증대



빠른 전개

중단 없는 Upgrade



중앙 관리 및 통제



물리적 측면



관리 대상 서버의 감소



전력 소모량 감소



점유 공간 감소



Cable 감소

네트워크



서버



스토리지



Software-Defined
XXX

컴퓨팅 자원
Delivery Time

10배 ↑

서버, 네트워크, 스토리지의
자동화된 프로비전과 확장



엔지니어 & 운영자 관점에서...

- ✓ SW License 비용은 줄었나?
- ✓ App. 관련 작업량이 달라졌는가?
- ✓ 장애 처리 절차나 방법이 바뀌었는가?
- ✓ 소스 커밋 → 빌드 → 배포 과정에서 개발 생산성이 향상되었는가?



개발자 관점에서...

- ✓ 난 개발에만 집중하고 싶은데...
- ✓ 개발 및 테스트 환경 구축에 빼앗기는 시간들
- ✓ 개발, 테스트, 운영 환경 간 호환성 문제 발생
- ✓ 인프라 Level에서 발생하는 문제에는
관여할 수 없는 상황들



주 2회
평균 배포

다양한
요구사항

약 150건
컨텐츠

빠른
변화 속도

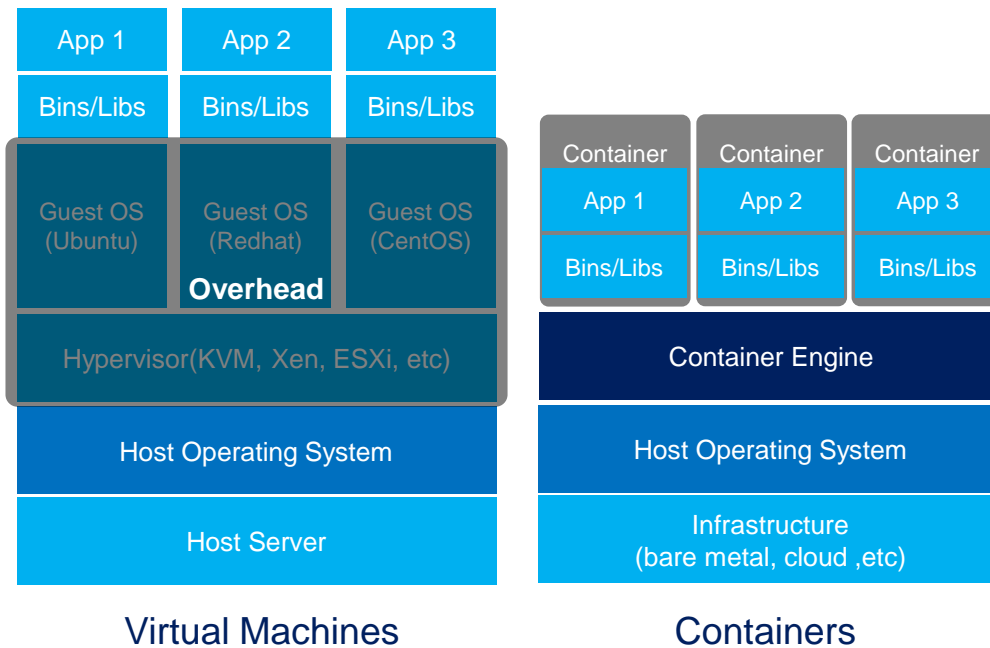


인프라의 확장과 관리는 좋았지만...

- ✓ 하지만, 인프라만 편리해지고
개발자와 SW엔지니어의 **수작업은 계속**
- ✓ 특히, WAS 인스턴스 확장 시
수작업은 무한 반복



- ✓ 필수 아이템 1
- ✓ 필수 아이템 2
- ✓ 추가 요소들



- ✓ VM보다 더 가볍고 뛰어난 성능
- ✓ 부팅시간 : 수초 ~ 수십초 내
- ✓ 커널을 공유하지만 격리 가능
- ✓ 이기종 인프라 환경에서 호환성 이슈 없음 (이식성 우수)
- ✓ 추가적인 OS License 이슈 없음





