

# JAVA TROUBLESHOOTING



**Application Performance Management**

**Cloud Ready System**

**KHAN**  
[ a p m ]

# 비즈니스 측면에서 IT 과제들

다른 기업들은 비용절감을 위하여 클라우드와 오픈 소스를 적극 검토 중이라고 하던데..  
우리 IT 부서는 어떤 것을 준비하고 계신가요?



- 가상화/
- 클라우드/
- 오픈 소스

새로 오신 전무님은 맥북을 쓰시던데,  
우리 회사 IT는 인터넷 익스플로러만 되나요?  
모바일에서 결재할 수는 없나요?



- 모바일
- 크로스 브라우저
- 멀티 OS

증권거래소와 증권사의 거래시스템이 모두 리눅스라고 하는데,  
아직도 안정성이라는 이유로 고가의 유닉스를 도입해야 하나요?  
X86 서버에 리눅스인데, 고가의 상용 소프트웨어를 구매해야 하나요?



- IT Commodity

우리 회사 쇼핑몰이 1등인데, 웹사이트 속도는 왜 3 등이지?  
시스템 성능이 곧 비즈니스 성능이라고 하던데 ...  
~~시스템 장애가 곧 매출 장애 라 매우 민감한데.....~~



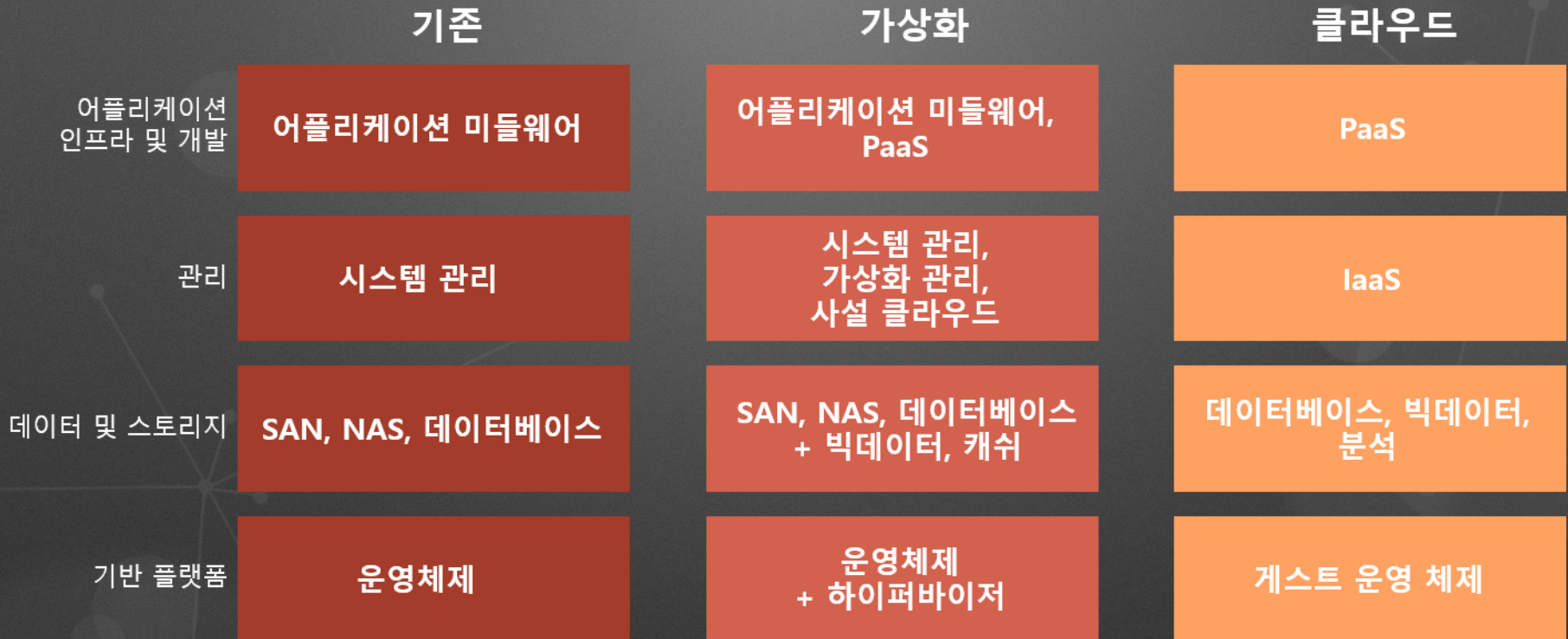
- 지속적인 서비스 품질과
- 성능 개선

▶ ⏮ 🔊 1:50 / 1:56



<https://www.youtube.com/watch?v=WbL9Cb3FYvg&list=PLaFPOkYzLL-9iG9YcG9XuORBc088IXH1ww&index=2>

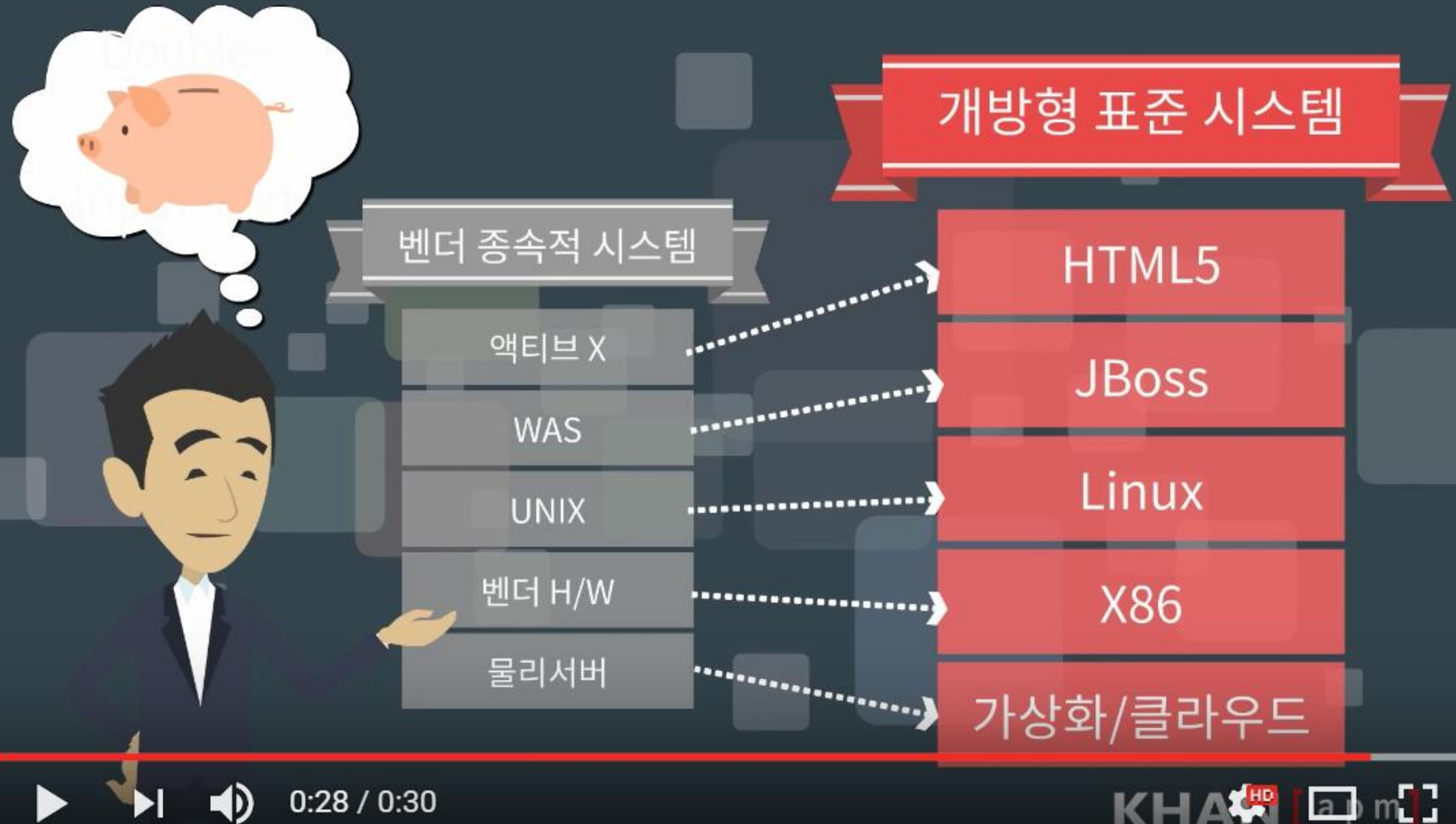
# IT 인프라 변화



시간경과에 따른 기술 채택:



# 업무 시스템 현대화



<https://www.youtube.com/watch?v=iuMx7AVdSyg&index=3&list=PLaFP0kYz1L-9iGYcG9XuORBc088IXH1ww>

# IT 인프라 선택의 고민

|                         |                      |                          |                          |
|-------------------------|----------------------|--------------------------|--------------------------|
| Active-X                | Active-X             | HTML5                    | HTML5                    |
| MS Internet Explorer    | MS Internet Explorer | Chrome/IE/Firefox/Safari | Chrome/IE/Firefox/Safari |
| X-Internet              | X-Internet           | Sencha                   | Extjs                    |
| Commercial Framework    | Commercial Framework | Spring Framework         | Spring Framework         |
| JEUS/WebLogic/WebSphere | WebSphere            | JBoss                    | Wildfly/Tomcat           |
| Oracle/DB2/MS SQL       | Oracle 9i            | MariaDB/PPAS             | MariaDB/PostgreSQL       |
| HP-UX/AIX/Solaris       | AIX                  | Red Hat Linux            | CentOS                   |
| PA-RISC/Power/SPARC     | Power                | X86                      | X86                      |

독점  
소프트웨어

현재 구성

유상  
오픈소스

무상  
오픈소스

저비용 고효율

고비용 저효율  
0:23 / 0:27

물리 환경

Private/Public

Cloud  
Khan (apm)

<https://www.youtube.com/watch?v=ILl4gBwEGzg&list=PLaFPOkYzLL-9IGYcG9XuORBc088IXH1ww&index=1>

# 업무 시스템 세계화와 현대화 -상세



## 적용방안

- 인프라 (서버/OS/미들웨어) 를 단일 벤더 지원
- Cross-Platform과 Cross Browsing을 지원
- 각 부분별로 개방형 표준 환경으로 전환
- 오픈 소스 프레임워크를 통한 개발
- 라이선스 비용 없기 때문에 초기 도입 비용 절감

## 기대효과

- 특정 서버 벤더와 기술에 얽매이지 않는 환경
- Cost Performance 가 좋은 인프라를 선택
- 서비스 품질이 높은 개발/운영 업체 선택
- 운영 효율성 향상
- 저비용 고효율 시스템 구축

Application Performance Management

# 웹시스템 운영 현실

**KHAN**  
[ a p m ]

# WAS 관리자 - 고민들

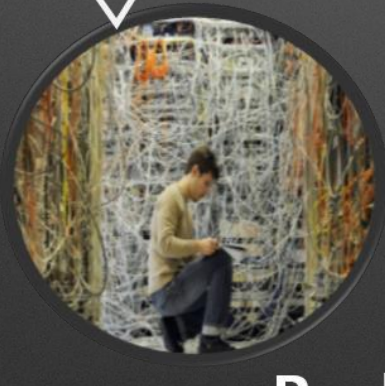
오픈 소스 WAS 익숙하지 않은 데 어떻게 하지?

WAS에서  
세션 클러스터링이  
되지 않네, 헐!!!

매번 비슷한 작업인데  
... 언제 이 많은 서버를  
다 설치하지??!!!

서비스 오픈이 이번 주  
인데, 튜닝은 어떻게  
해야 할까?

아무리 구글을 뒤져봐  
도 어떻게 해야 하는  
건지 모르겠어!!!!



Real Voice

# 시스템 구축 전 - 고민들



비즈니스  
담당자

- ✓ **오픈 소스**는 처음이라서 **걱정**이 되는데요?
- ✓ 혹시 **응답속도가 느리지는 않조?**
- ✓ 매번 **전문가에게 도움**을 받아야 하는 건가요?
- ✓ 이번 주가 오픈 인데 아직도 **구성 문제**로 테스트를 못하고 있네요.

