

## 전자정부프레임워크 호환성 검증

---

### JBoss EAP6 호환성 검증 작업서

문서번호 : P80

프로젝트명	전자정부 프레임워크 JBoss EAP6 호환성 검증		
업무명	기술지원		
단계		버전	
작성자	최현우	최종작성일	2013.03.04
승인자		승인일	

## 개정이력

[illegible]

# Table of Contents

Table of Contents .....	iii
Revision History .....	v
<b>1. 개요 .....</b>	<b>1</b>
1.1 목적 .....	1
1.2 작업 일시 .....	1
1.3 Project Scope .....	1
1.4 주의점 .....	2
1.5 테스트 환경 .....	2
<b>2. 회사 및 제품 정보 .....</b>	<b>3</b>
2.1 JBOSS ENTERPRISE MIDDLEWARE 정보 .....	3
2.2 Red Hat 회사 정보 .....	3
<b>3. Red Hat JBoss EAP 6.0.x 운영환경 .....</b>	<b>4</b>
3.1 Java Virtual Machine 호환성 .....	4
3.2 인증된 구성 환경 .....	4
3.3 JBoss EAP 6.0.x 의 표준 지원 현황 .....	6
3.3.1 Java Enterprise Edition (EE) 지원 현황 .....	6
3.3.2 Web Services Standards / APIs .....	7
3.3.3 Additional Standards / APIs .....	8
3.4 JBoss 컴포넌트 버전 정보 .....	9
<b>4. 개발환경 구성 .....</b>	<b>12</b>
4.1 Eclipse 와 JBoss Tool 다운로드 .....	12
4.1.1 Eclipse 다운로드 .....	12
4.1.2 Eclipse Plugin - JBoss Tools 다운로드 .....	12
4.2 eGovFrame 개발환경 구성 .....	13
4.2.1 jboss-eap-6.0.1 제품 설치 .....	13
4.2.2 Eclipse 에서 JBoss EAP Runtime 구성 하기 .....	14
4.3 eGovFrame 개발 환경에서 어플리케이션 구성 및 배포 .....	17
<b>5. JBoss EAP 운영 가이드 .....</b>	<b>19</b>
5.1 JBoss EAP 설치 .....	19
5.1.1 jboss 설치 .....	19
5.1.2 User 추가 .....	19

5.1.3	srv01 서버 복사 .....	20
5.1.4	war 파일 복사.....	20
5.2	Env/Start/Shutdown Shell.....	20
5.3	어플리케이션 배포 .....	23
5.4	Apache Http + JBoss EAP 6.0 세션 클러스터링 구성 .....	24
5.4.1	개요 .....	24
5.4.2	Apache 설정 .....	25
5.4.3	jboss 설정 .....	28
5.5	Session Clustering 구성환경 점검.....	32
5.5.1	Session clustering 검증 방법 및 어플리케이션 .....	32
5.5.2	Session Clustering Test.....	33
5.6	Hot deplyment.....	34
5.6.1	hot deployment 중지 하기.....	34
5.7	CXF Web Services 통합 .....	35
5.7.1	개요 .....	35
5.7.2	CXF 확인.....	35
5.8	JBoss Datasource 설정.....	36
5.8.1	JDBC 드라이버 추가 및 등록.....	36
5.8.2	Data Source 추가 및 등록 .....	38
5.8.3	Data Source 를 이용한 DB 호출 샘플 JSP .....	41
5.8.4	JDBC DataSource 설정시 JNDI 이름 지정 .....	42
5.9	Spring 에서 JTA Transaction 설정 .....	44
5.10	jconsole 을 이용한 JMX 모니터링.....	46
5.10.1	jconsole.sh 실행.....	46
5.10.2	jconsole 을 실행한 후 연결 .....	47
5.11	Logging.....	48
<b>6.</b>	<b>전자정부 표준프레임워크 응용 어플리케이션 운영 환경 검증.....</b>	<b>51</b>
6.1	eGovFrame 운영 환경 어플리케이션 구성 및 배포.....	52
6.1.1	JBoss EAP 6 에서 valang.tld 파일 수정 .....	52
6.1.2	JBoss EAP 6 에서 pom.xml 파일 수정 .....	53
6.1.3	JBoss EAP 6 에서 Spring 2.5 사용시 수정.....	53
<b>7.</b>	<b>주요 질문과 답변들 .....</b>	<b>56</b>
7.1.1	Basic 30-Day Trial 신청 방법.....	56

7.1.2	JBoss Community 버전과 Enterprise 버전의 차이점은 무엇인가요? .....	56
7.1.3	JBoss EAP 6.0.x 설치 지원 환경은 어떻게 되나요? .....	57
7.1.4	JBoss EAP 6.0.x 에서 표준 지원 현황은 어떻게 되나요?.....	58
7.1.5	JBoss EAP 6.0.x 의 라이선스 정책은 어떻게 되나요?.....	58
7.1.6	JBoss EAP 6.0.x 의 문의는 어디로 하나요? .....	58

## Revision History

Name	Date	Reason For Changes	Version
최현우( <a href="mailto:sensewoo@penta.co.kr">sensewoo@penta.co.kr</a> )	2013/02/13	Initial Version	1.0

# 1. 개요

## 1.1 목적

Red Hat JBoss EAP 6 버전이 전자정부 표준프레임워크 2.5 버전과 상호 운용성이 적합한지 여부를 확인한다.

전자정부 표준프레임워크의 아키텍처 준수 여부, 실행 환경과의 라이브러리 충돌 및 치명적 오류 여부와 연동 환경이 지원 여부 등을 종합적으로 검증한다.

## 1.2 작업 일시

구분	내용
시간	2013 년 2 월 5 일 화요일 ~ 2013 년 3 월 4 일 월요일
장소	한국정보화진흥원
작업자	최현우
검증	

## 1.3 Project Scope

Red Hat JBoss EAP 6 는 전자정부 표준프레임워크센터에서 실시한 『전자정부 표준프레임워크 상용 SW 상호호환성 점검』에서 아래 호환성 점검항목에 대해 모두 점검 완료를 목표로 한다.

점검대상	점검결과
아키텍처	표준프레임워크에서 제시하는 아키텍처를 준수하고 있음
가이드	표준프레임워크 연동을 위한 가이드를 제공하고 있음
솔루션 호환성 점검	표준프레임워크 및 공통컴포넌트 기능이 정상 동작하고 있음

확장 및 연계 포인트 점검	표준프레임워크 기반 환경하에 연동을 위한 기능이 정상 동작하고 있음
-------------------	---------------------------------------

## 1.4 주의점

본문서에서 설명하고 있는 JBoss EAP 6 를 구성하고 관리하는 내용은 JBoss EAP 6 에 대한 매뉴얼을 기초로하여 필요한 간단한 지식을 정리하였으므로 정확하게 상세한 내용은 JBoss EAP 매뉴얼을 참조한다.

[https://access.redhat.com/knowledge/docs/JBoss\\_Enterprise\\_Application\\_Platform](https://access.redhat.com/knowledge/docs/JBoss_Enterprise_Application_Platform)

또한 본 문서의 내용은 참고용이므로 어떠한 법적 책임이 없다.

## 1.5 테스트 환경

### JVM

```
Java version: 1.6.0_29,Sun Microsystems Inc.
Java Runtime: Java(TM) SE Runtime Environment (build 1.6.0_29-b11)
Java VM: Java HotSpot(TM) 64-Bit Server VM 20.4-b02,Sun Microsystems Inc.
OS-System: Linux 2.6.18-238.el5,amd64
```

### OS System

```
[root@localhost ~]# date
2013. 03. 05. (화) 12:40:39 KST
[root@localhost ~]#
[root@localhost ~]#
[root@localhost ~]# cat /etc/redhat-release
Red Hat Enterprise Linux Server release 5.6 (Tikanga)
[root@localhost ~]#
[root@localhost ~]#
[root@localhost ~]# uname -a
Linux localhost.localdomain 2.6.18-238.el5 #1 SMP Sun Dec 19 14:22:44 EST 2010 x86_64 x86_64 x86_64 GNU/Linux
```

## 2. 회사 및 제품 정보

### 2.1 JBOSS ENTERPRISE MIDDLEWARE 정보

JBoss Enterprise Middleware 는 표준 기반의 오픈 소스이며 Java 어플리케이션 개발, 어플리케이션 및 서비스 호스팅, 콘텐츠 집적 및 프레젠테이션, 데이터 통합, 서비스 통합 및 조정을 위한 포괄적인 엔드 투 엔드 엔터프라이즈 미들웨어 포트폴리오입니다.

JBoss Enterprise Middleware 에는 전체 어플리케이션 수명 주기의 솔루션이 포함됩니다. 혁신적인 Web 2.0 개발 도구 및 프레임워크부터 SOA 기반 어플리케이션 및 서비스 배포, 완전한 미들웨어 관리 및 모니터링에 이르기까지 다양한 수명 주기의 솔루션이 여기에 해당됩니다.

JBoss Enterprise Middleware 는 서브스크립션을 통해 사용할 수 있고 여기에는 인증 소프트웨어, 업계 선두의 지원, 업데이트 및 패치, 설명서, 다년간의 유지 보수 정책 등이 포함됩니다.

[www.jboss.com](http://www.jboss.com) 에서 자세한 내용을 확인하십시오.

### 2.2 Red Hat 회사 정보

세계 선두의 오픈 소스 솔루션 제공업체인 Red Hat 은 노스캐롤라이나주 롤리에 본사를 두고 있고 전 세계에 50 개가 넘는 위성 사무소를 두고 있습니다. 최고 정보 책임자들은 CIO Insight Magazine Vendor Value 조사에서 4 년 연속으로 Red Hat 을 엔터프라이즈 소프트웨어 가치 부문 1 위로 선정했습니다.

Red Hat 은 당사의 운영 체제 플랫폼인 Red Hat Enterprise Linux 와 함께 높은 품질의 경제적인 기술을 제공하며 JBoss Enterprise Middleware 를 비롯해 어플리케이션, 관리 및 서비스 지향 아키텍처(SOA) 솔루션도 함께 제공합니다. Red Hat 은 또한 전 세계 고객들에게 지원, 교육 및 컨설팅 서비스를 제공합니다. [www.redhat.com](http://www.redhat.com) 에서 자세한 내용을 확인하십시오.



## 3. Red Hat JBoss EAP 6.0.x 운영환경

### 3.1 Java Virtual Machine 호환성

다음 URL 에서 JBoss EAP 6.0.x 버전에서 인증된 운영환경에 대한 정보를 확인 할 수 있다.

<https://access.redhat.com/knowledge/ko/articles/297473>

Red Hat 은 아래와 같은 구성환경에 대한 호환과 인증 모두를 지원한다.

Java Virtual Machine	Version
Oracle JDK	1.6, 1.7
OpenJDK	1.6, 1.7
IBM JDK	1.6, 1.7

### 3.2 인증된 구성 환경

다음과 같은 OS, 칩아키텍처 그리고 JVM 구성 환경에 대하여 JBoss Enterprise Application Platform 6 에 대한 완벽한 지원을 인증한다.

Operating System	Chip Architecture	Java Virtual Machine(s)
Red Hat Enterprise Linux 6 (latest update)	x86, x86_64	Oracle JDK 1.6, 1.7 IBM JDK 1.6, 1.7 OpenJDK 1.6, 1.7
Red Hat Enterprise Linux 5 (latest update)	x86, x86_64	Oracle JDK 1.6, 1.7 IBM JDK 1.6, 1.7 OpenJDK 1.6
Microsoft Windows 2008 Server	x86	Oracle JDK 1.6, 1.7

Microsoft Windows 2008 Server R2	x86_64	Oracle JDK 1.6, 1.7
Solaris 10	x86, x86_64, SPARC 64	Oracle JDK 1.6, 1.7
Solaris 9	x86, x86_64, SPARC 64	Oracle JDK 1.6, 1.7

JBoss Enterprise Application Platform 6 는 다음과 같은 데이터베이스와 데이터베이스 드라이버에 대해서 지원을 인증한다.

Database	Version	Database Driver
IBM DB2	9.7	IBM DB2 JDBC Universal Driver Architecture 4
Oracle	11g R2 11g R2 RAC 11g R1 11g R1 RAC	Oracle JDBC Driver v11 Oracle JDBC Driver v11 Oracle JDBC Driver v11 Oracle JDBC Driver v11
Microsoft	SQL Server 2008 R2 SQL Server 2008	Microsoft SQL Server JDBC Driver 3 Microsoft SQL Server JDBC Driver 3
MySQL	Cluster 7.1 5.5 5.1	MySQL Connector/J 5 MySQL Connector/J 5 MySQL Connector/J 5
PostgreSQL	9.1 8.4	JDBC4 Postgresql Driver, Version 9 JDBC4 Postgresql Driver, Version 8
Sybase	ASE 15.7	Sybase jConnector JDBC Driver v7
FUJITSU	Integrated System HA Database Ready SX1	Open SQL Client V1.0, JDBC4 Driver

JBoss Enterprise Application Platform 6 는 다음과 같은 웹서버와 커넥터에 대해서 지원을 인증한다.