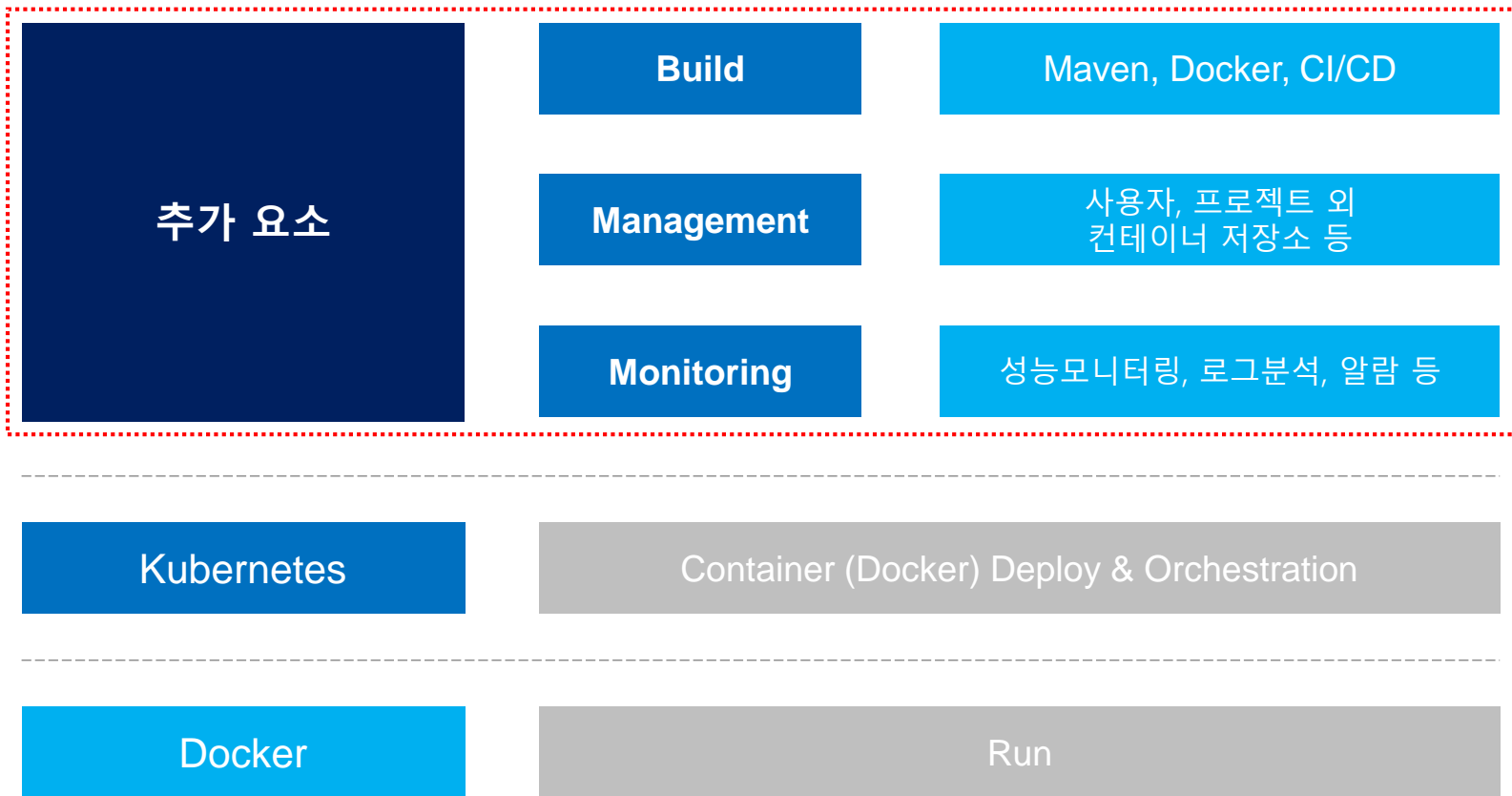
“ 배포, 확장, 관리의 자동화를 통한 생산성 극대화 구현 ”

without Kubernetes



with Kubernetes





배우고, 배우고, 배우자...