- ✓ **개발도 하고 서버 구성도** 해야 하고 매일 야근이네요.
- ✓ 기존 쓰던 제품은 알겠는데 **오픈 소스 제품은 처음이라 모르겠어요.**
- ✓ 오픈 소스라서 **네이버나 구글을 검색**해도 무엇인지 모르겠어요.
- ✓ **문서 없으세요? 문서 좀 주세요.**



애플리케이션  
개발자

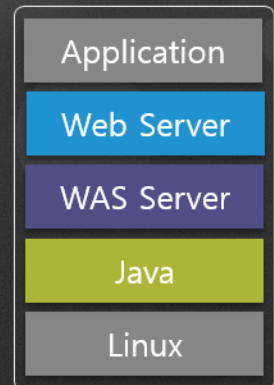
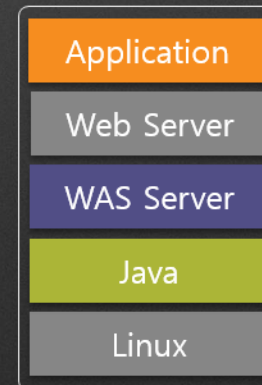
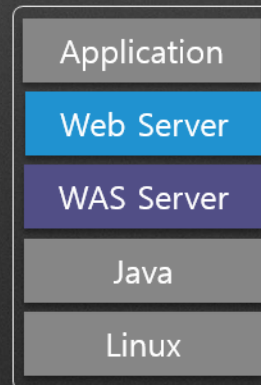
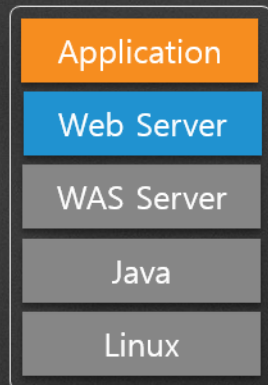
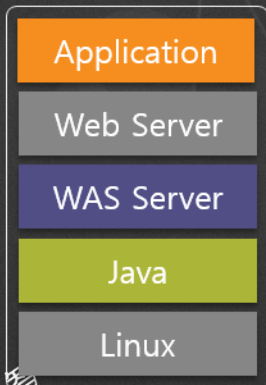
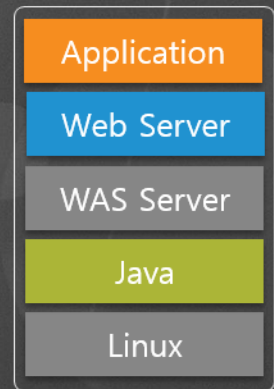
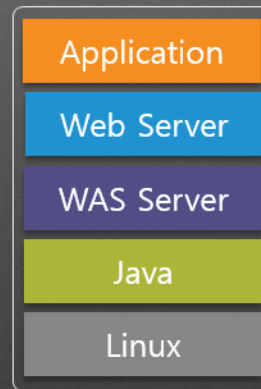
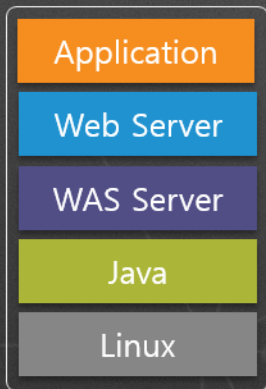


WAS운영자

- ✓ 이 많은 서버를 **언제다 설치**하지요?
- ✓ **오픈 소스가 익숙하지 않아서** 웹시스템 구축이 엄두가 나지 않아요.
- ✓ 웹서버와 WAS 구성을 **저희 회사에 맞게 구성**해 주세요.
- ✓ 안정된 서비스를 위한 **최적화 및 튜닝 값**을 적용해 주세요.
- ✓ **세션 클러스터링**을 해야 하는 데 어떻게 해야 하는 건가요?

# WAS 관리자 - 운영 환경 표준화 이전

## A NON Standard Operating Environment



# 시스템 구축 후 - 고민들



- ✓ 오늘 온라인방문자 수가 급격히 줄었는데요?
- ✓ 다음달에 온라인 광고를 해야 하는데 문제는 없겠죠?
- ✓ 주기적으로 장애가 발생하는데 대책은 없나요?
- ✓ 사용자가 조금 만 늘어도 시스템이 다운이니 불안해서 뭘 할 수가 없네요.

- ✓ 확인해 봤는데 소스는 문제가 없어요. WAS 문제 아닌가요?
- ✓ 느린 페이지에서 사용하는 DB 쿼리를 보고 싶는데요.
- ✓ 개발할 땐 빠르는데 운영시스템에서만 느린 이유가 무엇인가요?



- ✓ 서비스가 갑자기 느려졌는데 WAS 문제일까요?
- ✓ 실시간 동시 사용자수나 TPS 등의 기본 운영 정보를 알고 싶는데요.
- ✓ 서비스가 느려질 때 WAS를 Restart 하면 정상적으로 동작해요.
- ✓ 특정페이지 나 서비스가 느린 이유는 무엇일까요?



0:28 / 0:28



[https://www.youtube.com/watch?v=jYjGoNEV-\\_E&index=12&list=PLaFPOkYzLL-QWKsbT40cxwr\\_-cijeU6L](https://www.youtube.com/watch?v=jYjGoNEV-_E&index=12&list=PLaFPOkYzLL-QWKsbT40cxwr_-cijeU6L)

# 고객이 체험하는 것은?



# 엔터프라이즈 모니터링에서는 문제 없음

네트워크, 서버 모두 잘 운영되고 있습니다.  
심지어 어제보다도 성능이 좋은데요.



[https://www.youtube.com/watch?v=eE7rPuksd1c&index=7&list=PLaFPOkYzLL-QWKsbT40cxwr\\_-cijeU6L](https://www.youtube.com/watch?v=eE7rPuksd1c&index=7&list=PLaFPOkYzLL-QWKsbT40cxwr_-cijeU6L)

# 각 분야에서는 전문가 – 코끼리 만지기

KHAN [ a p m ]



0:48 / 0:51



[https://www.youtube.com/watch?v=P5S0gl-4hZI&list=PLaFPOkYzLL-QWKsbT40cxwr\\_-cijeU6L&index=10](https://www.youtube.com/watch?v=P5S0gl-4hZI&list=PLaFPOkYzLL-QWKsbT40cxwr_-cijeU6L&index=10)

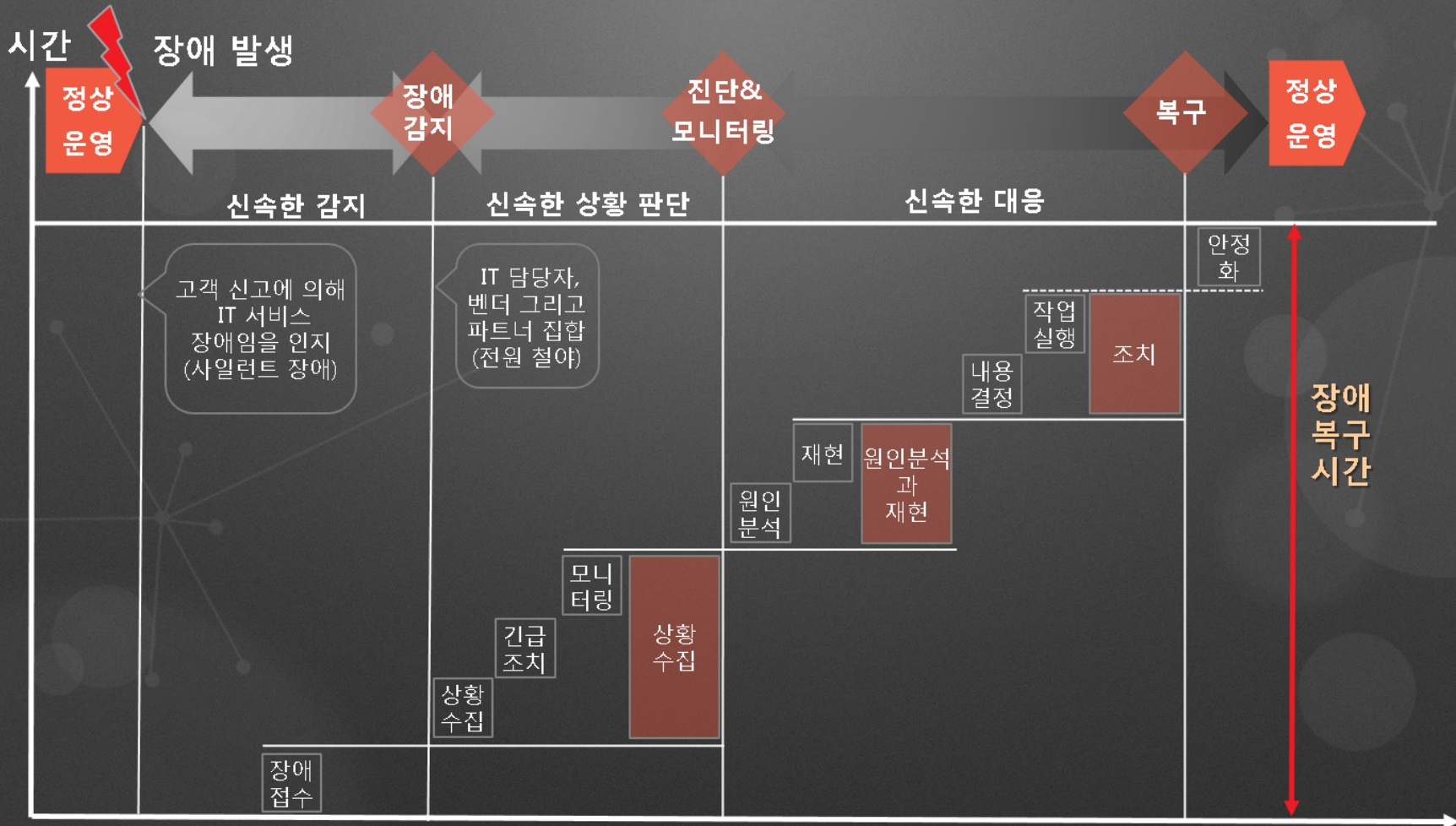
Application Performance Management

# Java Troubleshooting 프로세스

**KHAN**  
[ a p m ]

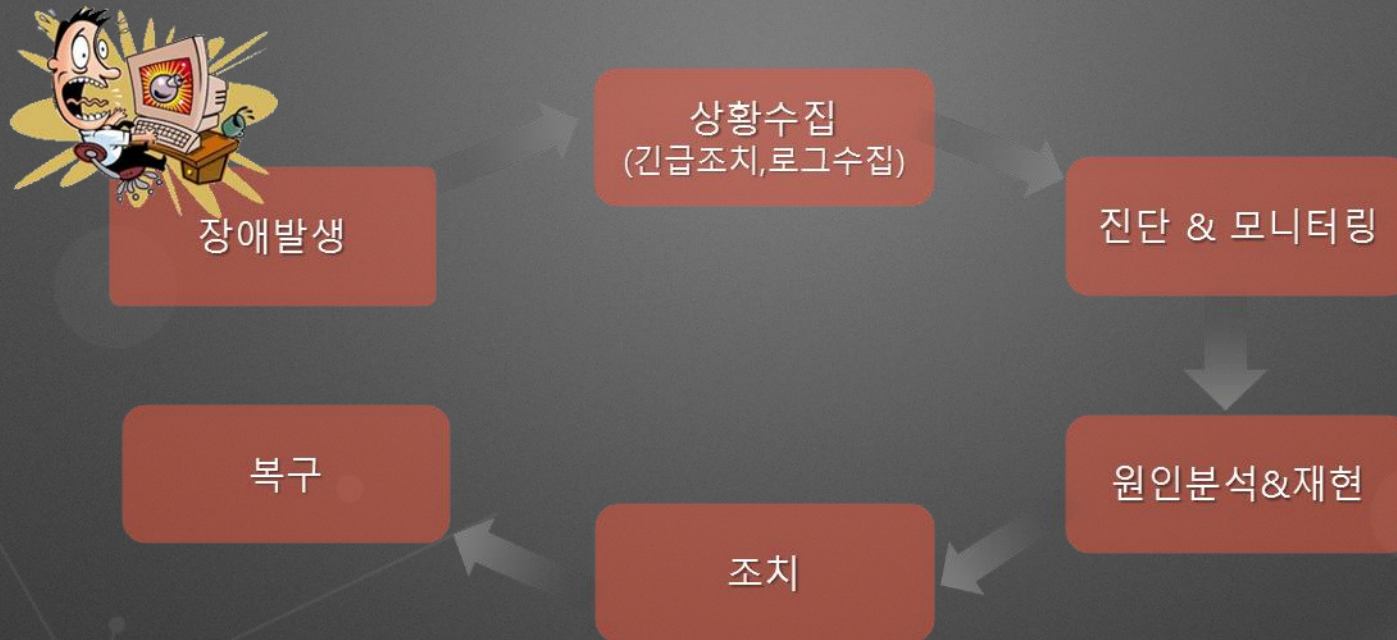
# Incident & Problem Management

- 장애복구시간을 단축할 수 있는 도구는 ?