Web Server	Version	OS	Connector [1]
Application Platform 6 Apache Web Server (Apache 2.2.17 기반)	6.0.0	Red Hat Enterprise Linux 6(x86, x86_64) Red Hat Enterprise Linux 5(x86, x86_64) Solaris 11(x86, x86_64, SPARC64)	mod_cluster 1.2.1 Final mod_jk 1.2.31

		Solaris 10(x86, x86_64, SPARC64) Windows Server 2008(x86, x86_64) Windows Server 2008 R2(, x86_64)	
Microsoft IIS	7.5	Windows Server 2008 R2 (x86, x86_64)	isapi_redirect 1.2.30
Oracle iPlanet Web Server	7.0	Solaris 11(x86, x86_64, SPARC64) Solaris 10(x86, x86_64, SPARC64)	nsapi_redirector 1.2.32

### 3.3 JBoss EAP 6.0.x 의 표준 지원 현황

아래의 URL 에서 최신 표준 지원 현황에 대한 확인이 가능합니다.

<https://access.redhat.com/knowledge/ko/articles/299283>

JBoss EAP(JBoss® Enterprise Application Platform) 릴리스에서 다음과 같은 업계 표준이 지원됩니다.  
호환성 검증을 위해 각 주요 및 업데이트 릴리스에 표준 인증 테스트를 수행했습니다(적용 가능한 경우).

#### 3.3.1 Java Enterprise Edition (EE) 지원 현황

JBoss EAP 릴리즈 에 해당 하는 Java EE 표준 지원 현황은 다음과 같다.

Technology/Specification	EAP 6.0	EAP 5.1
Java Platform, Enterprise Edition (Java EE) [JSR-151, 244, 316]	6.0	5.0
Java API for XML-based RPC (JAX-RPC) [JSR-101]	1.1	1.1
SOAP with Attachments API for Java (SAAJ) [JSR-067]	1.3	1.3
Java Servlet [JSR-154, 315]	3.0	2.5
JavaServer Faces (JSF) [JSR-252 , 314]	2.0	1.2
JavaServer Pages (JSP) [JSR-245 ]	2.2	2.1
Java Transaction API (JTA) [JSR-907 ]	1.1	1.1
Java Authorization Contract for Containers (JACC) [JSR-115 ]	1.3	1.1
Enterprise Java Beans (EJB) [JSR-153, 220, 318]	3.1	2.1, 3.0

J2EE Connector Architecture [JSR-112, 322]	1.6	1.5
J2EE Management API [JSR-077]	1.1	1.0
Enterprise Web Services [JSR-109]	1.3	1.2
J2EE Deployment API [JSR-088 ]	1.2	1.2
JavaMail [JSR-919 ]	1.4	1.4
Java Messaging Service (JMS) [JSR-914 ]	1.1	1.1
Java Persistence API (JPA) [JSR-220, 317]	2.0	1.0
Web Services Metadata for the Java Platform [JSR-181 ]	2.0	2.0
Java API for XML Web Services (JAX-WS) [JSR-224]	2.2	2.1
JSP Debugging [JSR-045]	1.0	1.0
JStandard Tag Library for JavaServer Pages (JSTL) [JSR-052]	1.2	1.2
Common Annotations for the Java Platform [JSR-250]	1.1	1.0
Java Architecture for XML Binding (JAXB) [JSR-222 ]	2.2	2.0
Java API for RESTful Web Services (JAX-RS) [JSR-311]	1.1	1.0
JAXR(Java API for XML Registries) [JSR-093]	1.0	1.0
CDI(Contexts and Dependency Injection for Java) [JSR-299]	1.0	–
Bean Validation [JSR-303]	1.0	–

### 3.3.2 Web Services Standards / APIs

JBoss EAP 에서 지원하는 웹서비스 표준은 다음과 같다.

Technology/Specification	EAP 6.0	EAP 5.1
Simple Object Access Protocol (SOAP)	1.2	1.2
SOAP Message Transmission Optimization Mechanism (MTOM)	Yes	Yes

XML-Binary Optimized Packaging (XOP)	Yes	Yes
Web Services Description Language (WSDL)	1.1, 2.0	1.1, 2.0
WS-I Basic Profile	1.1	1.1
WS-Addressing	1.0	1.0
WS-Reliable Messaging	1.1	1.0 *
WS-Security	1.1	1.1
Universal Description Discovery and Integration (UDDI)	2.0, 3.0	2.0
Java API for XML Web Services Addressing (JAX-WSA) [JSR-261]	1.0	1.0
WS-Atomic Transactions	1.1	1.1
Fast Infoset (FI - ISO/IEC 24824-1, ITU-T Rec. X.891)	Yes	Yes
WS-Business Activity	1.1	1.1
WS-Coordination	1.1	1.1
WS-Security Policy	1.3	1.3'
WS-Policy	1.5	1.5*
WS-Policy Attachment	1.5	1.5*
WS-I Attachments Profile	1.0	1.0
WS-Eventing	–	–
WS-Trust	1.3	1.3'
SAML	–	2.0**
XACML	2.0	2.0

\*Included in optional JBoss WS Apache CXF

\*\* Included in Picket Link Federation Technical Preview

### 3.3.3 Additional Standards / APIs

JBoss EAP 에서 지원하는 추가적인 표준 지원은 다음과 같다.

Technology/Specification	EAP 6.0	EAP 5.1
Java Transaction Service (JTS)	1.0	1.0
CORBA	2.3.1	2.3.1
JDBC [JSR-054]	[3]	3.0, 4.0
Streaming API for XML Processing(StAX) [JSR-173]	[3]	1.0
JAF(JavaBeans Activation Framework) [JSR-925]	[3]	1.1
JMX(Java Management Extensions) [JSR-255]	[3]	2.0
JDBC(Java Database Connectivity) [JSR-221]	[3]	4.0
JAXP(Java API for XML Processing) [JSR-205]	[3]	1.3
SPNEGO/Kerberos	Yes	Yes
Notes: [3] Now implemented via Java SE		

### 3.4 JBoss 컴포넌트 버전 정보

JBoss EAP 에 포함된 컴포넌트에 대한 버전 정보는 아래의 URL 에서 확인 가능하다.

<https://access.redhat.com/knowledge/ko/articles/299223>

Component	EAP 5.1.1	EAP 5.1.2
JBoss AS	7.1.2 Final	7.1.3 Final
WELD	1.1.8.Final	1.1.10.Final
Rest Easy	2.3.3.Final	2.3.4.Final
mod_cluster	1.2.1.Final	1.2.3.Final
Picketlink Federation	2.1.1.Final	2.1.3.Final
JBoss Modules	1.1.2.Final	1.1.3.Final
JBoss DMR	1.1.1.Final	1.1.1.Final

JBoss Metadata	7.0.3.Final	7.0.4.Final
JBoss XNIO	3.0.4.GA	3.0.7.GA
PicketBox	4.0.9.Final	4.0.14.Final
PicketBox Commons	1.0.0.Final	1.0.0.Final
Hibernate Core	4.1.3.Final	4.1.6.Final
Hibernate Infinispan (2LC)	4.1.3.Final	4.1.6.Final
Hibernate Envers	4.1.3.Final	4.1.6.Final
Hibernate Validator	4.2.0.Final	4.2.0.Final
Hibernate JPA 2.0 API	1.0.1.Final	1.0.1.Final
JBoss Web	7.0.16.Final	7.0.17.Final
Infinispan	5.1.4.Final	5.1.8.Final
IronJacamar (JCA 1.6)	1.0.11.Final	1.0.13.Final
JGroups	3.0.9.Final	3.0.14.Final
JBoss Transactions	4.16.4.Final	4.16.6.Final
JBoss EJB Client	1.0.10.Final	1.0.11.Final
JBoss EJB3 EXT API	2.0.0	2.0.0
JBoss WS API	1.0.0.GA	1.0.0.GA
JBoss WS SPI	2.0.3.GA	2.0.4.GA
JBoss WS Common	2.0.4.GA	2.0.4.GA
JBoss WS Common Tools	1.0.1.GA	1.0.2.GA
JBoss WS-CXF	4.0.4.GA	4.0.6.GA
JBoss WS-Native(jaxrpc 만 해당)	4.0.4.GA	4.0.6.GA
JBoss VFS	3.1.0.Final	3.1.0.Final
JBoss Remoting3	3.2.8.GA	3.2.14.GA
JBoss Marshalling	1.3.14.GA	1.3.15.GA

JBoss SASL	1.0.1.Final	1.0.3.Final
JSF	1.2_15-b01-redhat-1	1.2_15-b01-redhat-2
JSF2	2.1.7-redhat-1	2.1.13-redhat-1
JacORB	2.3.2-redhat-1	2.3.2-redhat-2
JBoss Security Negotiation	2.2.0.SP1	2.2.1.Final
JBoss Management Console	1.3.1.Final	1.4.2.Final
Hornet Q	2.2.16.Final	2.2.23.Final
Netty(HornetQ and Infinispan 에 포함)	3.2.6.Final	3.2.6.Final
JBoss Logging	3.1.1.GA	3.1.2.GA
Apache CXF	2.4.6	2.4.9
Apache CXF XJC-Utils	2.4.0	2.4.0
Apache WSS4J(WS 용)	1.6.5	1.6.7
Xalan	2.7.1-redhat-1	2.7.1-redhat-3
Xerces	2.9.1-redhat-1	2.9.1-redhat-3
JAXB	2.2.5	2.2.5-redhat-3
Application Platform 6.0.0 Apache Web Server	based on Apache 2.2.17	based on Apache 2.2.17



## 4. 개발환경 구성

이후에 다루어지는 내용은 JBoss EAP6 에서의 개발환경 구성에 대한 것입니다.

Eclipse 환경에서 JBoss Tools 를 설치하고 개발 환경을 설정하는 내용에 대해서 설명합니다.

### 4.1 Eclipse 와 JBoss Tool 다운로드

#### 4.1.1 Eclipse 다운로드

<http://www.eclipse.org/downloads/>



JBoss EAP6 와 함께 사용될 수 있는 eclipse 버전은 Juno 와 Indigo 입니다.

- JBoss Tools 4.0 Development Milestone (requires Eclipse 4.2 Juno): Core, SOA Tooling
- JBoss Tools 3.3 Development Milestone (requires Eclipse 3.7.2 Indigo SR2): Core, SOA Tooling

#### 4.1.2 Eclipse Plugin - JBoss Tools 다운로드

아래의 디렉토리에서 설치된 eclipse 버전에 맞는 JBoss Tools 의 url 을 선택합니다.

JBoss Tools 다운로드 - <http://download.jboss.org/jbosstools/updates/development/>

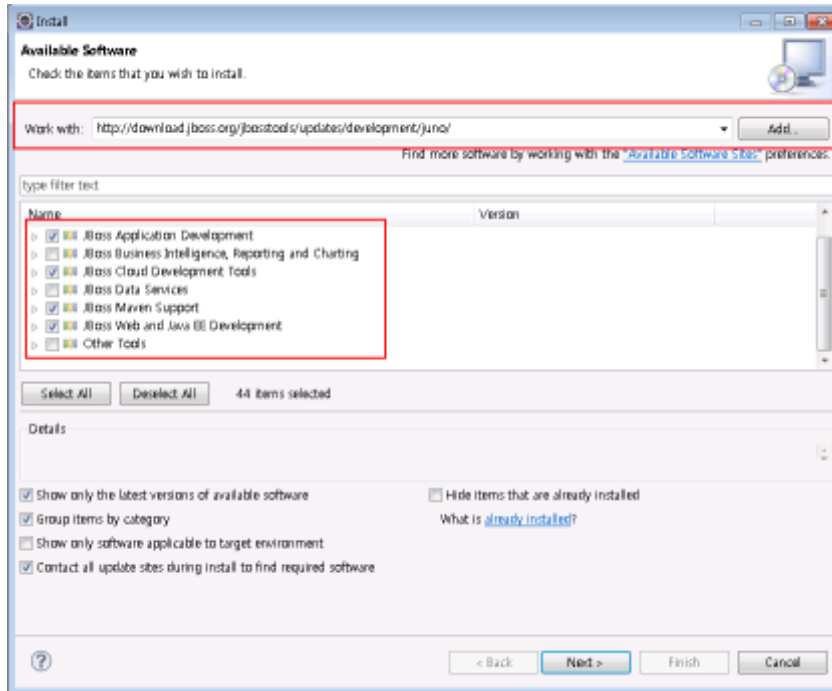
Help -> Install New Software-> Available Software 에서 아래의 URL 추가

JBoss Tools4.0 설치 URL- <http://download.jboss.org/jbosstools/updates/development/juno/>

설치 할 플러그인은 다음과 같습니다.

- [Application Development]
- [Maven Support]
- [Web and Java EE Development]

를 선택한 후 [Next]로 진행합니다.



quickstarts 응용 프로그램을 Eclipse 에서 시작하는 경우 m2eclipse 도 필요합니다.

## 4.2 eGovFrame 개발환경 구성

eGovFramework 을 위한 JBoss 개발환경을 구성하기 위해서는 다음과 같은 작업을 진행한다.  
개발환경 구성 기준은 JBoss EAP 6.0.1 이상의 버전으로 한다.