- ✓ Docker를 또 배워야 하는가?
- ✓ 미들웨어 & 앱의 컨테이너 化 or 禍
- ✓ IaaS와의 연계를 위한 개발 필요한가?
- ✓ 추가 Stack 구성에만 몇 개월이 걸릴 듯...



IV. IaaS에서 PaaS로의 고도화 여정

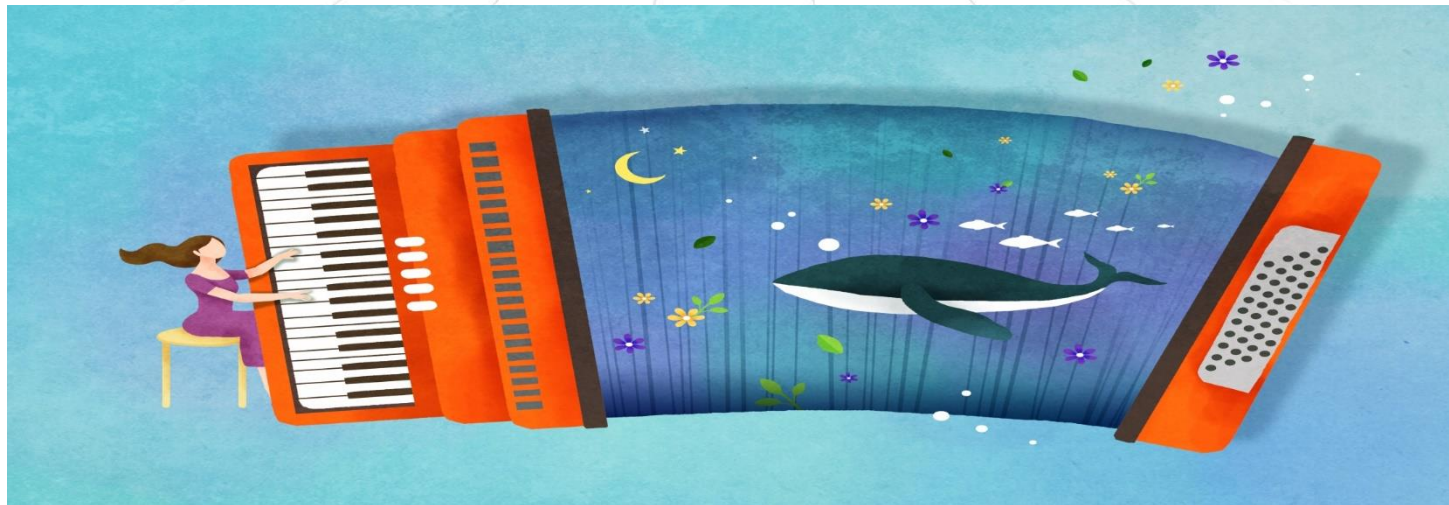
- ✓ 프로젝트 아코디언
- ✓ 아코디언 이란?
- ✓ 아코디언의 주요 기능

“ Kubernetes와 Docker 기반의 Container Platform ”