장애 복구 과정 ( Incident Lifecycle Management)

# Troubleshooting 공통 프로세스



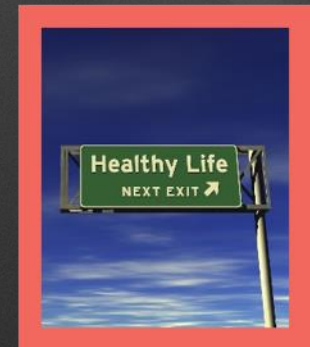
증상과 원인



진단 및 모니터링



조치



방지대책

# 증상과 원인 - 현황 파악

- 기본 정보 파악

- JBoss EAP 버전
- OS 종류/버전
- JVM 종류/ 버전
- 데이터베이스 종류/ 버전
- etc...



증상과 원인

- 설정 파일 파악

- JBoss 설정 파일(domain.xml, host.xml, standalone.xml 등 )
- JBoss 시작 스크립트
- etc...

# 증상과 원인 - 장애 내용 파악

- 장애 내용 및 상황 확인
  - 장애 발생 과정
  - 시스템 정지 여부
  - 장애 발생 빈도
  - 장애 발생 조건
  - 장애 재현 가능 여부
  - 장애 재현 절차
  - 회피 방법 유무



증상과 원인

# 증상과 원인 - 수집 자료는?

- 기본 수집 자료
  - 로그!
  - Thread dump!
  - 기본 환경 설정 정보
- 관련된 상황은?
  - 언제
  - 어떤 서버, 어느 인스턴스에서 정확한 OS, JVM, WAS 버전은?
  - 발생 주기는?
  - 재현은 되는지?
- 어떤 영향을 미치는지?
  - Business 부서의 영향은?
- 관련된 네트워크, H/W 및 S/W는 어떤 것인지?
  - 대상 시스템에 로그나 상황은 어떤지 체크



증상과 원인

# 진단 및 모니터링 - 정보수집

- 기본 정보의 수집
  - 종류 / 버전 등 정보...
- 설정 정보의 수집
  - 설정파일, 스크립트 등
- 로그 정보의 수집
  - boot.log
  - server.log
  - access.log      etc...



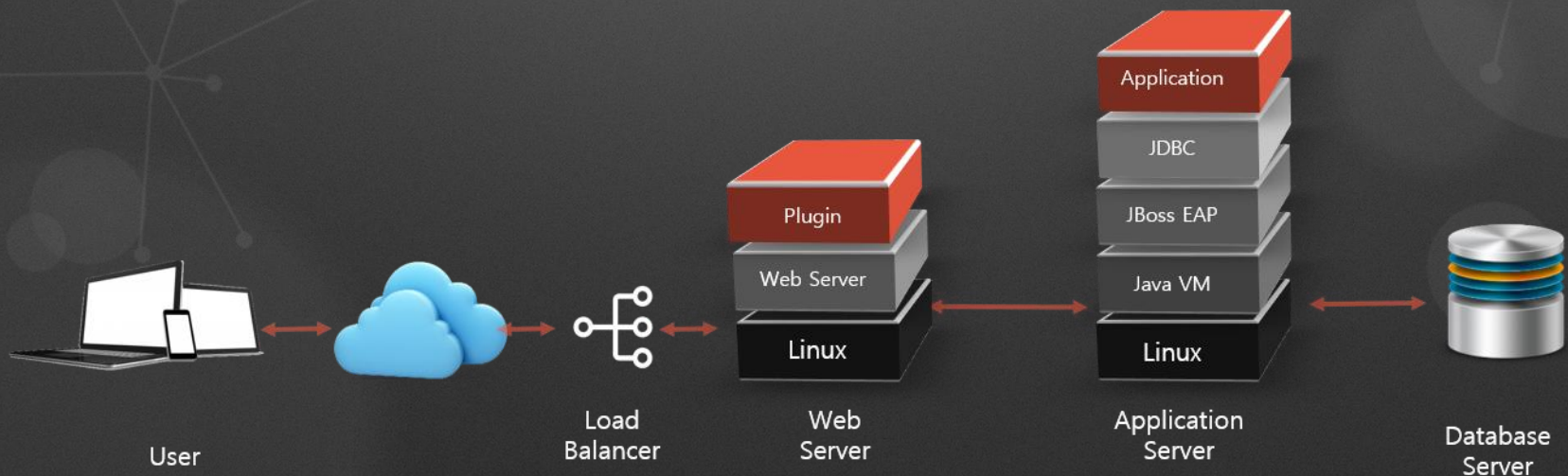
진단 및 모니터링

# 진단 및 모니터링 - 문제 구간 식별

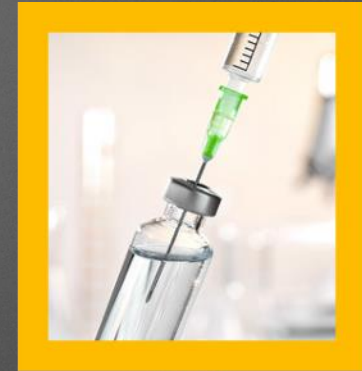
- 문제가 발생 구간 파악
  - Application
  - JBoss
  - Database
  - Web Server
  - JVM
  - OS
  - H/W



진단 및 모니터링



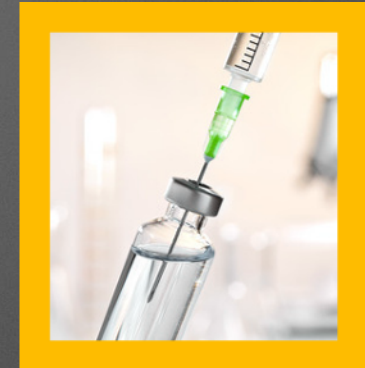
- 조치 내용
  - 설정 변경
  - 어플리케이션 수정
  - 버전업
  - Patch
  - etc...
- 조사 방법
  - 재현 스텝
  - 디버그 정보



조치

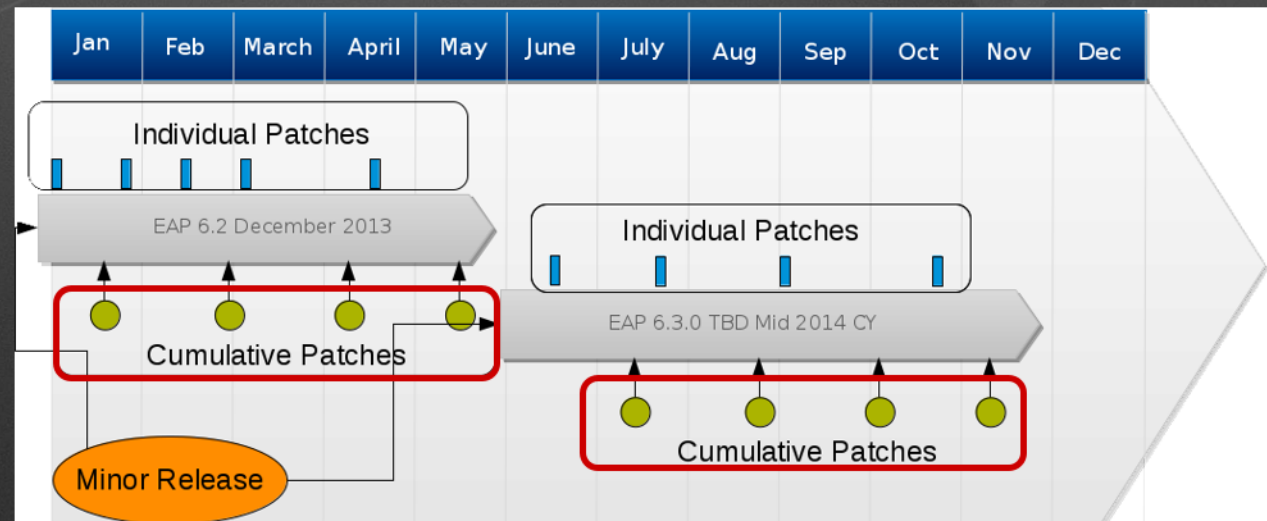
# 조치 - Patch

- 패치 업데이트는 ZIP 및 EAP의 설치 기반의 RPM 형식으로 전달되며, zip 형태의 패치 파일은 JBoss CLI 의 Patch 명령어를 통해 적용
- 적은 개별 패치들은 관리자로 하여금 관리 부담 감소
- 심각도가 높고 시간이 급한 케이스에 대한 버그 픽스는 기존 방법대로 개별 패치들로 배포



조치

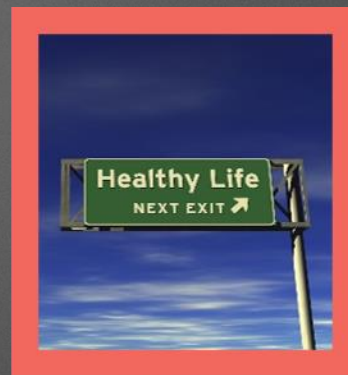
- 여러 수정사항을 포함한 누적패치는 함께 테스트 되어 제품의 질을 향상시키고, 위험성을 감소



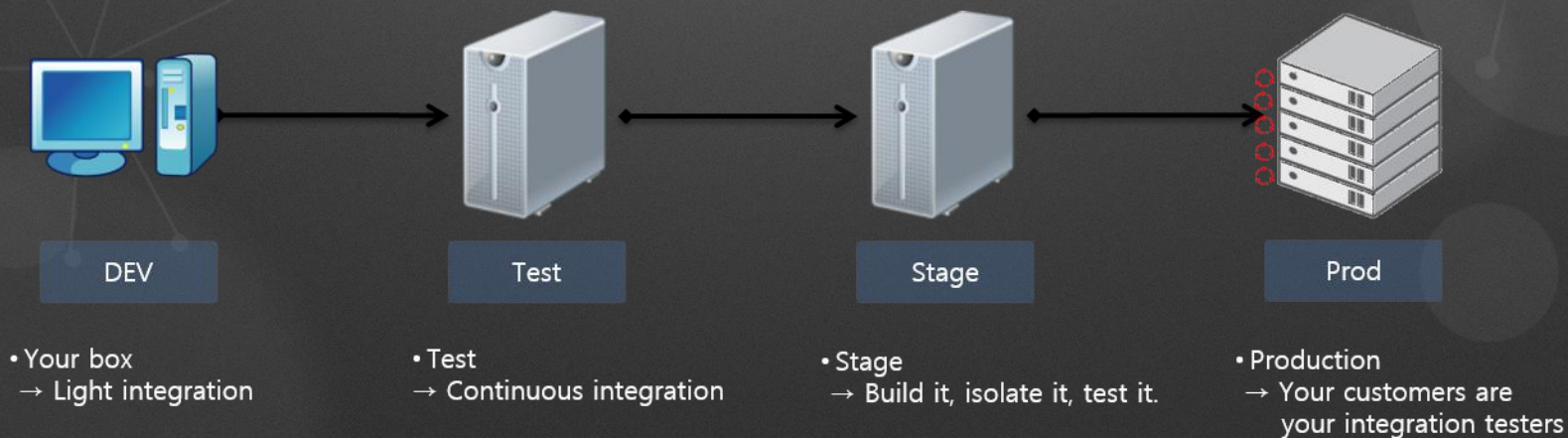
Source : EAP 6.2 이상부터 유지관리 배포방식 변경 -  
<https://access.redhat.com/site/ko/articles/645133>

# 방지 대책

- 안정된 코드 테스트를 애플리케이션과 검증을 위한 환경 구축
- 서비스 품질을 높이기 위한 기본 프레임워크
- Subversion (SVN) 또는 GitHub
- 빈번한 커밋과 자동화된 빌드와 자동 피드백
- 테스트 프레임워크 적용