### 4.2.1 jboss-eap-6.0.1 제품 설치

JBoss EAP6 다운로드 - <https://access.redhat.com/home> (계정 등록 필요)

다운로드 받은 "jboss-eap-6.0.1.zip" 파일을 풀니다.

```
unzip jboss-eap-6.0.1.zip
```

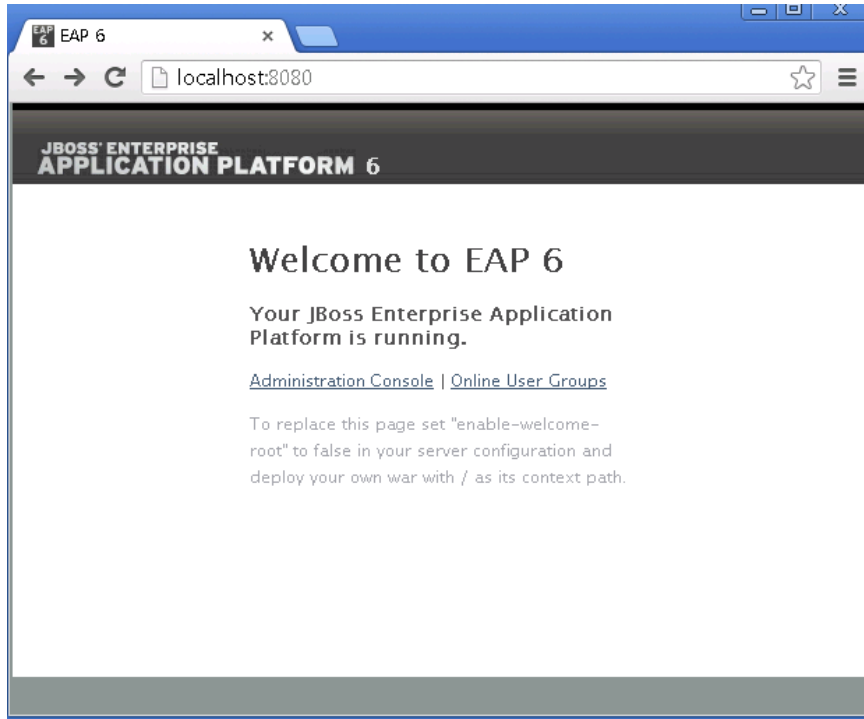
JBoss 설치 디렉토리에 "/bin"으로 이동하여 standalone 인스턴스를 실행합니다.

```
$JBoss_Home\bin\standalone.bat
```

이상으로 JBoss EAP6 의 설치 및 시작되었습니다.

<http://localhost:8080> 에 액세스하여 server 가 제대로 작동하고 있는지 확인하십시오.

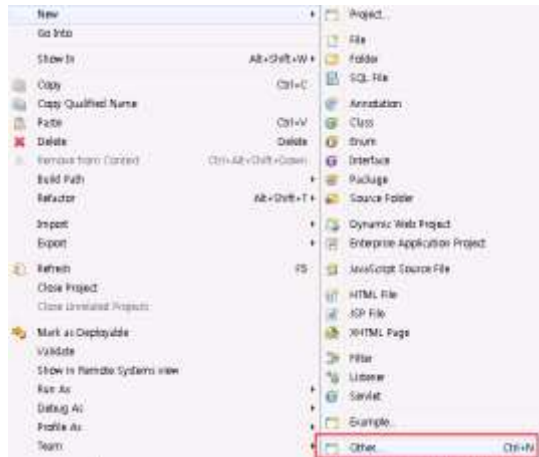
<http://localhost:8080/>



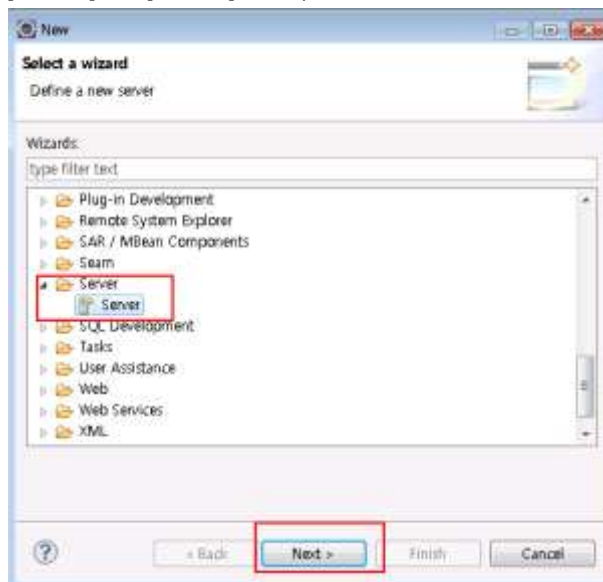
## 4.2.2 Eclipse 에서 JBoss EAP Runtime 구성 하기

JBoss Tools 를 설치한 후 개발하는 응용 프로그램을 실행할 JBoss EAP Server Runtime 설정은 다음과 같은 과정으로 진행 한다.

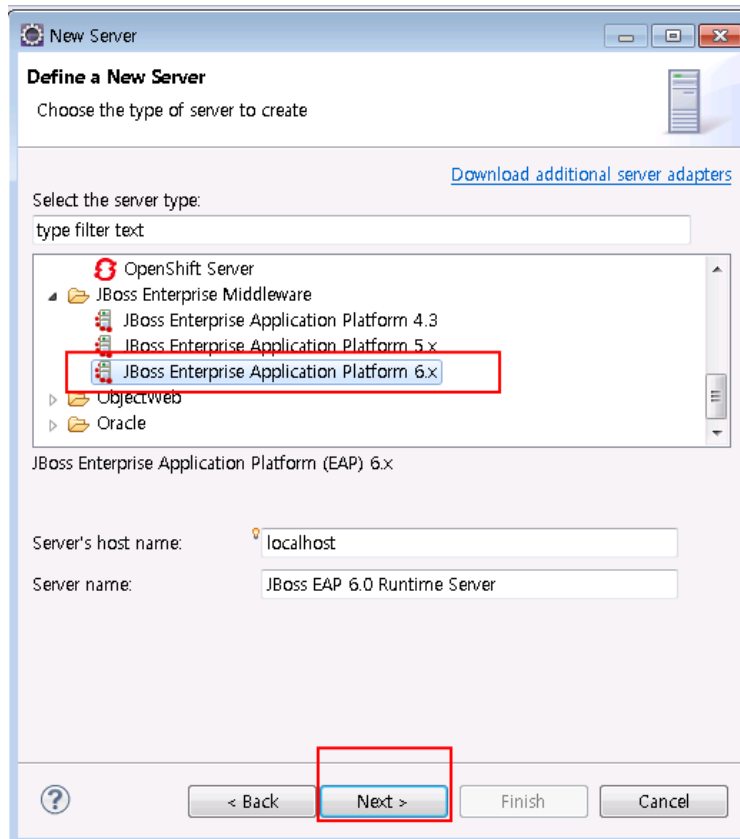
- [New] -> [Other] 를 선택



- [Server] -> [Server] 선택



- [JBoss Enterprise Middleware] -> [JBoss Enterprise Application Platform 6.x] 선택



- [JBoss Runtime ]



위의 화면에서 JBoss EAP6 설치 디렉토리와 설치된 JDK 환경을 설정해 줍니다.

- [Create a new JBoss Server]



위의 화면에서 다른 설정 없이 바로 [Finish]를 선택합니다.

### 4.3 eGovFrame 개발 환경에서 어플리케이션 구성 및 배포

JBoss 에서 개발환경은 편의상 JBoss EAP 6.0.1 이상의 버전에서 구성하는 것을 권장하며, 유의할 점은 "spring-modules-validation-0.9.jar/META-INF/valang.tld" 에 포함된 내용 중 잘못된 "body-content" 값이 있는 버그가 있다.

다음의 "spring-modules-validation-0.9.jar" 파일의 META-INF/valang.tld 파일에서 "None"을 "empty" 로 설정한다.

## valang.tld 파일 수정하기

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<!DOCTYPE taglib PUBLIC "-//Sun Microsystems, Inc.//DTD JSP Tag Library 1.2//EN" "http://java.sun.com/dtd/web-
jsptaglibrary_1_2.dtd">
<taglib>
  <tlib-version>1.1.1</tlib-version>
  <jsp-version>1.2</jsp-version>
  <short-name>Valang</short-name>
  <uri>http://www.springmodules.org/tags/valang</uri>
  <description>A set of tags that enable the translation of Valang
    validators into JavaScript.</description>
  <tag>
    <name>codebase</name>
    <tag-class>org.springmodules.validation.valang.javascript.taglib.ValangCodebaseTag</tag-class>
    <body-content>empty</body-content>
    <description>Generates the JavaScript code that is necessary for the
      translated JavaScript validation rules to execute. This code needs
      to be included once per page and before any generated validation
      rules.</description>
```

## 5. JBoss EAP 운영 가이드

### 5.1 JBoss EAP 설치

아래와 같은 절차 대로 JBoss 인스턴스 구성 및 어플리케이션을 배포한다.

#### 5.1.1 jboss 설치

<https://access.redhat.com/> 로 사용자 등록 후 제품을 다운로드 한다.

"/was/jboss-eap-6.0/" 디렉토리에 다운로드 한 jboss eap 6.0 을 설치한다.

#### 5.1.2 User 추가

관리자 계정을 추가한다.

"/was/jboss-eap-6.0/bin/add-user.sh" 를 실행한다.

- ㉠ - Management/Application User 선택 (default = a)
- ㉡ - 관리영역 (default = ManagementRealm)
- ㉢ - 등록할 관리자 ID
- ㉣ - 등록할 관리자 PassWord

What type of user do you wish to add?

- a) Management User (mgmt-users.properties)
- b) Application User (application-users.properties)

(a): **a**Enter

Enter the details of the new user to add.

Realm (ManagementRealm) : **b**Enter

Username : **test**

Password : **password**

Re-enter Password : **password**

About to add user 'test' for realm 'ManagementRealm'

Is this correct yes/no? **yes**

Added user 'test' to file '/home/jboss/eap6/jboss-eap-6.0/standalone/configuration/mgmt-users.properties'

Added user 'test' to file '/home/jboss/eap6/jboss-eap-6.0/domain/configuration/mgmt-users.properties'

Is this new user going to be used for one AS process to connect to another AS process?

e.g. for a slave host controller connecting to the master or for a Remoting connection for server to server EJB calls.

yes/no? **yes**



To represent the user add the following to the server-identities definition <secret value="dGVzdDAx" />

### 5.1.3 srv01 서버 복사

설치된 jboss 디렉토리에 standalone(/was/jboss-eap-6.0/standalone) 서버 구성을 "srv01" 구성으로 복사한다.

"/was/jboss-eap-6.0/srv01/" 디렉토리에 설치되어 있다.

### 5.1.4 war 파일 복사

기존에 정상적으로 동작하는 war 어플리케이션을 새로 작성된 srv01 서버의 deployments 디렉토리에 복사한다.

"/was/jboss-eap-6.0/srv01/deployments" 디렉토리에 복사한다.

## 5.2 Env/Start/Shutdown Shell

JBoss EAP 실행 스크립트는 다음의 디렉토리에 작성되어 있다.

```
[jboss@localhost srv01]$ pwd
```

```
/was/jboss-eap-6.0/srv01
```

#### env.sh

```
#!/bin/sh
```

```
DATE=`date +%Y%m%d%H%M%S`
```

```
##### JBOSS Directory Setup #####
```

```
export JBOSS_HOME=/was/jboss-eap-6.0
```

```
export DOMAIN_BASE=/was/jboss-eap-6.0
```

```
export SERVER_NAME=srv01
```

```
##### Configuration File #####
```

```
export CONFIG_FILE=standalone-full-ha.xml
```

```
export HOST_NAME=localhost.localdomain
```

```
export NODE_NAME=$SERVER_NAME
```

```
export PORT_OFFSET=0
```

```
export JBOSS_USER=jboss
```

```
##### Bind Address #####
export BIND_ADDR=192.168.100.4

export MULTICAST_ADDR=230.1.0.1
export JMS_MULTICAST_ADDR=231.7.0.1
export MODCLUSTER_MULTICAST_ADDR=224.0.1.105

export MGMT_ADDR=192.168.100.4

export CONTROLLER_IP=$MGMT_ADDR
let CONTROLLER_PORT=9999+$PORT_OFFSET
export CONTROLLER_PORT

export LAUNCH_JBOSS_IN_BACKGROUND=true

##### JBoss System module and User module directory #####
export JBOSS_MODULEPATH=$JBOSS_HOME/modules:$JBOSS_HOME/modules.ext

# JVM Options : Server
export JAVA_OPTS="-server $JAVA_OPTS"

# JVM Options : Memory
export JAVA_OPTS=" $JAVA_OPTS -Xms512m -Xmx1024m -XX:MaxPermSize=256m"

export JAVA_OPTS=" $JAVA_OPTS -XX:+PrintGCTimeStamps "
export JAVA_OPTS=" $JAVA_OPTS -XX:+PrintGCDetails "
export JAVA_OPTS=" $JAVA_OPTS -Xloggc:gclogs/gc_$(DATE).log "
export JAVA_OPTS=" $JAVA_OPTS -XX:+UseParallelGC "
#export JAVA_OPTS=" $JAVA_OPTS -XX:+UseConcMarkSweepGC "
export JAVA_OPTS=" $JAVA_OPTS -XX:+ExplicitGCInvokesConcurrent "
export JAVA_OPTS=" $JAVA_OPTS -XX:-HeapDumpOnOutOfMemoryError "