서비스 중단 없이
요구 사항에 즉시 배포 & 확장



미들웨어, 인프라 관리 등
Management



※ www.accordions.co.kr 에서 다운로드 가능



- 쉬운 설치 및 구성
- 수 분내, App. 배포
- 원클릭 인스턴스 확장



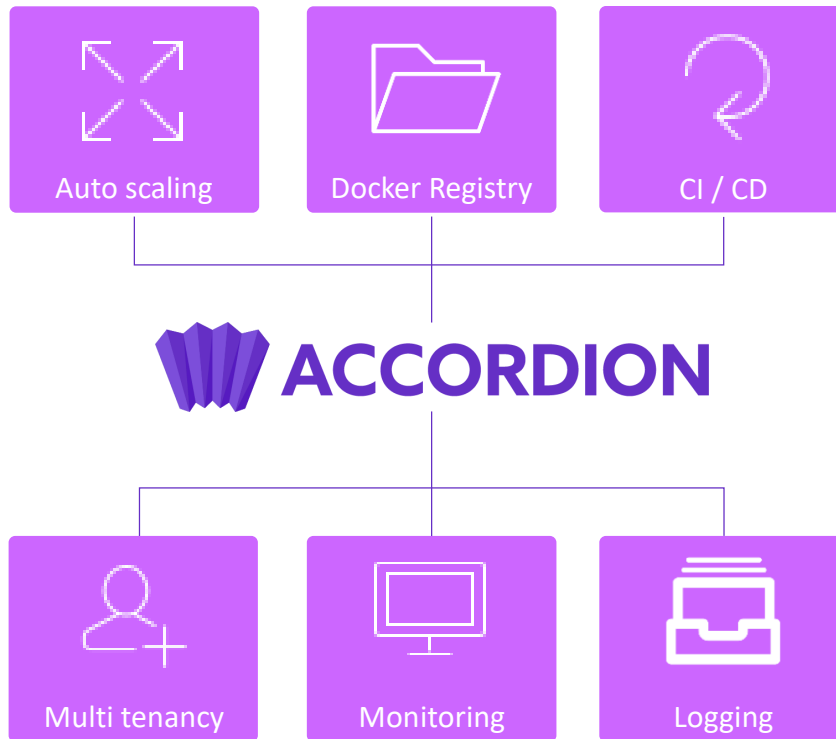
- 롤링 업그레이드/롤백
- Health check
- 모니터링 & 로그분석



- Auto-scaling
- CI/CD 통한 자동 배포
- 자동 N/W 및 L/B

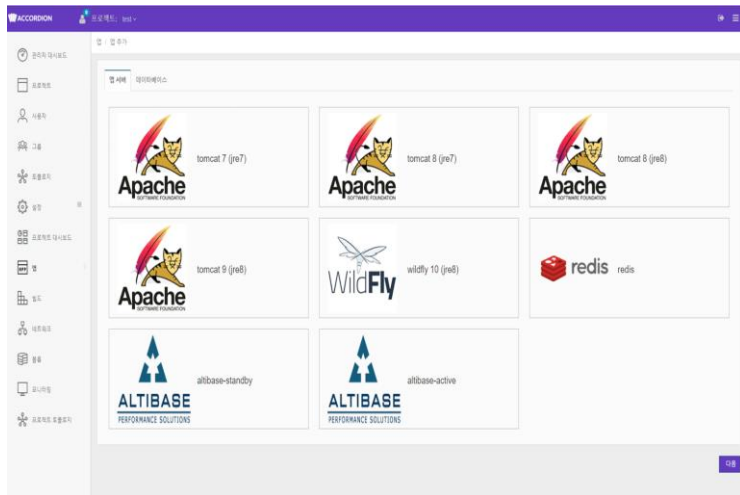


- Bare-Metal 환경
- Private/Public 클라우드
- 플랫폼간 이식성 제공



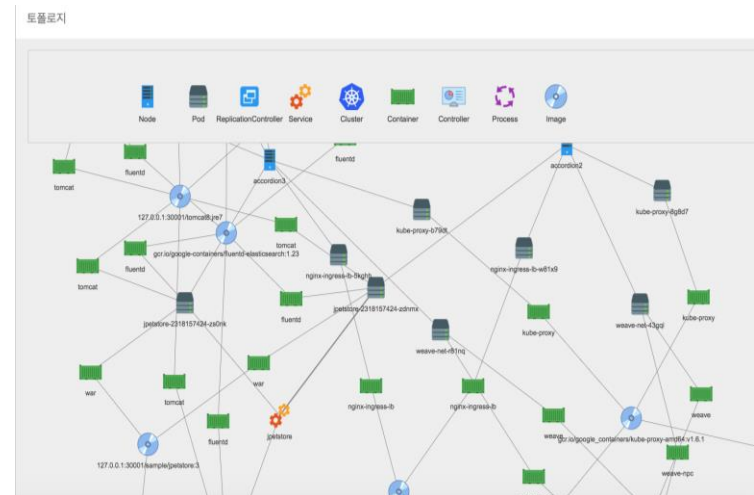
- ✓ All in One
- ✓ One stop service
- ✓ 설치와 구축 단 15분
- ✓ 기존 IaaS 호환
- ✓ Cloud ready