방지대책



The OpenNaru logo is a stylized, three-dimensional wooden symbol resembling a continuous loop or a calligraphic 'S' shape, mounted on a dark wood-grain surface.

opennaru

# Java Troubleshooting 개요

# 장애나 성능 이슈 - 즉각적인 확인이나 조치가 가능한가요?



[https://www.youtube.com/watch?v=vnFkLtxOHPY&list=PLaFPOkYzLL-QWKsbT40cxwr\\_-cijeU6L&index=11](https://www.youtube.com/watch?v=vnFkLtxOHPY&list=PLaFPOkYzLL-QWKsbT40cxwr_-cijeU6L&index=11)

# 주요 Troubleshooting

- WAS 주요 Troubleshooting case
  - WAS의 hang 상태로 인하여 응답이 없는 상태
  - WAS Crash 로 인하여 core 발생
  - 데이터베이스로부터의 응답속도가 낮아짐
  - CPU 사용률이 100%되어 응답이 없음
  - Java Heap Memory 사용량이 증가됨

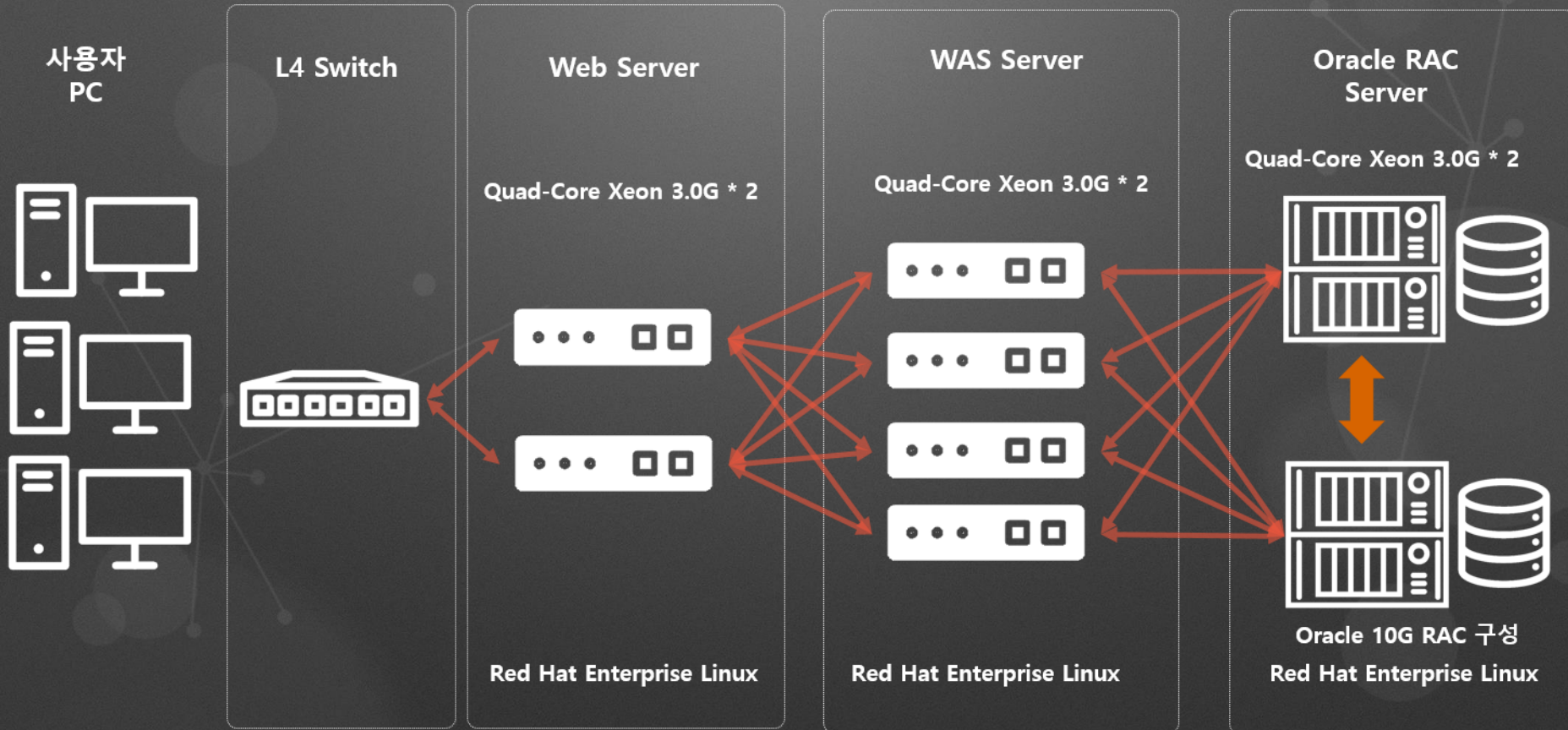


Too many  
Open files

OutOfMemoryError  
(heap,perm...)