# Linux Large Page Setting
#export JAVA_OPTS=" $JAVA_OPTS -XX:+UseLargePages "

#export JAVA_OPTS=" $JAVA_OPTS -verbose:gc"
export JAVA_OPTS=" $JAVA_OPTS -Djava.net.preferIPv4Stack=true"
export JAVA_OPTS=" $JAVA_OPTS -Dorg.jboss.resolver.warning=true"
export JAVA_OPTS=" $JAVA_OPTS -Dsun.rmi.dgc.client.gcInterval=3600000 "
export JAVA_OPTS=" $JAVA_OPTS -Dsun.rmi.dgc.server.gcInterval=3600000"
export JAVA_OPTS=" $JAVA_OPTS -Djboss.modules.system.pkgs=org.jboss.byteman"
export JAVA_OPTS=" $JAVA_OPTS -Djava.awt.headless=true"

export JAVA_OPTS=" $JAVA_OPTS -Djboss.server.base.dir=$DOMAIN_BASE/$SERVER_NAME"
export JAVA_OPTS=" $JAVA_OPTS -Djboss.socket.binding.port-offset=$PORT_OFFSET"
export JAVA_OPTS=" $JAVA_OPTS -Djboss.node.name=$NODE_NAME"
```

```

export JAVA_OPTS=" $JAVA_OPTS -Djboss.bind.address.management=$MGMT_ADDR"
export JAVA_OPTS=" $JAVA_OPTS -Djboss.bind.address=$BIND_ADDR"
export JAVA_OPTS=" $JAVA_OPTS -Djboss.bind_addr=$MULTICAST_ADDR"
#export JAVA_OPTS=" $JAVA_OPTS -Djboss.default.jgroups.stack=tcp"
export JAVA_OPTS=" $JAVA_OPTS -Djboss.default.multicast.address=$MULTICAST_ADDR"
export JAVA_OPTS=" $JAVA_OPTS -Djboss.messaging.group.address=$JMS_MULTICAST_ADDR"
export JAVA_OPTS=" $JAVA_OPTS -Djboss.modcluster.multicast.address=$MODCLUSTER_MULTICAST_ADDR"

export JAVA_OPTS=" $JAVA_OPTS -Dorg.jboss.as.logging.per-deployment=false"

export JAVA_OPTS=" $JAVA_OPTS -Dserver.mode=local"

echo "=====
echo "JBOSS_HOME=$JBOSS_HOME"
echo "DOMAIN_BASE=$DOMAIN_BASE"
echo "SERVER_NAME=$SERVER_NAME"
echo "CONFIG_FILE=$CONFIG_FILE"
echo "BIND_ADDR=$BIND_ADDR"
echo "PORT_OFFSET=$PORT_OFFSET"
echo "MULTICAST_ADDR=$MULTICAST_ADDR"
echo "CONTROLLER=$CONTROLLER_IP:$CONTROLLER_PORT"
echo "=====

```

### Start.sh

```

#!/bin/sh

DATE=`date +%Y%m%d%H%M%S`

. ./env.sh

PID=`ps -ef | grep java | grep "=$SERVER_NAME " | awk '{print $2}`
echo $PID

if [ e$PID != "e" ]
then
    echo "JBoss SERVER - $SERVER_NAME is already RUNNING..."
    exit;
fi

UNAME=`id -u -n`
if [ e$UNAME != "e$JBOSS_USER" ]
then
    echo "Use $JBOSS_USER account to start JBoss SERVER - $SERVER_NAME..."
    exit;
fi

```

```

echo $JAVA_OPTS

if [ ! -d "./nohup" ];
then
    mkdir nohup
fi

mv $SERVER_NAME.out ./nohup/$SERVER_NAME.out.$DATE

nohup $JBoss_HOME/bin/standalone.sh -Dserver=$SERVER_NAME -b $BIND_ADDR -u $MULTICAST_ADDR -
P=$DOMAIN_BASE/$SERVER_NAME/env.properties -c $CONFIG_FILE >> $SERVER_NAME.out &

if [ e$1 = "enotail" ]
then
    echo "Starting... $SERVER_NAME"
    exit;
fi

tail -f $SERVER_NAME.out

```

#### shutdown.sh

```

#!/bin/sh

. ./env.sh

$JBoss_HOME/bin/jboss-cli.sh --connect --controller=$CONTROLLER_IP:$CONTROLLER_PORT --command=:shutdown

```

## 5.3 어플리케이션 배포

Application 배포 배포 방법은 기본적으로 설치된 JBoss 의 deployments 디렉토리에 war, ear, Jar 파일 또는 extracted war, ear, jar 형태의 배포를 한다.

상세한 내용은 "JBoss EAP 6.0 Administration And Configuration Guide"를 참조한다.

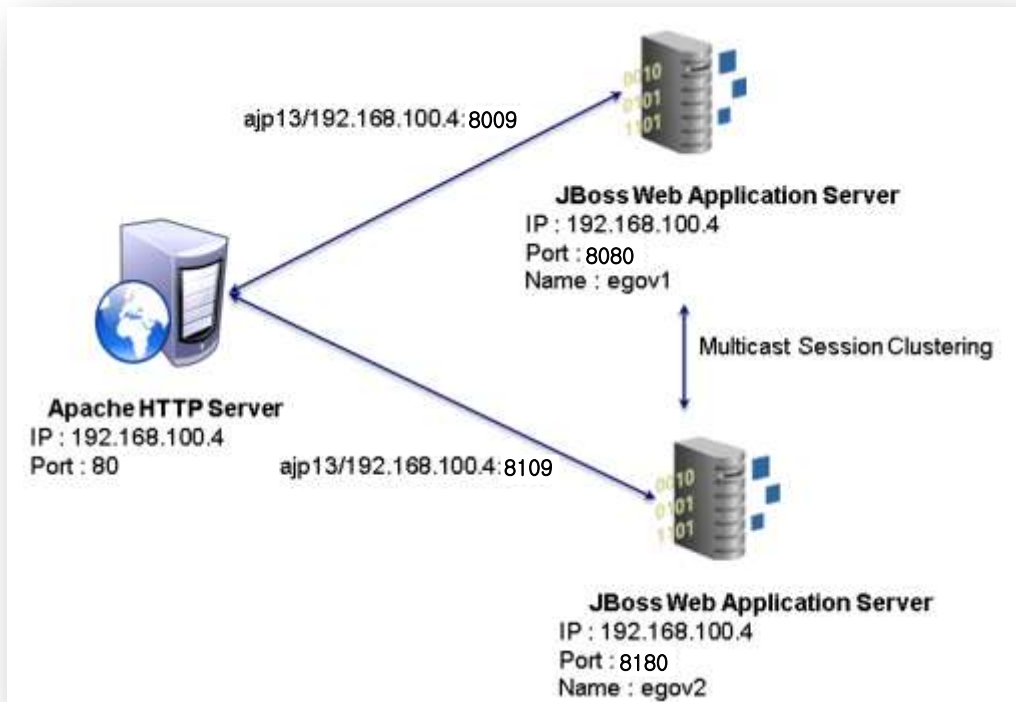
[https://access.redhat.com/knowledge/docs/en-US/JBoss\\_Enterprise\\_Application\\_Platform/6/html/Administration\\_and\\_Configuration\\_Guide/chap-Application\\_Deployment.html](https://access.redhat.com/knowledge/docs/en-US/JBoss_Enterprise_Application_Platform/6/html/Administration_and_Configuration_Guide/chap-Application_Deployment.html)

## 5.4 Apache Http + JBoss EAP 6.0 세션 클러스터링 구성

Apache HTTP(Application Platform 6.0.1 Apache HTTP Server for RHEL 6 x86\_64) + mod\_jk.so(Application Platform 6.0.1 Webserver Connector Natives for RHEL 6 x86\_64) + JBoss EAP 6.0.x 환경에서 JBoss 와 Apache HTTP 서버 간의 세션 클러스터링 및 로드 밸런싱 환경을 구성한다.

### 5.4.1 개요

JBoss WAS 의 Http Session Clustering 위해서 아래의 그림과 같은 구성을 하였다.  
단일 서버에서 HTTP 서버 1 개와 JBoss 인스턴스 2 개를 실행시켰다.



아래의 표에서 JBoss Http Session Clustering 구성에 필요한 파일과 위치를 확인 할 수 있다.

구분	파일명	위치	설명
Apache	httpd.conf	/jboss-ews-2.0/httpd/conf	RHEL 6 이상일 경우에는 추가할 내용 없음
	mod_jk.so	/jboss-ews-2.0/httpd/modules	다운로드 후

			\$APACHE_HOME/modules 디렉터리로 복사
	mod_jk.conf	/jboss-ews-2.0/httpd/conf.d	mod_jk 모듈을 로드하고 worker 파일 및 uri mapping 파일을 설정
	workers.properties	/jboss-ews-2.0/httpd/conf.d	JBoss 인스턴스별로 worker 를 설정
	uriworkermapping.properties	/jboss-ews-2.0/httpd/conf.d	URL 과 Worker 맵핑
JBoss	Start shell	/was/jboss-eap-6.0/srv01/start.sh	추가할 내용 없음
	standalone*.xml	/was/jboss-eap-6.0/srv01/ configuration/ standalone*.xml	jvmRoute 의 이름과 maxThreads 수 조정
	web.xml	\$Application/WEB-INF/	web.xml 파일에 <distributable/>를 추가

## 5.4.2 Apache 설정

### 5.4.2.1 Linux Version 확인

#### Linux Version 확인

```
[root@localhost ~]# uname -a
Linux localhost.localdomain 2.6.18-238.el5 #1 SMP Sun Dec 19 14:22:44 EST 2010 x86_64 x86_64 x86_64 GNU/Linux
[root@localhost ~]# cat /etc/redhat-release
Red Hat Enterprise Linux Server release 5.6 (Tikanga)
```

### 5.4.2.2 Apache 설치

<https://access.redhat.com/jbossnetwork/restricted/listSoftware.html?downloadType=distributions&product=appplatform&version=6.0.1> 에서 "Application Platform 6.0.1 Apache HTTP Server for RHEL 6 x86\_64"를 다운로드한다.

"/" 디렉토리에 다운로드 한 "Application Platform 6.0.1 Apache HTTP Server for RHEL 6 x86\_64"을 설치한다.

#### Red Hat Linux 에서 Apache 설치

```
[root@localhost /]# unzip jboss-ews-httpd-2.0.0-RHEL6-x86_64.zip
[root@localhost /]# cd jboss-ews-2.0/httpd
[root@localhost httpd]# pwd
/jboss-ews-2.0/httpd
[root@localhost httpd]# . postinstall
```

### 5.4.2.3 Mod\_jk 다운로드

<https://access.redhat.com/jbossnetwork/restricted/listSoftware.html?downloadType=distributions&product=appplatform&version=6.0.1> 에서 "Application Platform 6.0.1 Webserver Connector Natives for RHEL 6 x86\_64"를 다운로드 한다.

다운로드된 "jboss-eap-native-webserver-connectors-6.0.1-RHEL6-x86\_64.zip" 파일의  
"/module/native/lib64/httpd/modules/mod\_jk.so" 파일을 "/jboss-ews-2.0/httpd/modules" 디렉토리에  
복사한다.

```
[root@localhost modules]#cp /module/native/lib64/httpd/modules/mod_jk.so .
```

### 5.4.2.4 mod\_jk.conf 파일 작성

"/jboss-ews-2.0/httpd/conf.d" 디렉토리에 "mod\_jk.conf" 파일을 다음과 같은 내용으로 작성한다.

#### mod\_jk.conf 파일 내용

```
<IfModule !mod_jk.c>
  LoadModule jk_module modules/mod_jk.so
</IfModule>

# jk 로그파일 위치 지정
JkLogFile logs/mod_jk.log
# jk 로그레벨 설정 [debug|error|info]   연동이 잘 안되면 로그 레벨을 debug 로 설정해서 로그를 확인할 것
JkLogLevel info
# jk 로그 포맷 설정
JkLogStampFormat "[%a %b %d %H:%M:%S %Y] "
JkRequestLogFormat "%w %V %T"

# JkOptions 지정
JkOptions +ForwardKeySize +ForwardURICompatUnparsed -ForwardDirectories

# worker 파일 위치 지정, 아래는 APACHE_HOME/conf.d 아래에 위치하는 것으로 가정
JkWorkersFile conf.d/workers.properties

# uri 매핑 파일 위치 지정, 아래는 APACHE_HOME/conf.d 아래에 위치하는 것으로 가정
JkMountFile conf.d/uriworkermmap.properties

# 로드 밸런싱을 위해 shared memory 설정
# SELinux 를 사용할 경우에는 문제가 발생할 수 있으므로, 이 때는 run/jk.shm 으로 변경
# https://bugzilla.redhat.com/show_bug.cgi?id=225452

JkShmFile logs/jk.shm
```

```
# Add jkstatus for managing runtime data
<Location /jkstatus/>
    JkMount jkstatus
    Order deny,allow
#    Deny from all
#    Allow from 127.0.0.1
</Location>
```

#### 5.4.2.5 workers.properties 파일 작성

"/jboss-ews-2.0/httpd/conf.d" 디렉토리에서 "workers.properties" 파일을 다음과 같은 내용으로 작성한다.

##### workers.properties 파일 내용

```
worker.list=wlb,jkstatus

worker.node01.host=192.168.100.4
worker.node01.port=8009
worker.node01.type=ajp13
worker.node01.connect_timeout=5000
worker.node01.prepost_timeout=5000
worker.node01.socket_keepalive=True
worker.node01.socket_timeout=10
#worker.node01.connection_pool_size=25
worker.node01.connection_pool_timeout=600

worker.node02.host=192.168.100.4
worker.node02.port=8109
worker.node02.type=ajp13
worker.node02.connect_timeout=5000
worker.node02.prepost_timeout=5000
worker.node02.socket_keepalive=True
worker.node02.socket_timeout=10
#worker.node02.connection_pool_size=25
worker.node02.connection_pool_timeout=600