UI를 통한 손쉬운 관리



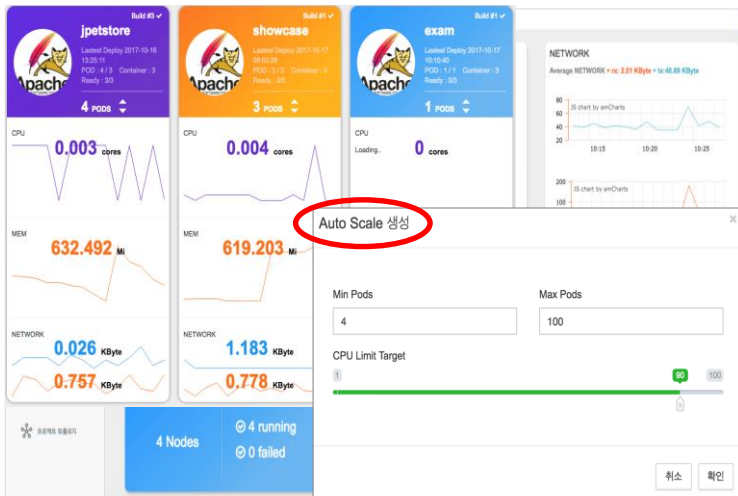
- 프로젝트 및 App. 등록 & 배포 관리
- 사용자 정의 App. 추가 및 관리

Topology View



- 컨테이너 확장 및 신규 업무 추가 시,
업무별 담당자는 토폴로지 뷰에서 맵 확인

자동 확장 (Auto Scaling)



- CPU 부하 시, 인스턴스 자동 확장 및 축소
- 자원 사용률에 따른 동적배치 & Sticky session

Target 설정 (Private / Public)

The screenshot displays the AWS Management Console interface for configuring the 'exam' instance. It shows the 'AWS 환경설정' (AWS Environment Settings) and 'VM 환경설정' (VM Environment Settings) sections. The 'AWS 환경설정' section includes fields for Access Key, Secret Key, Subnet Ids, Region, and IP Address. The 'VM 환경설정' section includes fields for Model, Image, Hostname, IP Address, and Volume Size. The '노드추가' (Add Node) button is highlighted with a red circle.

- 물리적 자원의 한계로 인한 확장 방안 제시
- 다른 플랫폼 & 클라우드로 이식 및 확장 가능

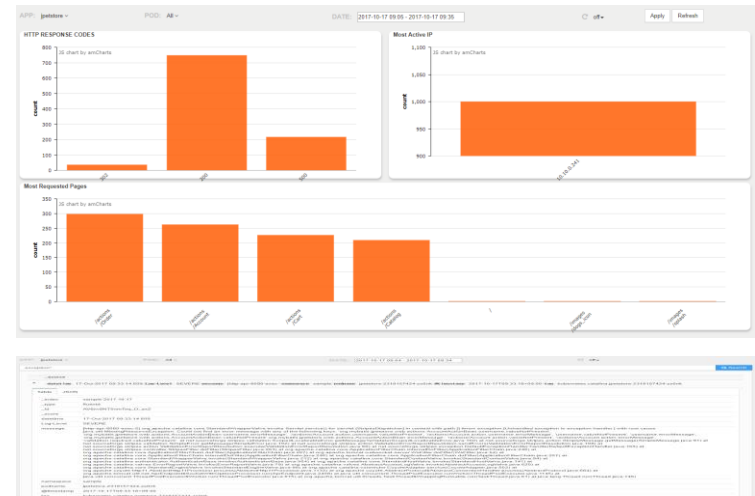
개발 과정에서 APM 망놓고 쓰기 IV. IaaS에서 PaaS로의 고도화 여정