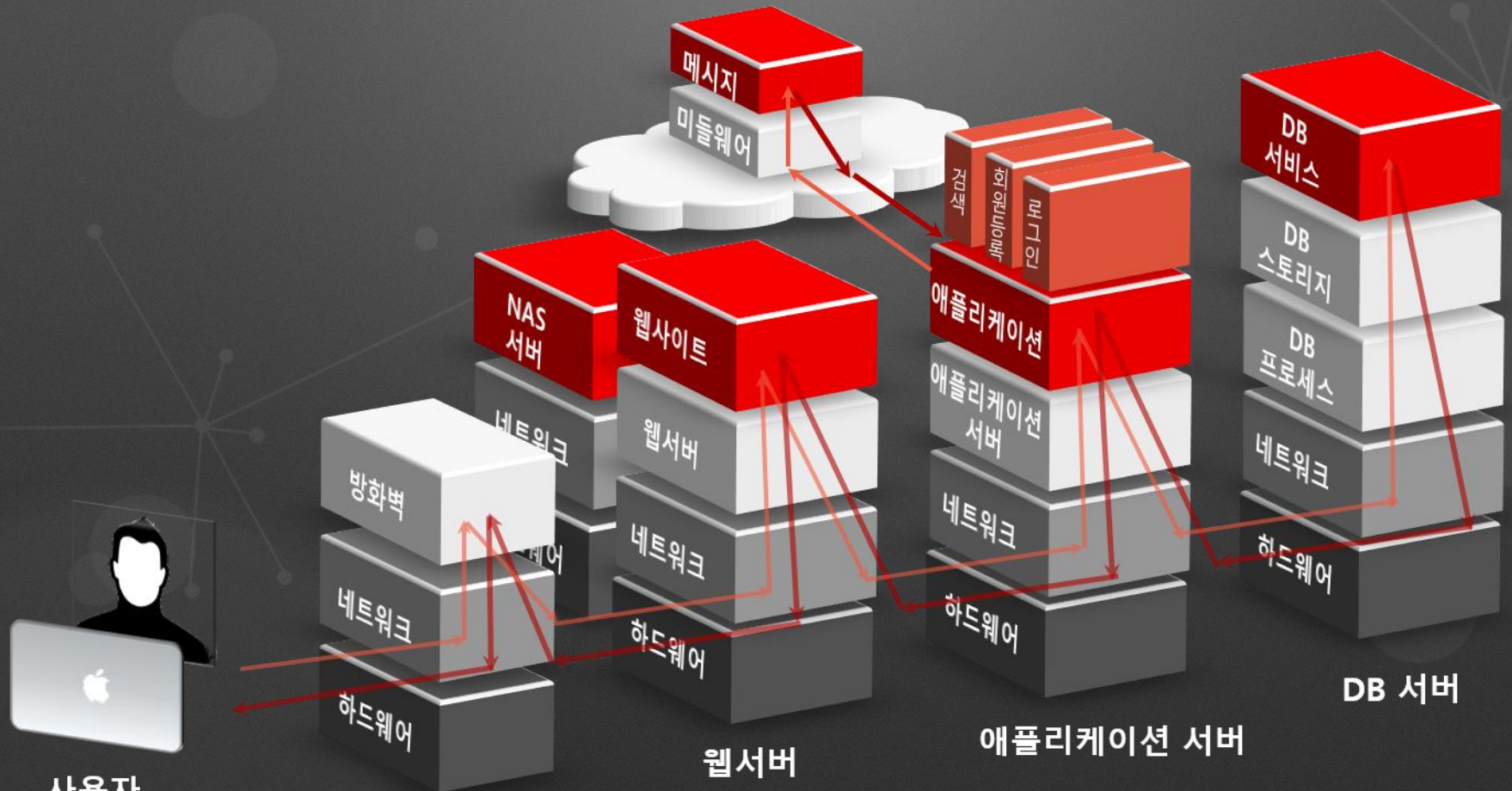
Slow Down/  
Timeout

# 웹시스템의 하드웨어 구성



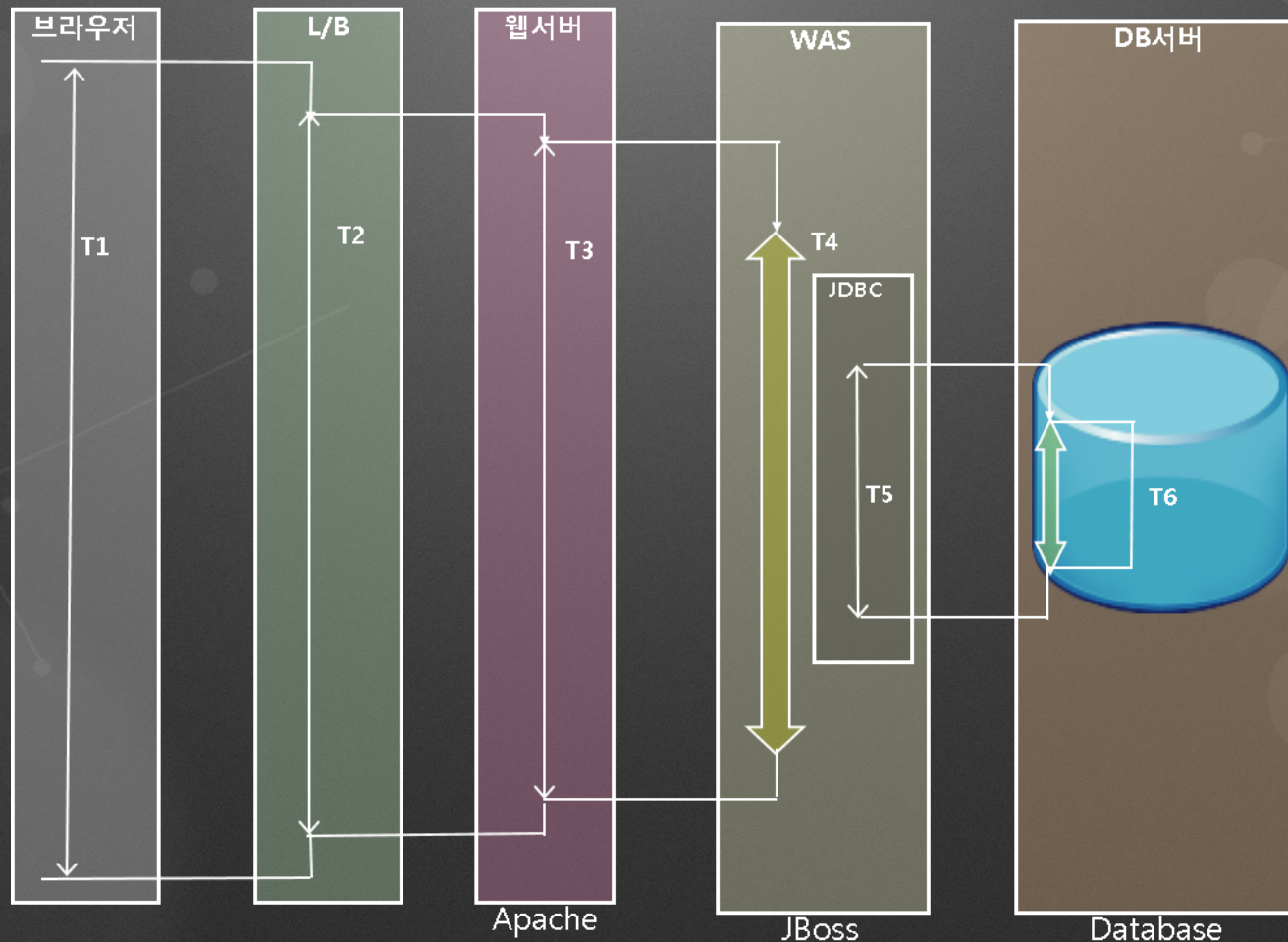
# 미들웨어는 시스템 장애의 관문이자 시작점

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

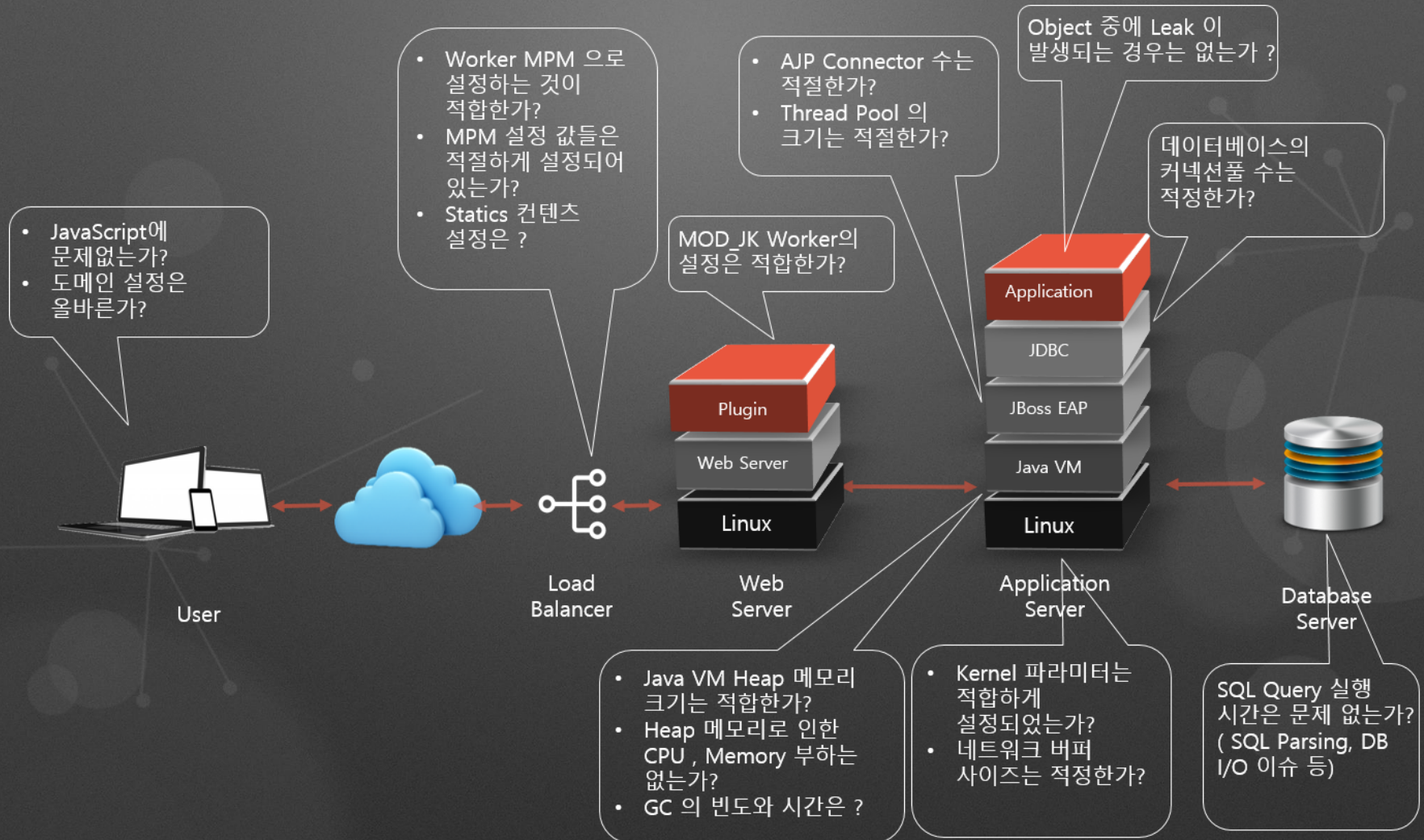


# 웹시스템 Request/Reply 구조

- $T1 > T2 > T3 > T4 > T5 > T6$
- WAS 는 Thread pool과 JDBC Connection Pool을 이용해서 성능을 보장



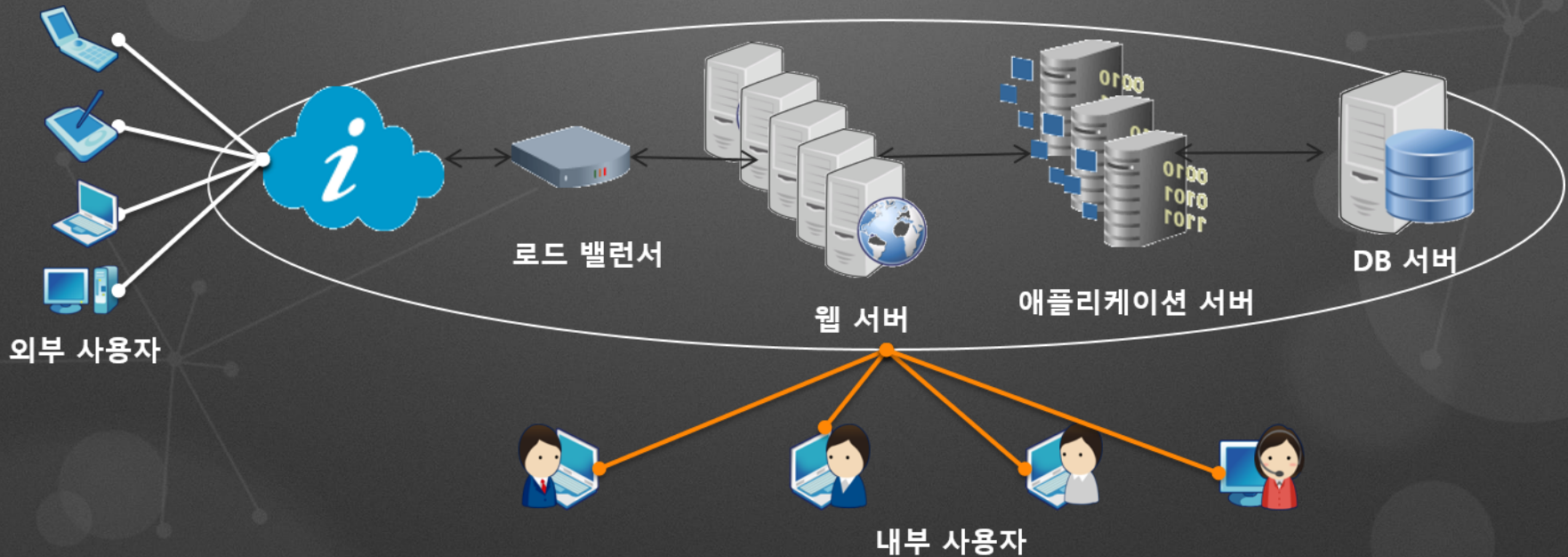
# 구간별 장애 포인트





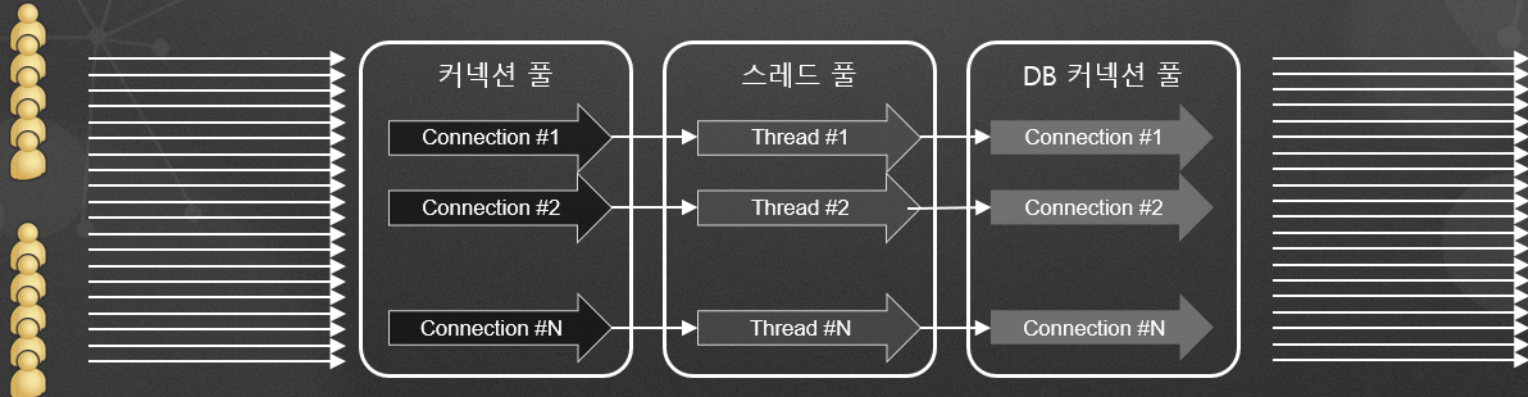
<http://www.brendangregg.com/linuxperf.html> 2015

# 일반적인 웹 시스템의 구성



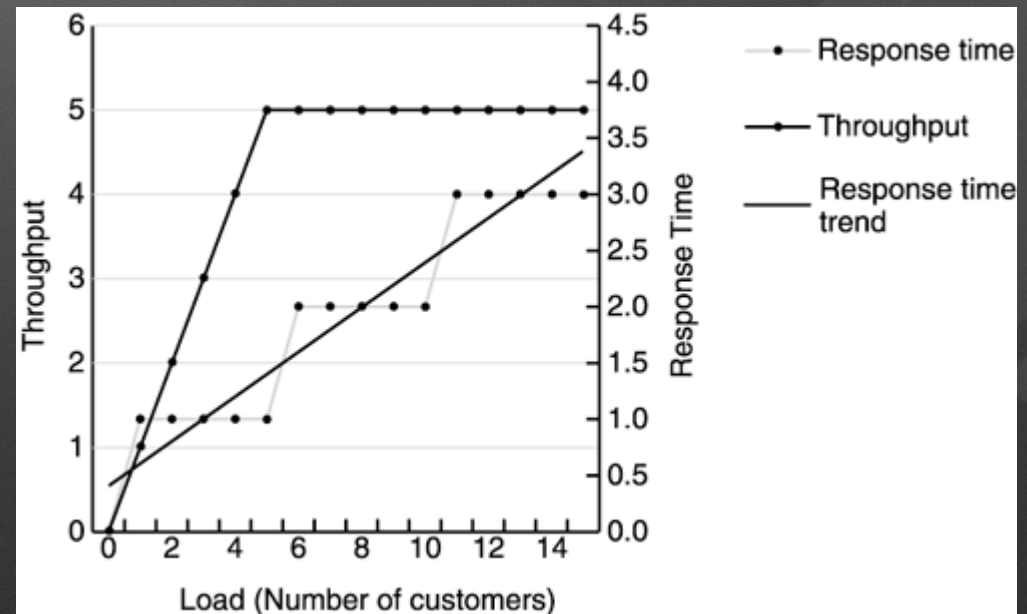
# 주요 시스템 컴포넌트

## 3 티어 구조에서 장애와 성능 문제 발생 시 신속한 진단과 조치



# 주요 용어

- TPS : Transaction Per Second
  - 초당 트랜잭션 처리건수 : Total Number of Transaction
- TPM : Transaction Per Minute
  - 분당 트랜잭션 처리건수
- RPS : Request per Second
  - 초당 요청 건수
- Transaction Response Time
  - 트랜잭션 응답 시간
- Throughput
  - 처리량
- Concurrent User
  - 동시 사용자



# TPS(Transaction Per Seconds)

- TPS → 초당 처리 량

- $L = \lambda W$   $\lambda = \frac{L}{W}$

- $L$  : 상점에 있는 사람들의 평균 → Active User
- $\lambda$  : 상점에 도착하는 비율(단위시간당) → TPS
- $W$  : 사람들이 상점에서 쇼핑하는 게 걸린 평균 시간 → Average Response Time