# jk 1.2.7 이전 버전에서는 balanced_workers 를 사용

# balance_workers 에 명시된 worker 는 worker.list 에는 명시하지 않아야 함

worker.wlb.type=lb
worker.wlb.balance_workers=node01,node02
worker.wlb.method=Session
worker.wlb.sticky_session=True
```



```
worker.jkstatus.type=status
```

#### 5.4.2.6 uriworkermapping.properties 파일 작성

"/jboss-ews-2.0/httpd/conf.d" 디렉토리에서 "uriworkermapping.properties" 파일을 다음과 같은 내용으로 작성한다.

##### uriworkermapping.properties 파일 내용

```
# /myapp 와 그 하위를 wlb worker 로 매핑

/enterprise=wlb
/enterprise/*=wlb

# /jmx-console 과 그 하위를 wlb worker 로 매핑 - /*

/jmx-console/*=wlb

# /myapp 아래의 *.jpg 는 웹서버에 있으므로 매핑에서 제외

!/enterprise/*.jpg=wlb

# *.jsp 와 *.do 는 wlb worker 로 매핑

/*.jsp=wlb
/*.do=wlb
```

### 5.4.3 jboss 설정

#### 5.4.3.1 Env 스크립트에서 Binding Port Set 을 설정

##### Env 스크립트에 JBoss 인스턴스별 Binding Port 에 대한 JVM 옵션 추가

```
-Djboss.socket.binding.port-offset=0
```

같은 장비 (즉 같은 IP) 에서 여러 개의 JBoss 인스턴스를 실행하는 경우 Port Set 을 달리하여 Port 가 충돌되지 않도록 설정한다.

예를 들자면 srv01 인스턴스는 "0" 로 하고 srv2 인스턴스는 "100" 로 설정 한다.

### Binding Port Set 설정 후 JBoss 실행시 확인

```
09:06:47,594 INFO [org.apache.coyote.http11.Http11Protocol] (MSC service thread 1-2) Starting Coyote HTTP/1.1 on http-localhost/127.0.0.1:8080
09:06:47,763 INFO [org.apache.coyote.ajp.AjpProtocol] (MSC service thread 1-2) Starting Coyote AJP/1.3 on ajp-localhost/127.0.0.1:8009
.....
09:06:54,766 INFO [org.jboss.as] (Controller Boot Thread) JBAS015961: Http management interface listening on http://127.0.0.1:9990/management
09:06:54,767 INFO [org.jboss.as] (Controller Boot Thread) JBAS015951: Admin console listening on http://127.0.0.1:9990
09:06:54,767 INFO [org.jboss.as] (Controller Boot Thread) JBAS015874: JBoss EAP 6.0.1.GA (AS 7.1.3.Final-redhat-4) started in 20023ms - Started 214 of 371 services (156 services are passive or on-demand)
```

### 5.4.3.2 CLI 를 이용한 jvmRoute 및 maxThreads 설정

#### Jboss-cli.sh

```
#!/bin/sh

. ./env.sh

export JAVA_OPTS="-Djava.awt.headless=false $JAVA_OPTS"

$JBoss_HOME/bin/jboss-cli.sh --controller=$CONTROLLER_IP:$CONTROLLER_PORT --connect $@
```

“\$JBoss\_Instance/jboss-cli.sh”를 이용하여 jvmRoute(instance-id)를 설정한다. jvmRoute(instance-id) 에 대한 이름을 지정하면 Session Cookie 값에 지정된 값이 저장되게 된다.

#### CLI 를 이용한 jvmRoute 설정

```
[jboss@localhost srv01]$ ./jboss-cli.sh
=====
JBoss_HOME=/was/jboss-eap-6.0
DOMAIN_BASE=/was/jboss-eap-6.0
SERVER_NAME=srv01
CONFIG_FILE=standalone-full-ha.xml
BIND_ADDR=192.168.100.4
PORT_OFFSET=0
MULTICAST_ADDR=230.1.0.1
CONTROLLER=192.168.100.4:9999
=====
## jvmRoute(instance-id 설정)
[standalone@192.168.100.4:9999 /] /subsystem=web:write-attribute(name=instance-id,value=node01)
{
  "outcome" => "success",
  "response-headers" => {
    "operation-requires-reload" => true,
```

```

    "process-state" => "reload-required"
  }
}
## 확인
[standalone@192.168.100.4:9999 /] /subsystem=web:write-attribute(name=instance-id,vread-resource(include-defaults=true)
{
  "outcome" => "success",
  "result" => {
    "default-virtual-server" => "default-host",
    "instance-id" => "node01",
    "native" => "false",
    "configuration" => {
      "container" => undefined,
      "static-resources" => undefined,
      "jsp-configuration" => undefined
    },
    "connector" => {
      "http" => undefined,
      "ajp" => undefined
    },
    "virtual-server" => {"default-host" => undefined}
  },
  "response-headers" => {"process-state" => "reload-required"}
}

```

세션 아이디의 형태는 다음과 같다.

"yp3bwjdBY0JMguf1-AFVHg\_\_.srv01"

AJP 1.3 Connector 에 대한 maxThreads 도 함께 지정해 두는 것을 권장한다.

### CLI 를 이용한 maxThreads 설정

```

[jboss@localhost srv01]$ ./jboss-cli.sh
=====
JBoss_HOME=/was/jboss-eap-6.0
DOMAIN_BASE=/was/jboss-eap-6.0
SERVER_NAME=srv01
CONFIG_FILE=standalone-full-ha.xml
BIND_ADDR=192.168.100.4
PORT_OFFSET=0
MULTICAST_ADDR=230.1.0.1
CONTROLLER=192.168.100.4:9999
=====
## maxThreads(max-connections 설정)
[standalone@192.168.100.4:9999 /] /subsystem=web/connector=ajp:write-attribute(name=max-connections,value=256)
{

```

```

"outcome" => "success",
"response-headers" => {
  "operation-requires-reload" => true,
  "process-state" => "reload-required"
}
}
## 확인
[standalone@192.168.100.4:9999 /] /subsystem=web/connector=ajp:read-resource(includeinclude-defaults=true)
{
  "outcome" => "success",
  "result" => {
    "enable-lookups" => false,
    "enabled" => true,
    "executor" => undefined,
    "max-connections" => 256,
    "max-post-size" => 2097152,
    "max-save-post-size" => 4096,
    "name" => "ajp",
    "protocol" => "AJP/1.3",
    "proxy-name" => undefined,
    "proxy-port" => undefined,
    "redirect-port" => 8433,
    "scheme" => "http",
    "secure" => false,
    "socket-binding" => "ajp",
    "ssl" => undefined,
    "virtual-server" => undefined
  },
  "response-headers" => {"process-state" => "reload-required"}
}

```

#### 5.4.3.3 Webapp session replication 을 위한 web.xml 수정

Web Application 의 HTTP 세션 복제를 설정하려면 web.xml 파일에 <distributable/>를 추가한다.

##### Session Clustering 위하여 web.xml 수정

```

<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="http://java.sun.com/xml/ns/javaee" xmlns:web="http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd"
  xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd"
  id="WebApp_ID" version="2.5">
  <display-name>JBoss Sample Web App</display-name>
  <welcome-file-list>
    <welcome-file>/index.jsp</welcome-file>
  </welcome-file-list>

```

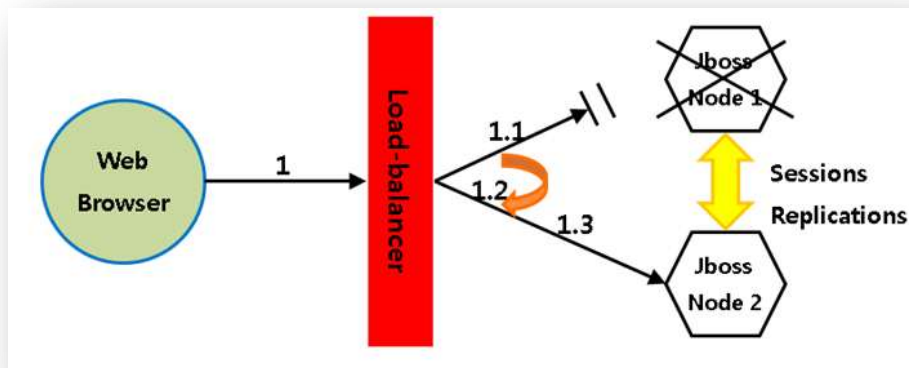
```
<distributable/>
</web-app>
```

## 5.5 Session Clustering 구성환경 점검

### 5.5.1 Session clustering 검증 방법 및 어플리케이션

jkstatus 를 통해 기본적인 구성에 대한 정보를 확인 할 수 있다.

<http://192.168.100.4/jkstatus/>



#### 5.5.1.1 Session Clustering 검증을 위한 어플리케이션

세션 테스트를 위한 Application 을 구성한 모든 인스턴스에 deploy 하였다.

세션 Tracking JSP 파일은 모든 요청에 대해 카운터 숫자를 하나씩 증가시키며, Session ID 를 웹페이지와 console 로그에 출력하도록 작성된 application 이다.

```
<html>
<head><title>Session Tracking Test</title>
</head>
<body>
<h1>Session Tracking Test</h1>
Session tracking with JSP is easy
<P>
<%@ page session="true" %>
<%
// Get the session data value
```

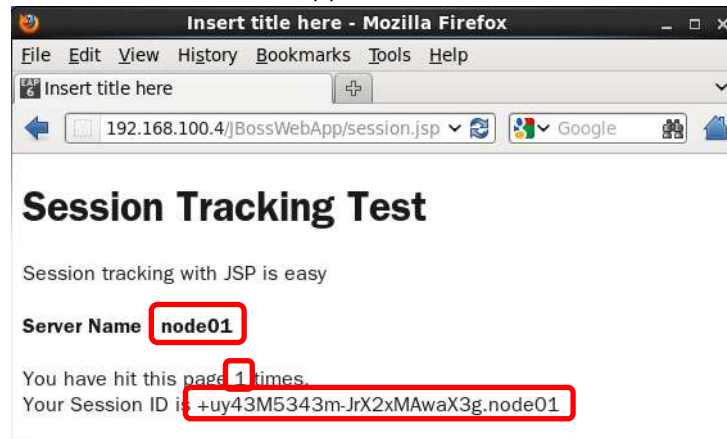
```
Integer ival = (Integer) session.getValue ("counter");
if (ival == null) ival = new Integer (1);
else ival = new Integer (ival.intValue() + 1);
session.putValue ("counter", ival);
%>
<h4>Server Name : <%=session.getId().substring(session.getId().indexOf(".") + 1) %></h4>
You have hit this page <%= ival %> times.<br>
<%
    out.println("Your Session ID is " + session.getId() + "<br>");
    System.out.println("session=" + session.getId() + ", counter=" + ival);
%>
</body></html>
```

## 5.5.2 Session Clustering Test

Session Clustering 확인을 위하여 아래의 URL 을 호출하여 그결과를 확인한다.

<http://192.168.100.4/JBossWebApp/session.jsp>

세션 Tracking JSP(session.jsp) 파일은 모든 요청에 대해 카운터 숫자를 하나씩 증가시키며, Session ID 를 웹페이지와 console 로그에 출력하도록 작성된 application 이다.



세션 Tracking JSP 파일은 모든 요청에 대해 카운터 숫자를 하나씩 증가시키며, Session ID 를 웹페이지와 console 로그에 출력하도록 작성된 application 이다.

아래의 그림에서처럼 접속된 JBoss WAS 인스턴스가 변경되더라도 counter 숫자는 1 씩 증가되는 것을 확인 하면 된다.



## 5.6 Hot deployment

### 5.6.1 hot deployment 중지 하기

Deployment Scanner 는 스캐닝을 실행하기 때문에 특히 느린 파일 시스템에서는 시스템의 속도가 매우 느려지게 된다.

#### CLI 를 이용한 hot deployment 중지 설정

```
[jboss@localhost srv01]$ ./jboss-cli.sh
=====
JBoss_HOME=/was/jboss-eap-6.0
DOMAIN_BASE=/was/jboss-eap-6.0
SERVER_NAME=srv01
CONFIG_FILE=standalone-full-ha.xml
BIND_ADDR=192.168.100.4
PORT_OFFSET=0
MULTICAST_ADDR=230.1.0.1
CONTROLLER=192.168.100.4:9999
=====
## hot deployment 중지 설정
[standalone@192.168.100.4:9999 /] /subsystem=deployment-scanner/scanner=default:write-attribute(name=scan-
enabled,value=false)
{"outcome" => "success"}
## 확인
[standalone@192.168.100.4:9999 /] /subsystem=deployment-scanner/scanner=default:read-resource(recursive=true)
{
```

```
"outcome" => "success",
"result" => {
    "auto-deploy-exploded" => false,
    "auto-deploy-xml" => true,
    "auto-deploy-zipped" => true,
    "deployment-timeout" => 600,
    "path" => "deployments",
    "relative-to" => "jboss.server.base.dir",
    "scan-enabled" => false,
    "scan-interval" => 0
}
}
```

아래와 같은 슬리밍 툴을 이용하여 스캐닝 횟수를 줄이거나 전혀 하지 않는 등의 방법으로 이와 같은 문제를 크게 완화시킬 수 있다.

## 5.7 CXF Web Services 통합

### 5.7.1 개요

상세한 통합 내용은 "JBoss EAP 6.0 Development Guide"를 참조한다.

[https://access.redhat.com/knowledge/docs/en-US/JBoss\\_Enterprise\\_Application\\_Platform/6/html-single/Development\\_Guide/index.html#chap-JAX-WS\\_Web\\_Services](https://access.redhat.com/knowledge/docs/en-US/JBoss_Enterprise_Application_Platform/6/html-single/Development_Guide/index.html#chap-JAX-WS_Web_Services)

### 5.7.2 CXF 확인

CXF Module 를 "JBoss\_HOME/modules/org/jboss/ws/cxf/"에서 확인 할 수 있다.

#### CXF Module

```
[jboss@localhost cxf]$ pwd
/was/jboss-eap-6.0/modules/org/jboss/ws/cxf
[jboss@localhost cxf]$ ls -al
total 24
drwxr-xr-x. 6 jboss jboss 4096 Feb 12 17:24 .
drwxr-xr-x. 3 jboss jboss 4096 Feb 12 17:23 ..
drwxr-xr-x. 3 jboss jboss 4096 Feb 12 17:24 jbossws-cxf-client
drwxr-xr-x. 3 jboss jboss 4096 Feb 12 17:24 jbossws-cxf-factories
drwxr-xr-x. 3 jboss jboss 4096 Feb 12 17:24 jbossws-cxf-server
drwxr-xr-x. 3 jboss jboss 4096 Feb 12 17:24 jbossws-cxf-transports-httpserver
```



JBoss 구동 시 Log 에서도 CXF 를 확인 할수 있다.