Container 모니터링

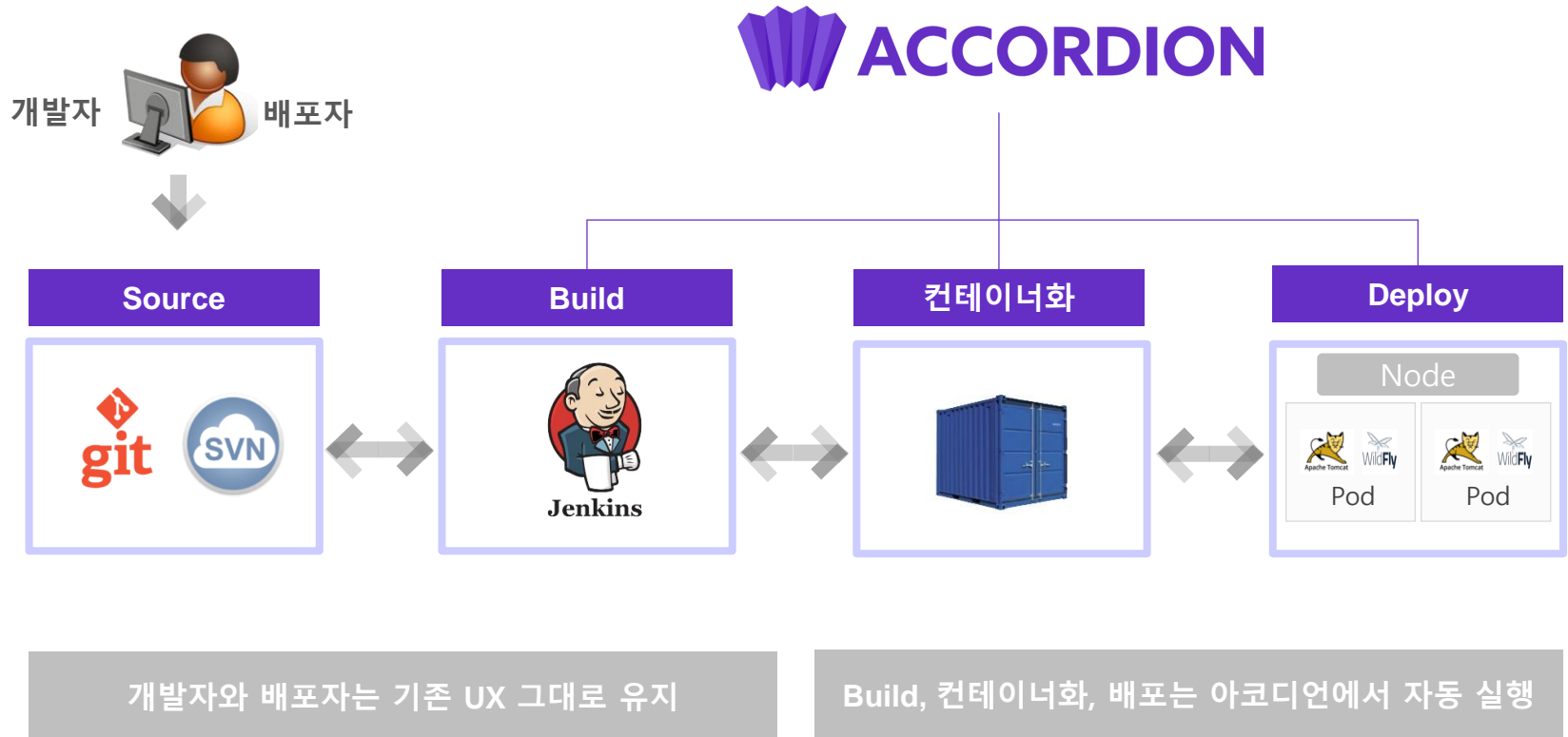


- X-Log / Active Service / GC count 및 Time
- Heap Used / Error Rate 및 TPS

로그 분석 및 검색 기능



- Access Log를 시각화 → 서비스 품질 분석
- 로그 검색 기능을 통한 빠른 분석 제공



기존 IaaS를 쉽고, 빠르게 PaaS로 업그레이드~!



모든 것을 한 번에 해결하려는 생각은 No! No!



인프라 관점에서의 자동화 → Application 관점의 자동화



기술의 변화 속도 만큼 고객의 needs도 빠르게 변화