- $TPS = \frac{\text{Active User}}{\text{Average Response Time}}$

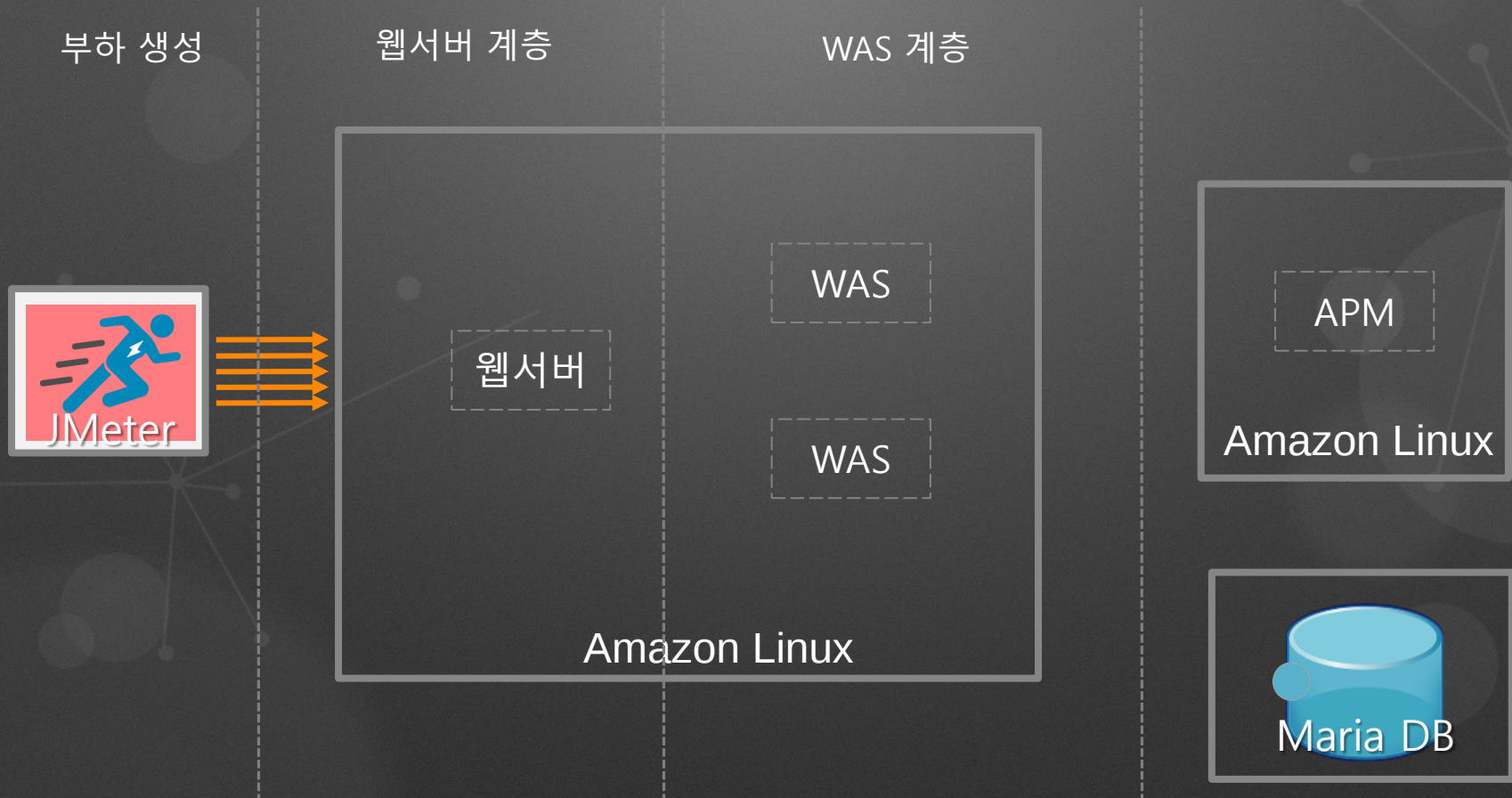
$$\text{Active User} = TPS \times \text{Average Response Time}$$



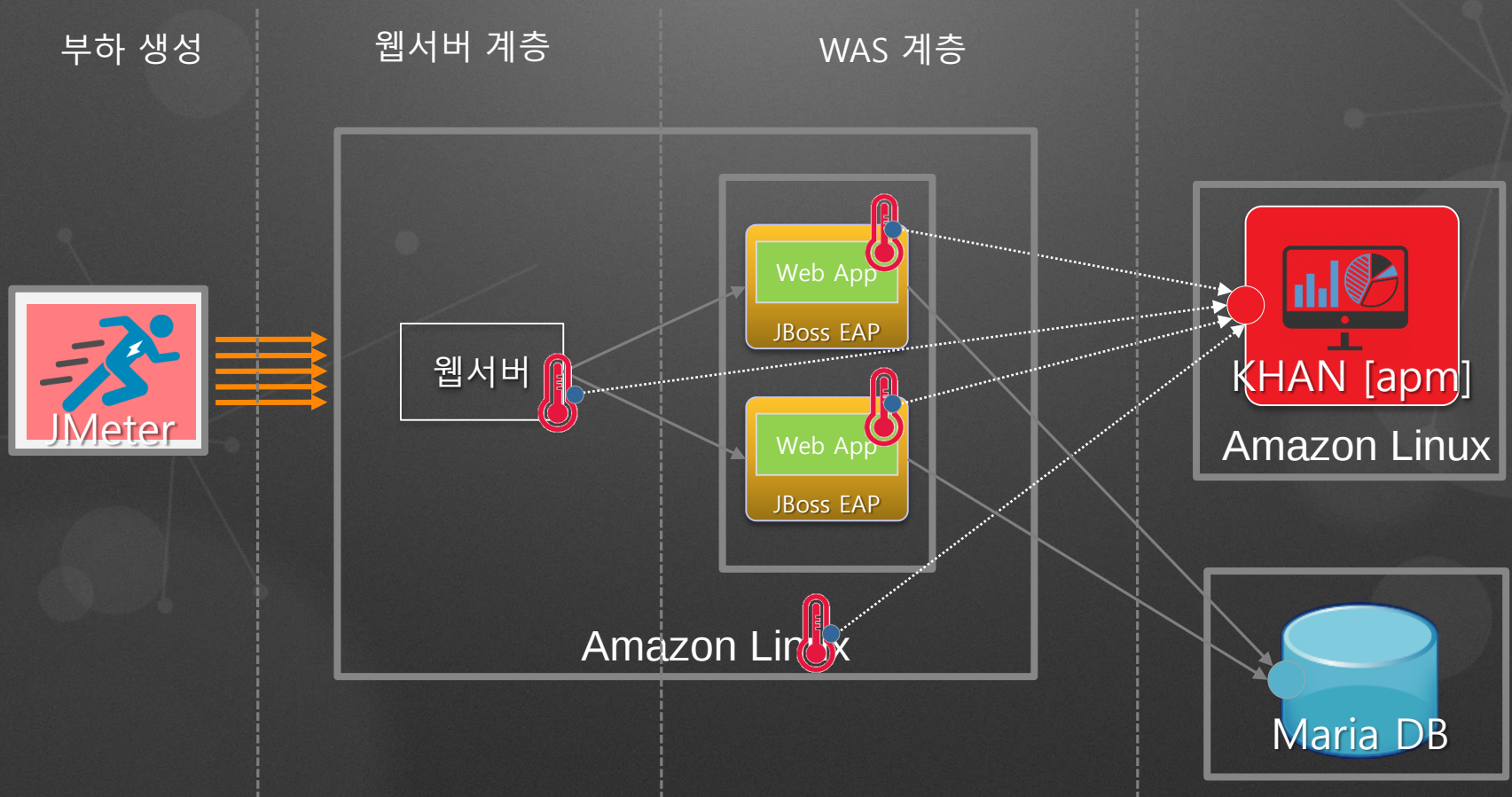


# 시스템 설치/구성/튜닝/모니터링 - DEMO

# DEMO #1 Amazon Linux에 10 분 안에 자동 설치전



# DEMO #1 Amazon Linux에 10 분 안에 자동 설치 - 아키텍처



# KHAN [apm] 제품 개요

# Application Management Challenge



## APM



### 개발팀

- 고성능 애플리케이션 개발
- 신속한 버그 픽스
- 이슈 재발 방지



### QA팀

- 부하테스트 시 성능 보장
- 서비스 오류 감지
- 최소 응답시간 유지



### 운영팀

- 성능에 대한 최적화와 서비스 정상 유/무
- 신속한 장애 감지와 원인 파악 그리고 조치
- Over/Under 사이징인가?



### 비즈니스 담당자

- 고객의 서비스 만족도는 충분한가?
- 고객 이탈 방지 및 브랜드 이미지 손상 방지
- 경쟁사 대비 성능은 충분한가?

# KHAN [apm] = Ah ! Provisioning + Monitoring



| 설치                                                                                               | 구성                                                                                                                              | 튜닝                                                                                   | 모니터링                                                                                             | 진단/조치                                                                                          |
|--------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------|
| <ul style="list-style-type: none"><li>•리눅스만 설치되어 있으면 WAS 와 웹서버 등 웹시스템에 필요한 소프트웨어를 자동설치</li></ul> | <ul style="list-style-type: none"><li>•WAS 클러스터링, 데이터그리드, 필수 운영스크립트 등 난이도 높은 구성을 지원</li><li>•개발자/관리자를 위한 운영 가이드 자동 생성</li></ul> | <ul style="list-style-type: none"><li>•전문가 수준의 튜닝 (리눅스 커널, 웹서버/JVM/WAS 튜닝)</li></ul> | <ul style="list-style-type: none"><li>•시스템과 OS 에 대한 모니터링 뿐만 아니라 Java 나 WAS 까지 포괄적으로 지원</li></ul> | <ul style="list-style-type: none"><li>•Threaddump, 네트워크 상태, 웹서버 상태 등을 진단할 수 있는 도구 제공</li></ul> |



## GUI 기반 웹시스템 설치/구성/튜닝 ( 웹시스템 프로비저닝)

- 웹서버/WAS/Datagrid 에 대한 자동 설치/구성/튜닝 GUI 도구 제공 (Red Hat Linux 계열)
- 입력한 정보를 기준으로 시스템 자동 구성



## Apache /Tomcat/ JBoss 등 오픈 소스에 최적화된 APM

- 리눅스 종류와 버전을 식별하여 해당 OS에 맞는 웹서버/WAS 제품으로 설치
- 리눅스 OS/ Apache /Tomcat / Jboss 에 최적화된 모니터링



## Troubleshooting 을 위한 다양한 분석 도구 제공

- 웹시스템 장애 시 즉각적인 원인 분석을 위한 다양한 도구 제공  
( Thread Dump / Netstat Dump / Isof dump )



## 전문가 수준의 튜닝과 정보 제공

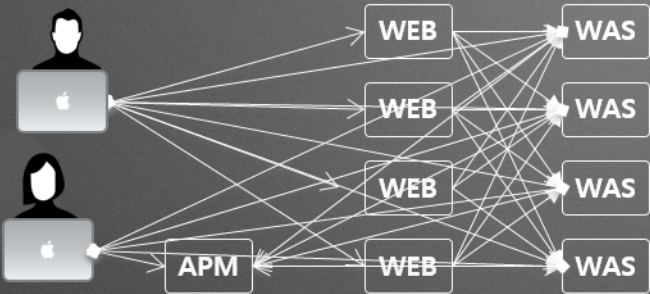
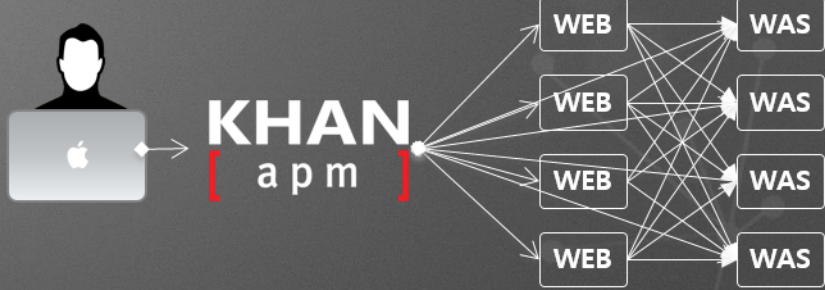
- OS 커널 파라미터나 WAS Thread Pool 이나 데이터소스 Pool을 설정
- JDK, mod\_jk, mod\_cluster , 운영 스크립트 파일 등을 구성 요청에 맞추어 설정하여 설치