```

server.log
.....
INFO [org.jboss.ws.common.management.AbstractServerConfig] (MSC service thread 1-1) JBoss Web Services - Stack CXF Server
4.0.6.GA-redhat-2
.....

```

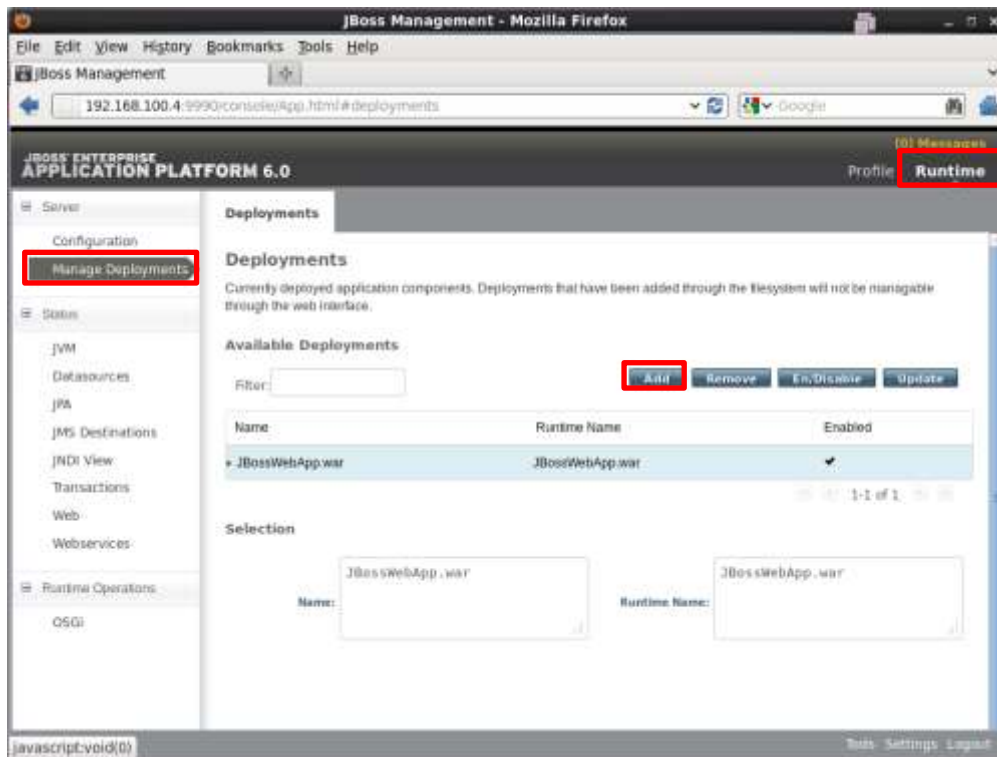
## 5.8 JBoss Datasource 설정

상세한 내용은 "JBoss EAP 6.0 Development Guide"를 참조한다.

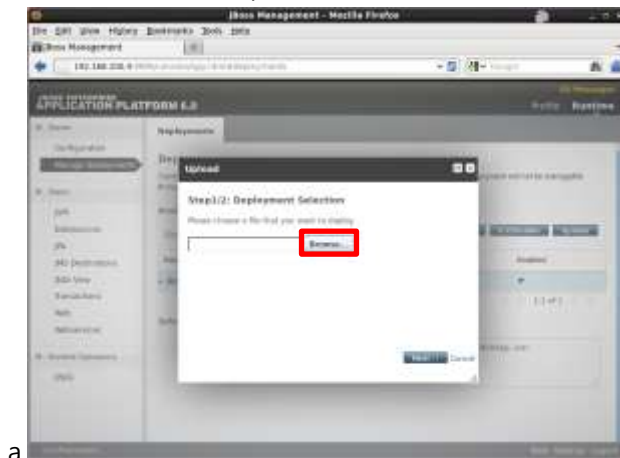
[https://access.redhat.com/knowledge/docs/en-US/JBoss\\_Enterprise\\_Application\\_Platform/6/html-single/Development\\_Guide/index.html#sect-Transactional\\_Datasource\\_Configuration](https://access.redhat.com/knowledge/docs/en-US/JBoss_Enterprise_Application_Platform/6/html-single/Development_Guide/index.html#sect-Transactional_Datasource_Configuration)

### 5.8.1 JDBC 드라이버 추가 및 등록

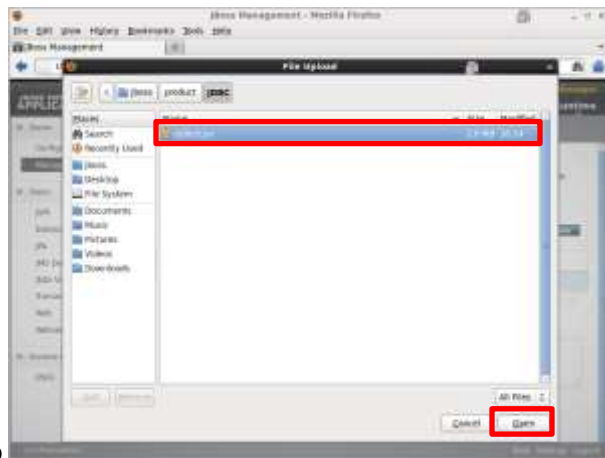
1. http://192.168.100.4:9990 접속
2. Add-user.sh 로 만든 관리자 계정으로 로그인
3. Runtime → Manage Deployments → add 클릭



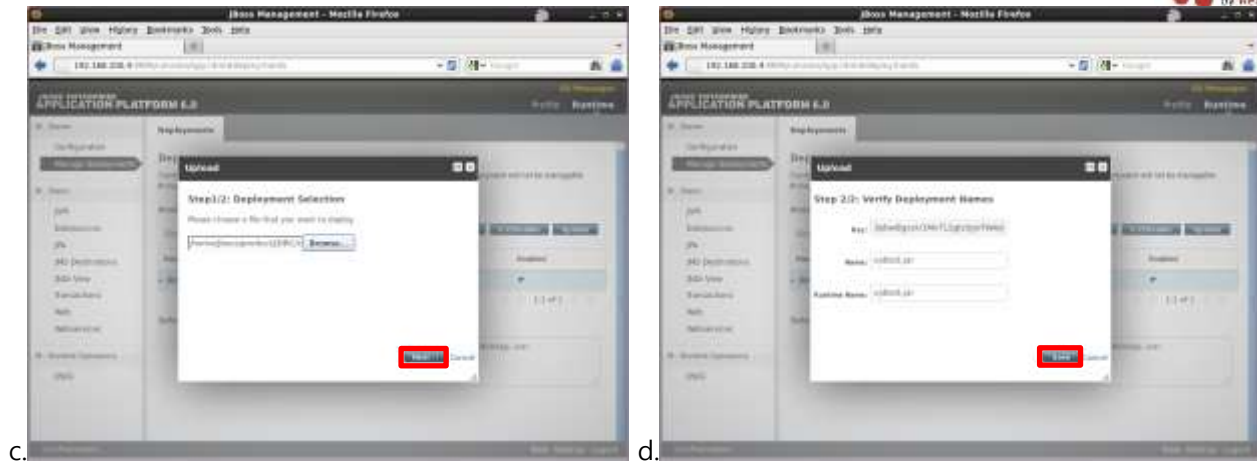
4. a. Browse...클릭
5. b. 등록할 JDBC 드라이버 선택 후 Open 클릭
6. c. Next 클릭
7. d. Save 클릭



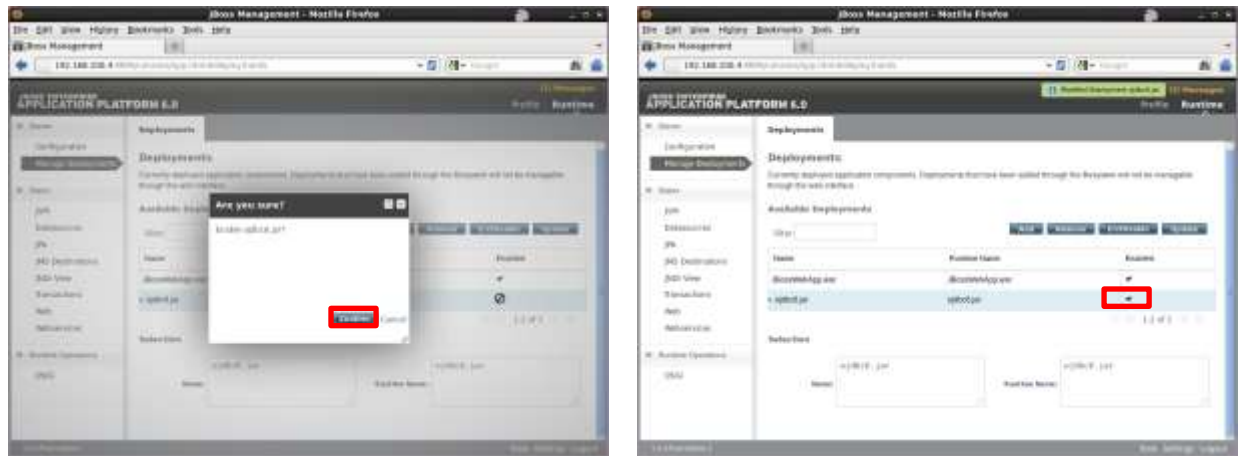
a.



b.

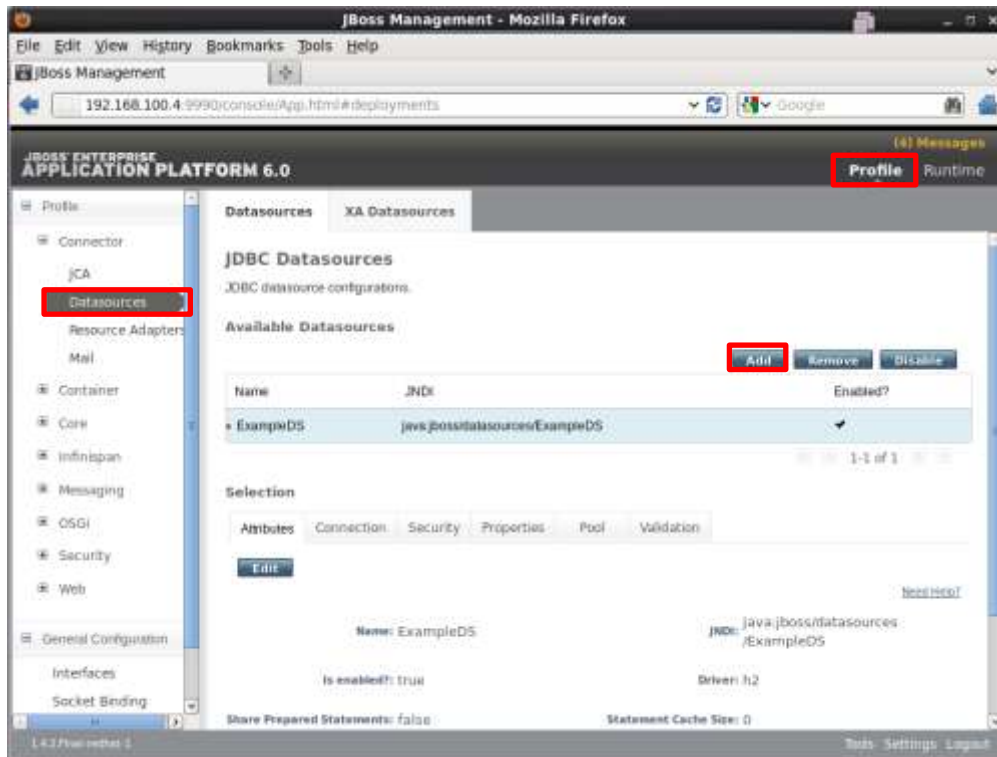


8. 등록된 JDBC 드라이버를 선택 → En/Disable 클릭 → Confirm 클릭 → Enabled V 확인

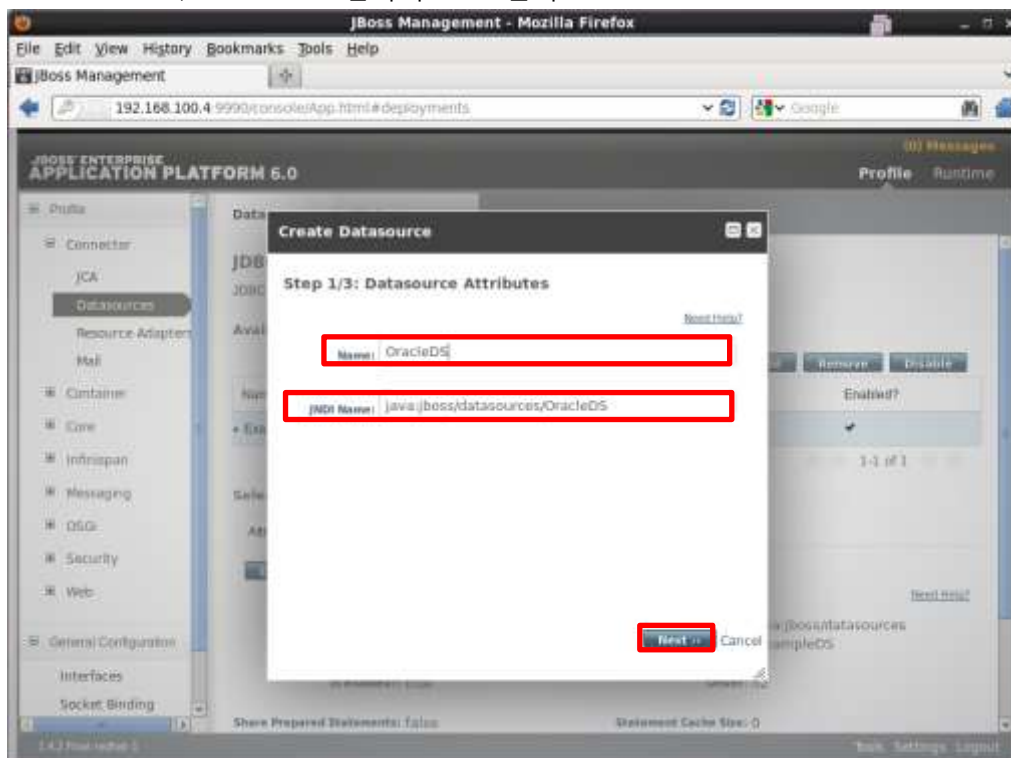


## 5.8.2 Data Source 추가 및 등록

1. <http://192.168.100.4:9990> 접속
2. Add-user.sh 로 만든 관리자 계정으로 로그인
3. Profile → Connector/Datasources → add 클릭



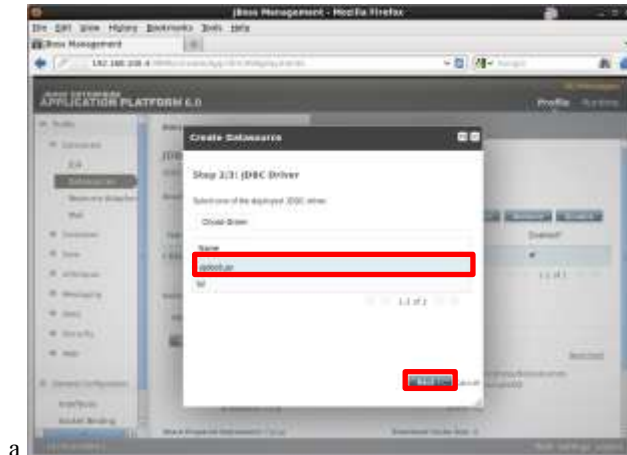
4. Name, JNDI Name 입력 후 Next 클릭



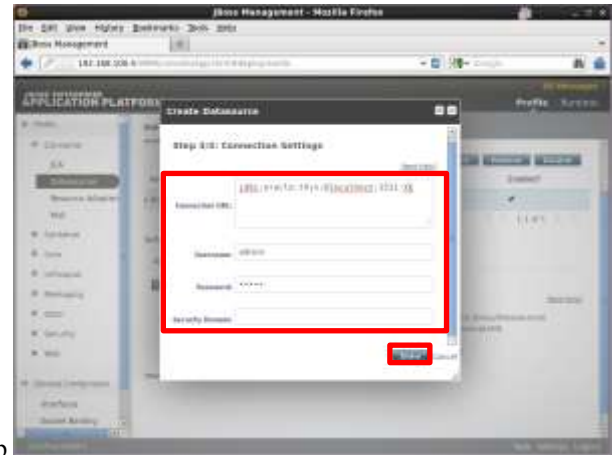
5. a. 등록할 JDBC 드라이버를 선택 후 Next 클릭

6. b. DB 정보를 입력 후 Done 클릭

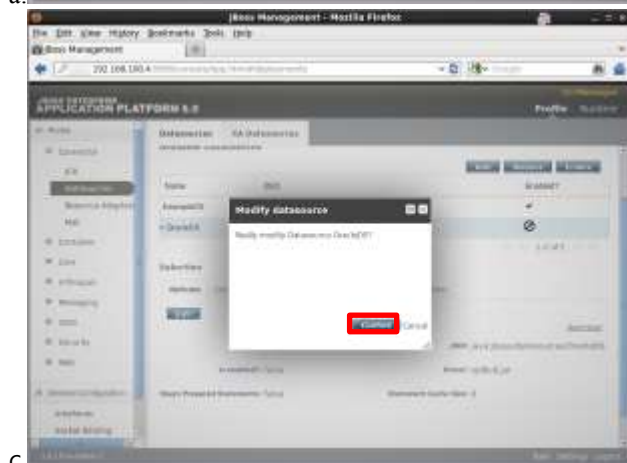
7. c. Confirm 클릭
8. Enabled V 확인



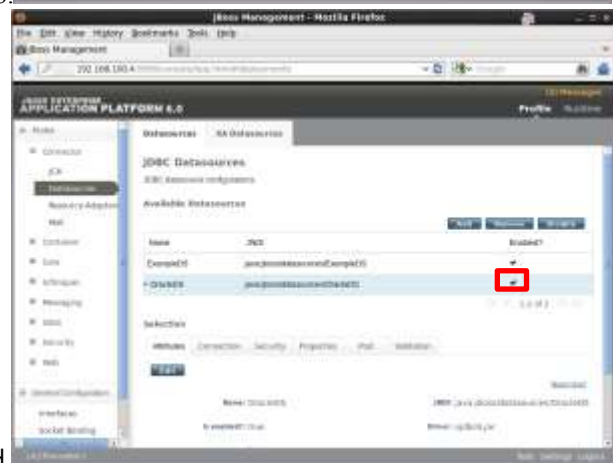
a.



b.



c.



d.

JDBC 드라이버 및 Datasource 등록 확인은 “/was/jboss-eap-6.0/srv01/configuration/standalone\*.xml” 에서도 확인 할수 있다.

### Standalone\*.xml 파일