## HTML5 인터페이스와 보고서 제공

- 웹브라우저 상에 별도의 Launch 없이 다양한 모니터링 기능 제공
- 웹관리 콘솔 접속 정보 등 웹/WAS 환경 운영시 참조할 수 있는 가이드 제공

# KHAN [apm] vs. Others – 차별화 요소

| 항목    | 수작업에 의한 웹시스템 구축                                                                   |                                                                                                                            | KHAN Provisioning을 통한 웹시스템 구축                                                       |                                                                                                                                        |
|-------|-----------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------|
| 설치 방법 |  |                                                                                                                            |  |                                                                                                                                        |
| 구축 기간 | ●                                                                                 | <ul style="list-style-type: none"> <li>수일 ~ 수 주</li> </ul>                                                                 | ●                                                                                   | <ul style="list-style-type: none"> <li>수 시간 ~ 수일</li> </ul>                                                                            |
| 안정성   | ●                                                                                 | <ul style="list-style-type: none"> <li>담당자의 기술 수준에 따른 품질의 차이</li> <li>수작업으로 인한 Human Error</li> </ul>                      | ●                                                                                   | <ul style="list-style-type: none"> <li>JBoss 전문가에 의해서 구성되는 최적화된 환경으로 웹서버/WAS 환경 구성</li> <li>자동화 구성으로 설치/구성의 오류 발생률 최소화</li> </ul>      |
| 성능    | ●                                                                                 | <ul style="list-style-type: none"> <li>설치 담당자의 기술력에 따라 튜닝 적용의 차이점 발생</li> <li>표준화하더라도 해당 가이드 적용에 대하여 지속적인 관리 필요</li> </ul> | ●                                                                                   | <ul style="list-style-type: none"> <li>전문가에 의한 각 부분별 튜닝 적용</li> <li>자동화된 구성으로 인하여 최적화된 표준환경 구성</li> </ul>                              |
| 유지보수  | ●                                                                                 | <ul style="list-style-type: none"> <li>구성 변경 시 마다 수작업</li> </ul>                                                           | ●                                                                                   | <ul style="list-style-type: none"> <li>구성 파일을 수정하여 자동화된 재설치</li> </ul>                                                                 |
| 생산성   | ●                                                                                 | <ul style="list-style-type: none"> <li>수작업에 의한 다운로드/튜닝</li> <li>OS/웹서버/WAS 서버 별 수작성 설치 구성</li> </ul>                       | ●                                                                                   | <ul style="list-style-type: none"> <li>자동화된 다운로드, 설치, 구성, 튜닝</li> <li>웹서버/WAS 서버에 최적화된 환경으로 표준화</li> </ul>                             |
| 벤더    | ●                                                                                 | <ul style="list-style-type: none"> <li>WebLogic/WebSphere/JEUS/Resin</li> <li>Apache httpd / Apache tomcat</li> </ul>      | ●                                                                                   | <ul style="list-style-type: none"> <li>Red Hat JBoss EAP + KHAN Provisioning</li> <li>Red Hat JBoss EWS + KHAN Provisioning</li> </ul> |

KHAN [apm]  
Provisioning 기능

# 새로운 시작 – 산 너머 산

Do **more** with **{less}**

IT조직 예산절감과  
혁신계획은 무엇인가?

가상화, 클라우드 그리고  
컨테이너 환경으로의 전환?

마이그레이션과  
오픈소스 내제화

저비용 고효율  
IT 인프라 기반 구축

0:19 / 0:49

KHAN HD

<https://www.youtube.com/watch?v=i9CtITakFw4&index=5&list=PLaFPokYzLL-QWKsbT40cxwr-cijeU6L>

# KHAN [apm] - 미들웨어 설치/구성/튜닝 자동화

또 뭘 자꾸 바꾸라고 그러지....쩍  
이거 원 하루 종일 삽질만 하네....

기존 수작업에 의한  
설치/구성/튜닝

▶ ⏮ 🔊 0:25 / 0:35

KHAN [ a p m ]

⚙️ HD □ 📺

<https://www.youtube.com/watch?v=0pefLIByALY&list=PLaFPOkYzL...QWKsbT40cxwr -cijeU6L&index=8>

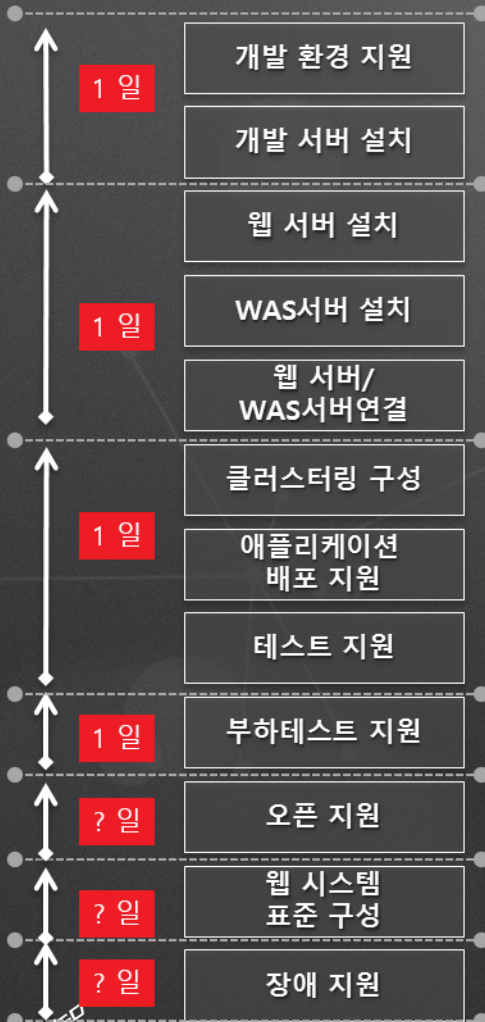
# 고객 요구사항에 따른 웹 시스템 작업

**KHAN**  
[ a p m ]



고객

수 일 이상  
기술 지원



기존 단순 설치

최적화된 웹시스템 구성

웹서버 설치

mod\_jk 설정

WAS 설치

JDK 설치

운영 계정 설정

방화벽 설정

배포 테스트

웹서버 튜닝

웹서버 설치  
보고서 작성

Worker MPM  
구성

로깅 디렉터리  
설정

mod\_cluster  
설정

웹취약점 조치

웹서버

도메인 모드 구성

멀티 인스턴스  
구성

JDBC 커넥션  
풀 튜닝

모니터링 설정

쓰레드 풀 튜닝

각종 Dump  
디렉터리 구성

모듈 설정

Log4j 설정

애플리케이션  
배포 구성

Native  
라이브러리 설정

JSP 변경  
체크 설정

WAS 서버  
설치보고서 작성

WAS

추가 rpm 설치

Huge Page 설정

OS 설정 보고서  
작성

블루투스 off

limits.conf  
설정

운영을 위한  
셸 구성

커널 파라미터  
설정

Cpusppd 설정

로깅 디렉터리  
설정

OS

기본 네트워크  
테스트

애플리케이션 배포  
테스트

운영관리  
테스트

클러스터링  
테스트

통합 테스트

멀티캐스트  
테스트

단위 테스트

Benchmark  
테스트

JDBC  
테스트

테스트

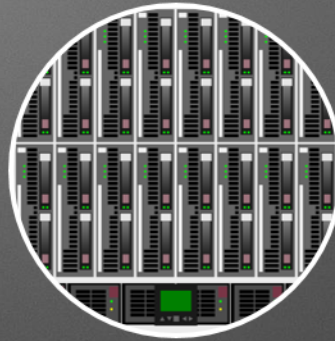
# KHAN [apm] - Provisioning 기능은?



OS 만 설치되어 있으면  
수분 이내에 설치 환경을  
테스트하고 웹서버  
와 WAS서버를 설치하  
고 즉시 서비스할 수 있  
는 환경 제공



미들웨어 전문가가 아니  
어도 전문가 수준의 시  
스템 튜닝이나 난이도  
높은 구성을 할 수 있도  
록 기능 제공



서버 구성에 대한 정보  
만 입력하면 한대에서  
수 십대까지 규모에 상  
관없이 자동으로  
웹시스템 운영환경을  
구성



웹시스템 설치/구성  
보고서를 시스템에 맞  
게 자동으로 생성하여  
개발팀과 운영 팀에게  
제공

수 분 내 튜닝된 웹 서버와 WAS 서버로 웹 시스템 구축

KHAN [apm]  
Monitoring 기능

# 장애와 성능 이슈에 대한 즉각적인 대응 체계 수립



현업 담당

개발자



직관적인 실시간  
모니터링



신속한 장애 파악  
및 예방 기능



운영 관련  
예측 및 조치

WAS 담당자

DBA

0:49 / 0:50

HD

Full Screen

Settings

<https://www.youtube.com/watch?v=A0oiFFUdOAA&list=PLaFPOkYzLL->

[WKsbT40cxwr\\_cijeU6L&index=1](#)

# KHAN [apm] 구성 아키텍처

- KHAN APM 구성 아키텍처
- 모니터링 기능은 대부분의 OS나 주요 WAS를 지원
- 프로비저닝 기능은 Apache /Tomcat/JBoss와 레드햇 계열 리눅스 환경을 지원

**KHAN** [ a p m ] = **A**h! **P**rovisioning & **M**onitoring



# APM = Application Performance Management

3티어구조에서 성능문제 발생시 신속한 진단과 조치를 위한 성능관리 도구!



- KHAN APM 은 실시간으로 어플리케이션 성능을 모니터링 하여 안정적인 시스템 운영을 지원하는 어플리케이션 성능 관리(Application Performance Management, APM) 솔루션
- 웹 어플리케이션의 내부를 실시간으로 모니터링 하면서 사용자들에게 쾌적하게 서비스가 제공되고 있는지를 "가용성"과 "성능"의 관점에서 분석
- 애플리케이션 서버에 에이전트를 배포 페이지 요청의 응답 시간과 쿼리에 대한 데이터베이스 응답 시간을 모니터링 하여 "서비스 장애 감지", "장애 원인 특정 시간 단축", "서비스 부하의 원인을 파악"

# KHAN [apm] – Monitoring 기능

## 실시간 성능 모니터링

- Request 모니터링
- Active Thread 모니터링
- 동시 접속사용자 모니터링
- CPU/Memory 사용량 모니터링
- 액티브서비스 모니터링
- APDEX 지수 리포팅

## 서비스 품질/ 장애 분석

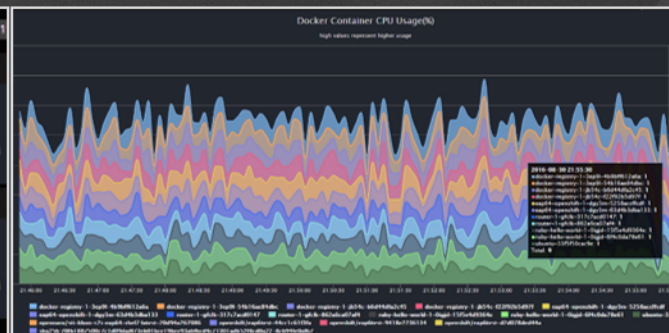
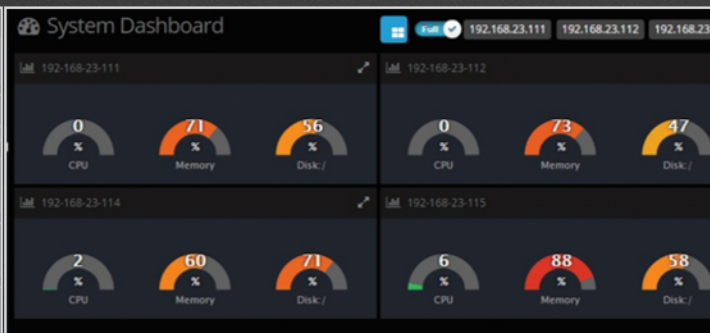
- 장애이벤트 모니터링
- 통계기반 Top 10 분석
- Thread 진단분석
- 장애진단 분석
- JVM 인스턴스 분석
- 진단분석 보고서