```

.....
<subsystem xmlns="urn:jboss:domain:datasources:1.1">
  <datasources>
    <datasource jta="false" jndi-name="java:jboss/datasources/OracleDS" pool-name="OracleDS" enabled="true" use-ccm="false">
      <connection-url>jdbc:oracle:thin:@localhost:1521:XE</connection-url>
      <driver-class>oracle.jdbc.OracleDriver</driver-class>
      <driver>ojdbc6.jar</driver>
      <security>
        <user-name>admin</user-name>
        <password>admin</password>
      </security>
      <validation>

```

```

        <validate-on-match>false</validate-on-match>
        <background-validation>false</background-validation>
    </validation>
    <statement>
        <share-prepared-statements>false</share-prepared-statements>
    </statement>
</datasource>
</datasources>
</subsystem>
.....
<deployments>
    <deployment name="JBossWebApp.war" runtime-name="JBossWebApp.war">
        <content sha1="4625fc4ff8d4f043e9c2087e584acfe92c43d2f1"/>
    </deployment>
    <deployment name="ojdbc6.jar" runtime-name="ojdbc6.jar">
        <content sha1="de96f0060c7353532f4cb4a0b73d20545580fc91"/>
    </deployment>
</deployments>
.....

```

### 5.8.3 Data Source 를 이용한 DB 호출 샘플 JSP

#### JBoss Datasource Lookup 하는 JSP- testDBQueryWithDataSource.jsp

```

<%@ page contentType="text/html; charset=euc-kr" session="true" %>
<%@ page import="java.sql.*" %>
<%@ page import="javax.sql.*" %>
<%@ page import="javax.naming.*" %>
<html>
<head>
<meta http-equiv="content-type" content="text/html; charset=euc-kr">
<title>JDBC Test with Datasource</title>
</head>
<body>
<%

try {
    InitialContext ctx = new InitialContext();
    DataSource ds = (DataSource) ctx.lookup("java:/OracleDS");
    Connection conn = ds.getConnection();

    String sql = "select * from DUAL";
    Statement st=null;
    ResultSet rs=null;
    st=conn.createStatement();

```

```

rs=st.executeQuery(sql);
out.println("Connection Success <br>");
out.println(rs.getType());

while(rs.next())
{
    out.println("result Type : "+rs.getType()+"<br>");
    out.println("TABLE_NAME : " + rs.getString(1)+"<br>");
}

conn.close();
} catch (SQLException sqlE) {
    sqlE.printStackTrace();
} catch (Exception e) {
    out.println("Connection Fail <br>");
    out.println(e.getMessage());
}
%>
</body>
</html>

```

### 5.8.4 JDBC DataSource 설정시 JNDI 이름 지정

JBoss 에서 JDBC DataSource 설정시에 <jndi-name>DataSourceName</jndi-name>에 이름을 지정하고 이를 application 에서 JNDI 를 통해서 참조하게 되는데, WebLogic 의 경우에는 작성된 DataSourceName 으로 바로 참조할 수 있지만, JBoss 의 경우엔 java:DataSourceName 와 같이 앞에 java: 를 붙여야 참조할 수 있다.

WebLogic 과 같이 java: 없이 참조하려면 CLI 를 이용하여 Datasource 의 use-java-context 를 false 로 설정하면 된다.

#### CLI 를 이용한 use-java-context 설정

```

[jboss@localhost srv01]$ ./jboss-cli.sh
=====
JBoss_HOME=/was/jboss-eap-6.0
DOMAIN_BASE=/was/jboss-eap-6.0
SERVER_NAME=srv01
CONFIG_FILE=standalone-full-ha.xml
BIND_ADDR=192.168.100.4
PORT_OFFSET=0

```

```

MULTICAST_ADDR=230.1.0.1
CONTROLLER=192.168.100.4:9999
=====
## use-java-context 설정
[standalone@192.168.100.4:9999 /] /subsystem=datasources/data-source=OracleDS:write-attribute(name=use-java-
context,value=false)
{
  "outcome" => "success",
  "response-headers" => {
    "operation-requires-reload" => true,
    "process-state" => "reload-required"
  }
}
## 확인
[standalone@192.168.100.4:9999 /] /subsystem=datasources/data-source=OracleDS:read-resource(recursive=true)
{
  "outcome" => "success",
  "result" => {
    "allocation-retry" => undefined,
    "allocation-retry-wait-millis" => undefined,
    "allow-multiple-users" => undefined,
    "background-validation" => false,
    "background-validation-millis" => undefined,
    "blocking-timeout-wait-millis" => undefined,
    "check-valid-connection-sql" => undefined,
    "connection-properties" => undefined,
    "connection-url" => "jdbc:oracle:thin:@localhost:1521:XE",
    "datasource-class" => undefined,
    "driver-class" => "oracle.jdbc.OracleDriver",
    "driver-name" => "ojdbc6.jar",
    "enabled" => true,
    "exception-sorter-class-name" => undefined,
    "exception-sorter-properties" => undefined,
    "flush-strategy" => undefined,
    "idle-timeout-minutes" => undefined,
    "jndi-name" => "java:jboss/datasources/OracleDS",
    "jta" => false,
    "max-pool-size" => undefined,
    "min-pool-size" => undefined,
    "new-connection-sql" => undefined,
    "password" => "admin",
    "pool-prefill" => undefined,
    "pool-use-strict-min" => undefined,
    "prepared-statements-cache-size" => undefined,
    "query-timeout" => undefined,
    "reauth-plugin-class-name" => undefined,
  }
}

```



```

"reauth-plugin-properties" => undefined,
"security-domain" => undefined,
"set-tx-query-timeout" => false,
"share-prepared-statements" => false,
"spy" => false,
"stale-connection-checker-class-name" => undefined,
"stale-connection-checker-properties" => undefined,
"track-statements" => "NOWARN",
"transaction-isolation" => undefined,
"url-delimiter" => undefined,
"url-selector-strategy-class-name" => undefined,
"use-ccm" => false,
"use-fast-fail" => false,
"use-java-context" => false,
"use-try-lock" => undefined,
"user-name" => "admin",
"valid-connection-checker-class-name" => undefined,
"valid-connection-checker-properties" => undefined,
"validate-on-match" => false,
"statistics" => {
    "jdbc" => undefined,
    "pool" => undefined
}
},
"response-headers" => {"process-state" => "reload-required"}
}

```

## 5.9 Spring 에서 JTA Transaction 설정

상세한 내용은 “JBoss EAP 6.0 Development Guide”를 참조한다.

[https://access.redhat.com/knowledge/docs/en-US/JBoss\\_Enterprise\\_Application\\_Platform/6/html-single/Development\\_Guide/index.html#chap-Java\\_Transaction\\_API\\_JTA](https://access.redhat.com/knowledge/docs/en-US/JBoss_Enterprise_Application_Platform/6/html-single/Development_Guide/index.html#chap-Java_Transaction_API_JTA)

### Spring 에서 Transaction Manager 설정