# 성능 모니터링과 위한 모니터링 기능 제공

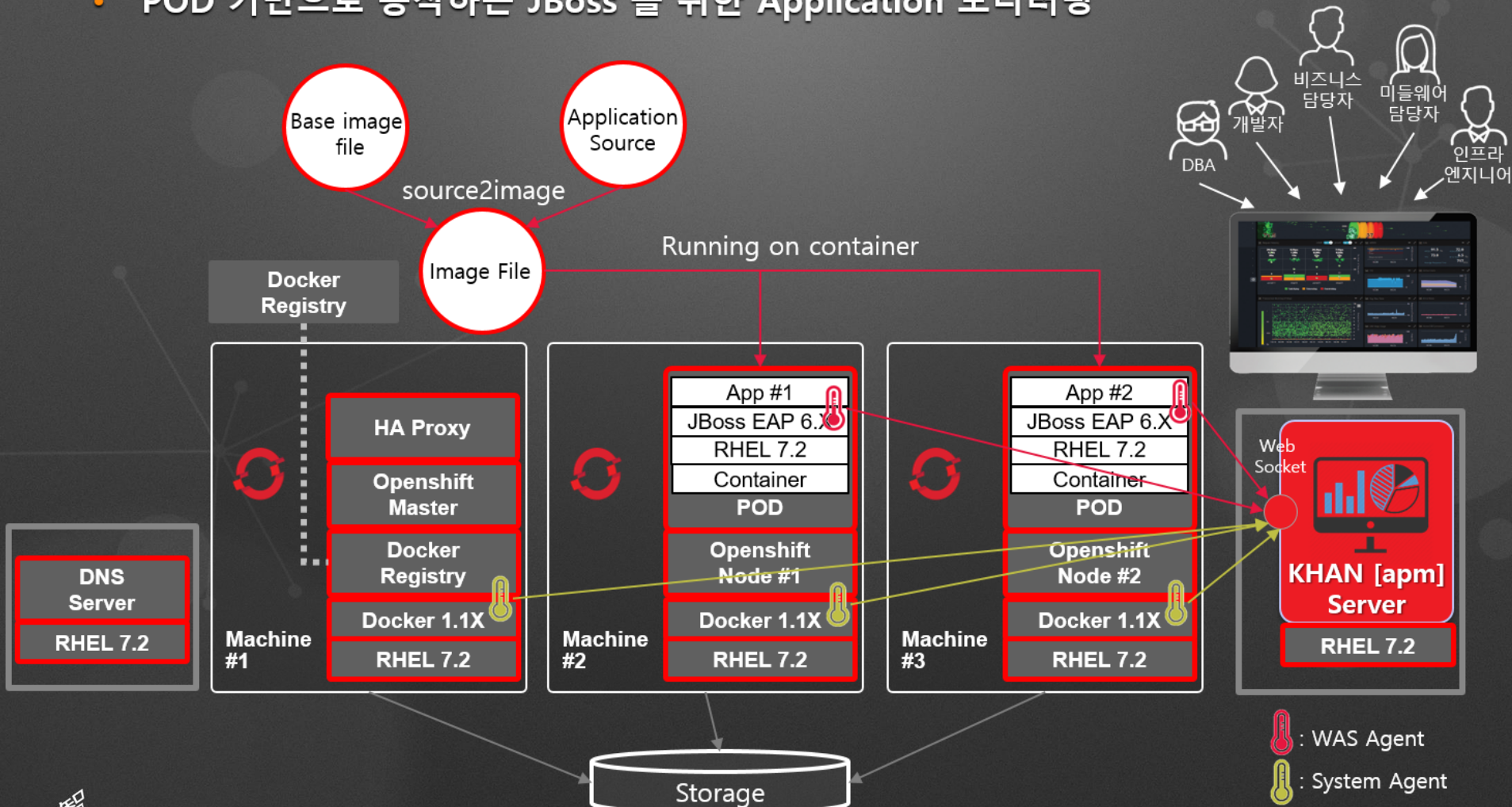


| User                                                                                                                                                                                         | Java                                                                                                                                                                                         | Hardware                                                                                                                                                                                                         | Container                                                                                                                                                                                                                                        |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Application                                                                                                                                                                                  | Application Server                                                                                                                                                                           | OS                                                                                                                                                                                                               | Docker                                                                                                                                                                                                                                           |
| 에러<br>느린 페이지<br>동시 사용자<br>서비스 만족도<br><div>장애 예측</div> <div>실시간 장애 감지</div> <div>서비스 오류율</div> <div>APDEX</div> <div>호출통계</div> <div>느린 URL 검색</div> <div>알림 센터</div> <div>일별/월별 통계 보고서</div> | SQL<br>TPS<br>Java<br>웹서버<br><div>JVM CPU 시간</div> <div>GC 시간</div> <div>메모리 사용률</div> <div>스레드 덤프 분석</div> <div>트랜잭션 히트 맵</div> <div>SQL 응답 시간</div> <div>수행 SQL</div> <div>웹서버 워커 상태</div> | Memory<br>Network<br>CPU<br>Disk<br><div>네트워크 패킷오류</div> <div>Netstat 상태</div> <div>디스크 I/O처리시간</div> <div>CPU 사용률</div> <div>디스크 I/O 사용량</div> <div>디스크 IO 평균대기수</div> <div>Load Average</div> <div>파일노드수</div> | Container<br>Images<br>Docker Info<br>Docker Stats<br><div>도커 컨테이너 리스트</div> <div>도커 이미지 리스트</div> <div>컨테이너 별 CPU 사용 현황</div> <div>컨테이너 별 메모리 사용률</div> <div>컨테이너 통계 정보</div> <div>이미지별 사이즈 정보</div> <div>도커 메타 정보</div> <div>이미지 히스토리 정보</div> |



# Openshift with KHAN APM Architecture

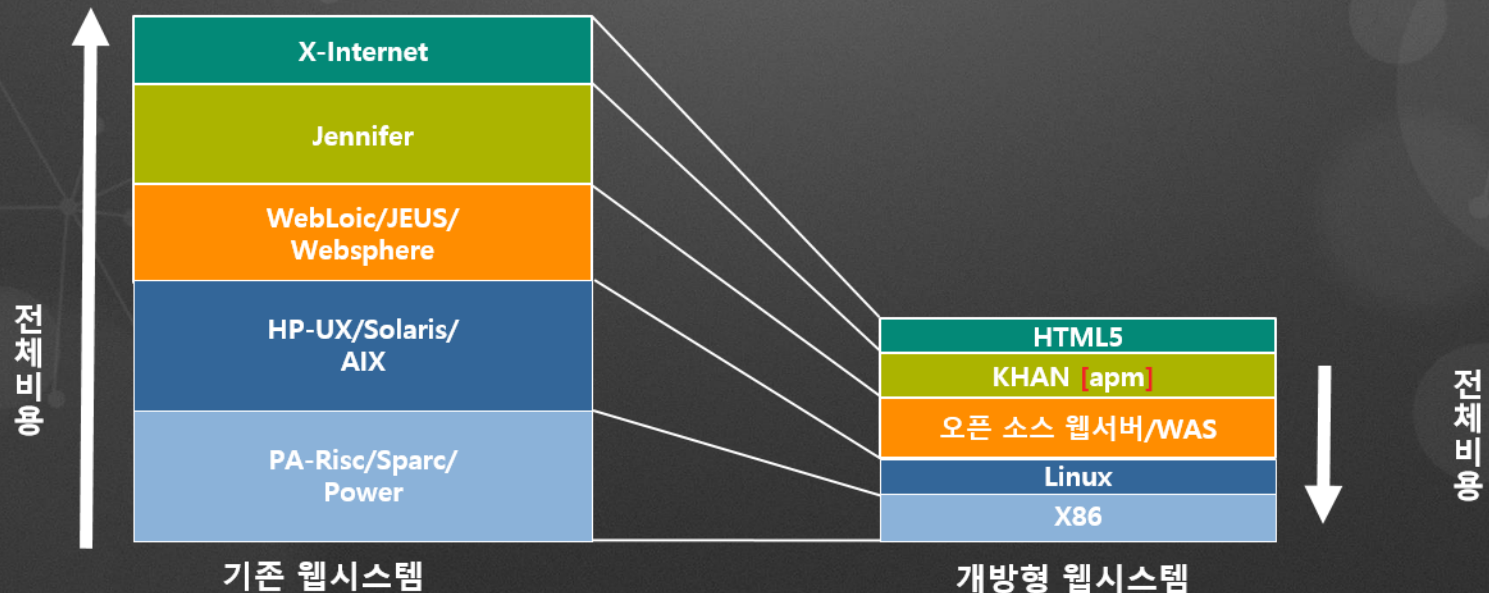
- 오픈시프트에서 성능 문제 발생 시 신속한 진단과 조치를 위한 성능 관리
- POD 기반으로 동작하는 JBoss 를 위한 Application 모니터링



# KHAN [apm] 제품 비교 - 비용 측면

- 오픈 소스 기반의 저비용 고효율 웹시스템 구축
- APM 을 통한 운영 효율성과 안정성 확보
- 서브스크립션 구매형태의 비용 절감과 벤더 종속성 탈피
- 웹시스템에 대한 일원화된 기술 지원 체계

## 저비용 고효율 시스템



# KHAN [apm] 제품 비교 – 기술 지원 서비스



|        | 설명                                                                                    | 오픈나루<br>기술지원 | 기존 기술지원      |             |
|--------|---------------------------------------------------------------------------------------|--------------|--------------|-------------|
|        |                                                                                       |              | 모니터링<br>기술지원 | WAS<br>기술지원 |
| 설치     | 목적 소프트웨어를 고객이 원하는 시스템 환경에 옮기고 정상적으로 작동되도록 하는 작업                                       | KHAN APM     |              | 수작업         |
| 최적화    | 시스템 또는 DBMS의 성능을 향상시키거나 저장장치를 효과적으로 사용하기 위해서 파일, 네트워크, 인덱스, 캐쉬, 버퍼에 관련된 파라미터를 변경하는 작업 | KHAN APM     |              | 수작업         |
| 튜닝     | 시스템 성능 향상을 위한 환경변수를 조정하는 작업                                                           | KHAN APM     |              | 수작업         |
| 문제 해결  | 목적 소프트웨어 자체의 문제(오동작, 에러, 버그, 해킹) 또는 운용 환경상의 문제(연동, 컨피그레이션) 등을 기술적으로 해결해 주는 작업         | KHAN APM     |              | 수작업         |
| 모니터링   | 설치된 소프트웨어의 실시간 운용 상황을 동적으로 관찰하여 통계적 자료를 제공하는 업무                                       | KHAN APM     | 모니터링         |             |
| 온라인 지원 | 포탈이나 이메일을 통해 질문이나 지원 요청을 접수하여 지식 베이스 또는 전담 기술인력을 동원하여 접수된 요청사항을 온라인으로 지원해주는 업무        | 오픈나루<br>서비스  |              | 기술지원<br>서비스 |
| 컨설팅    | 마이그레이션, 커스터마이징, 백업 등 공개소프트웨어 서비스에 관련된 사항을 지도해주는 업무                                    | 오픈나루<br>서비스  |              | 컨설팅         |
| 개발자 지원 | 아키텍처링, 파라미터 구성, 성능 튜닝, 최적화 등의 개발 업무에 대해 조언하는 업무                                       | 오픈나루<br>서비스  |              | 컨설팅         |
| 패치 제공  | 새로운 기술의 적용이나 운영체제의 변화 등으로 발생하는 불일치 조정                                                 | KHAN APM     |              | 수작업         |
| 업데이트   | 목적 소프트웨어뿐만 아니라 이에 관련된 라이브러리, 도구, 인터페이스 등 기존 설치 환경을 최신 버전으로 갱신 시키는 작업                  | KHAN APM     |              | 수작업         |
| 업그레이드  | 목적 소프트웨어의 버전을 향상시키는 작업으로 메이저 업그레이드와 마이너 업그레이드로 분류                                     | KHAN APM     |              | 수작업         |

# Log Monitoring vs. APM



With Logs With APM

0:11 / 1:02

<https://www.youtube.com/watch?v=8K0gbDcVfqY&list=PLaFPOkYzLL-QWKsbT40cxwr-cijeU6L&index=3>

“살아 남는 종(種)은 강한 종이 아니고,  
또 우수한 종도 아니다.  
변화에 적응하는 종이다.”

- Charles Darwin, 1809



감사합니다.



제품이나 서비스에 관한 문의

콜 센터 : 02-469-5426 ( 휴대폰 : 010-2243-3394 )

전자 메일 : [sales@opennaru.com](mailto:sales@opennaru.com)