```

[jboss@localhost srv01]$ ./jboss-cli.sh
=====
JBoss_HOME=/was/jboss-eap-6.0
DOMAIN_BASE=/was/jboss-eap-6.0
SERVER_NAME=srv01

```

```

CONFIG_FILE=standalone-full-ha.xml
BIND_ADDR=192.168.100.4
PORT_OFFSET=0
MULTICAST_ADDR=230.1.0.1
CONTROLLER=192.168.100.4:9999
=====
## JTA 설정
[standalone@192.168.100.4:9999 /] /subsystem=datasources/data-source=oracleDS:write-attribute(name=jta,value=true)
{
  "outcome" => "success",
  "response-headers" => {
    "operation-requires-reload" => true,
    "process-state" => "reload-required"
  }
}
## 확인
[standalone@192.168.100.4:9999 /] /subsystem=datasources/data-source=oracleDS:read-resource(recursive=true)
{
  "outcome" => "success",
  "result" => {
    "allocation-retry" => undefined,
    "allocation-retry-wait-millis" => undefined,
    "allow-multiple-users" => undefined,
    "background-validation" => false,
    "background-validation-millis" => undefined,
    "blocking-timeout-wait-millis" => undefined,
    "check-valid-connection-sql" => undefined,
    "connection-properties" => undefined,
    "connection-url" => "aa",
    "datasource-class" => undefined,
    "driver-class" => "oracle.jdbc.OracleDriver",
    "driver-name" => "oracle",
    "enabled" => true,
    "exception-sorter-class-name" => undefined,
    "exception-sorter-properties" => undefined,
    "flush-strategy" => undefined,
    "idle-timeout-minutes" => undefined,
    "jndi-name" => "java:jboss/datasources/oracleDS",
    "jta" => true,
    "max-pool-size" => undefined,
    "min-pool-size" => undefined,
    "new-connection-sql" => undefined,
    "password" => "aa",
    "pool-prefill" => undefined,
    "pool-use-strict-min" => undefined,
    "prepared-statements-cache-size" => undefined,
  }
}

```

```

"query-timeout" => undefined,
"reauth-plugin-class-name" => undefined,
"reauth-plugin-properties" => undefined,
"security-domain" => "aa",
"set-tx-query-timeout" => false,
"share-prepared-statements" => false,
"spy" => false,
"stale-connection-checker-class-name" => undefined,
"stale-connection-checker-properties" => undefined,
"track-statements" => "NOWARN",
"transaction-isolation" => undefined,
"url-delimiter" => undefined,
"url-selector-strategy-class-name" => undefined,
"use-ccm" => false,
"use-fast-fail" => false,
"use-java-context" => false,
"use-try-lock" => undefined,
"user-name" => "aa",
"valid-connection-checker-class-name" => undefined,
"valid-connection-checker-properties" => undefined,
"validate-on-match" => false
},
"response-headers" => {"process-state" => "reload-required"}
}

```

## 5.10 jconsole 을 이용한 JMX 모니터링

JDK 1.6 에 번들되어 있는 jconsole 을 이용하여 JBoss 서버의 JMX 서버를 통한 모니터링을 한다.

### 5.10.1 jconsole.sh 실행

#### jconsole.sh 스크립트

```

#!/bin/sh

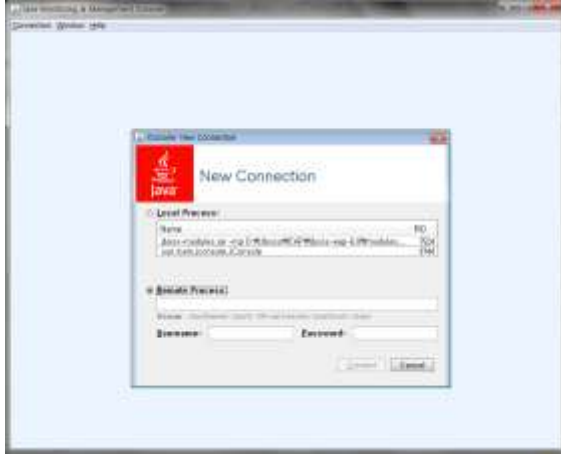
. ./env.sh

echo "=====
echo " JMX URL : service:jmx:remoting-jmx://$MGMT_ADDR:$CONTROLLER_PORT"
echo "=====
$JBoss_HOME/bin/jconsole.sh

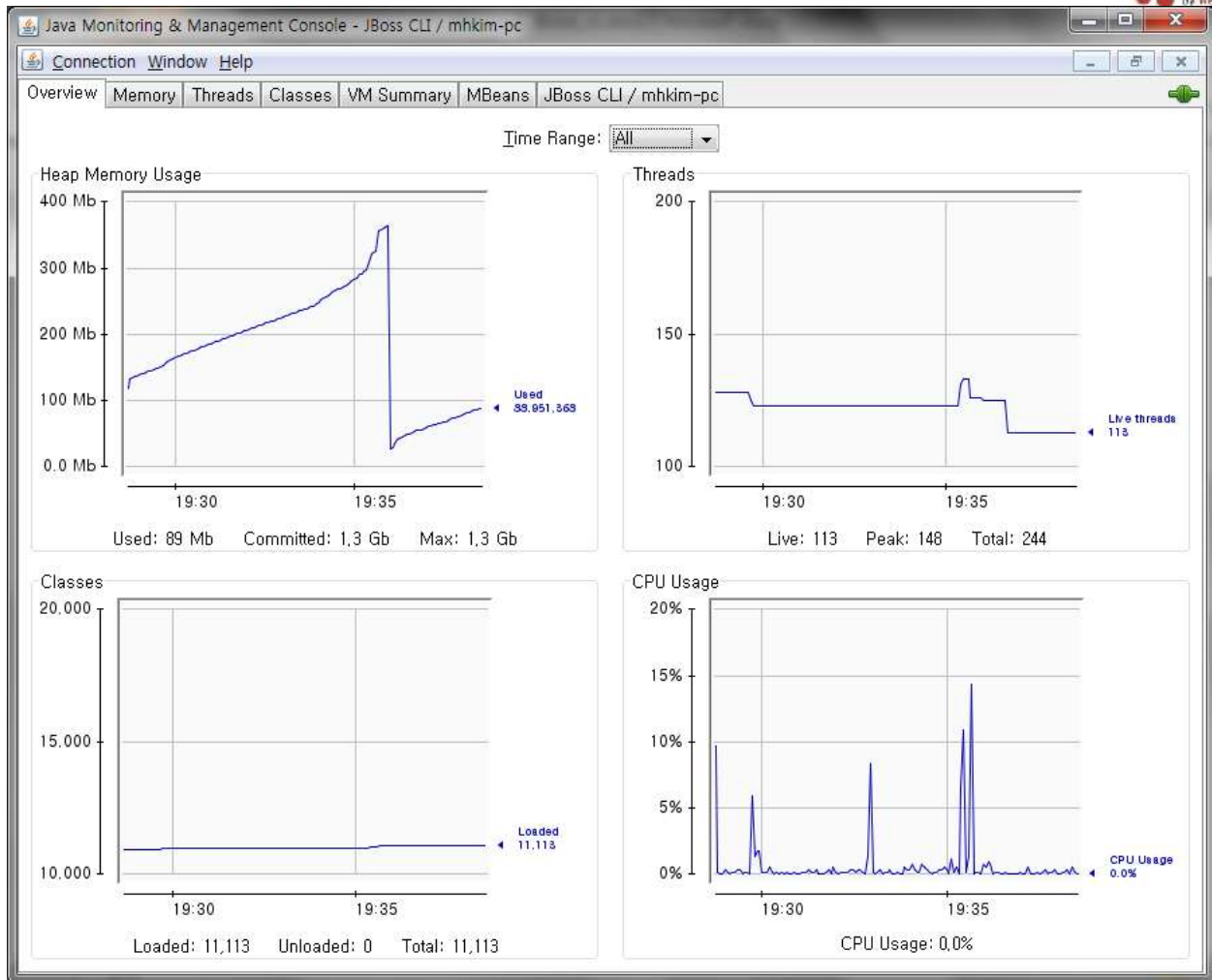
```

### 5.10.2 jconsole 을 실행한 후 연결

jconsole 을 실행후 "Remote Process"를 선택 후 "service:jmx:remoting-jmx://192.168.100.4:9999", Username, Password 를 입력하고 "OK" 버튼을 누른다.



jconsole 을 이용하여 System 리소스 및 MBeans, JBoss CLI 정보 및 설정을 할 수 있다.



## 5.11 Logging

상세한 내용은 "JBoss EAP 6.0 Development Guide"를 참조한다.

[https://access.redhat.com/knowledge/docs/en-US/JBoss\\_Enterprise\\_Application\\_Platform/6/html-single/Development\\_Guide/index.html#chap-Logging\\_for\\_Developers](https://access.redhat.com/knowledge/docs/en-US/JBoss_Enterprise_Application_Platform/6/html-single/Development_Guide/index.html#chap-Logging_for_Developers)

JBoss의 Logging 정보는 CLI를 통하여 확인할 수 있다.

### CLI로 로그 설정하기

```
[jboss@localhost srv01]$ ./jboss-cli.sh
```

```
=====
```

```
JBoss_HOME=/was/jboss-eap-6.0
```

```
DOMAIN_BASE=/was/jboss-eap-6.0
```

```

SERVER_NAME=srv01
CONFIG_FILE=standalone-full-ha.xml
BIND_ADDR=192.168.100.4
PORT_OFFSET=0
MULTICAST_ADDR=230.1.0.1
CONTROLLER=192.168.100.4:9999
=====
## 확인
[standalone@192.168.100.4:9999 /] /subsystem=logging:read-resource(recursive=true)
{
  "outcome" => "success",
  "result" => {
    "async-handler" => undefined,
    "console-handler" => undefined,
    "custom-handler" => undefined,
    "file-handler" => undefined,
    "size-rotating-file-handler" => undefined,
    "logger" => {
      "com.arjuna" => {
        "filter" => undefined,
        "handlers" => undefined,
        "level" => "WARN",
        "use-parent-handlers" => true
      },
      "org.apache.tomcat.util.modeler" => {
        "filter" => undefined,
        "handlers" => undefined,
        "level" => "WARN",
        "use-parent-handlers" => true
      },
      "sun.rmi" => {
        "filter" => undefined,
        "handlers" => undefined,
        "level" => "WARN",
        "use-parent-handlers" => true
      },
      "jacob" => {
        "filter" => undefined,
        "handlers" => undefined,
        "level" => "WARN",
        "use-parent-handlers" => true
      },
      "jacob.config" => {
        "filter" => undefined,
        "handlers" => undefined,
        "level" => "ERROR",

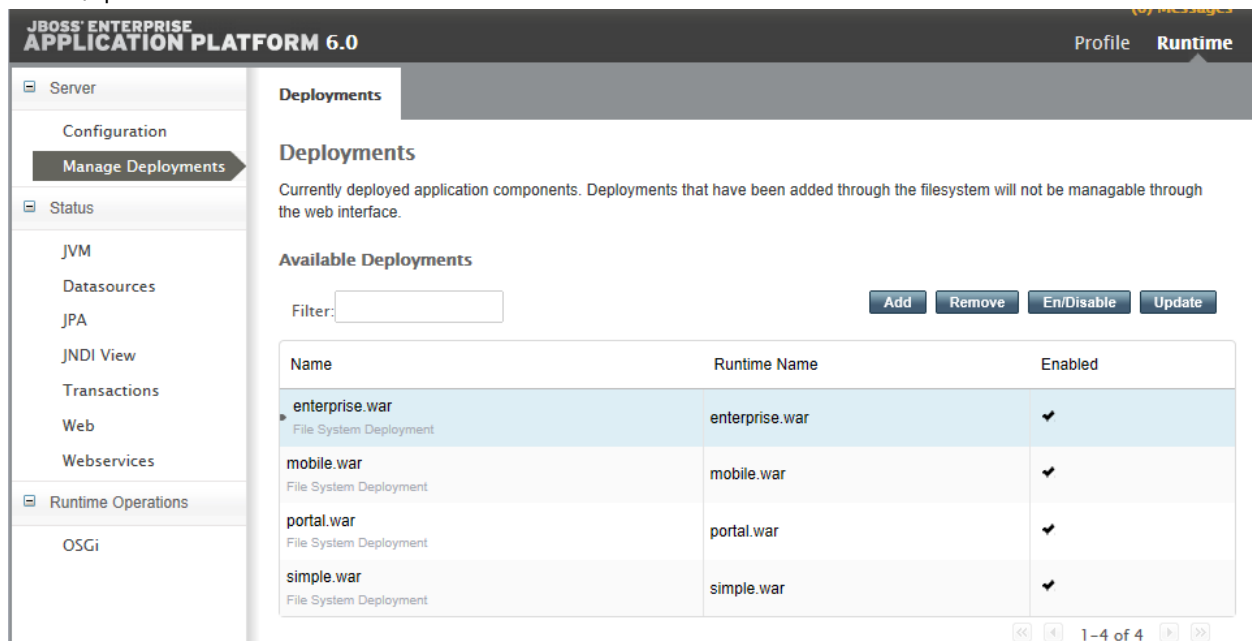
```

```
"use-parent-handlers" => true
}
},
"periodic-rotating-file-handler" => {"FILE" => {
  "append" => true,
  "autoflush" => true,
  "encoding" => undefined,
  "file" => {
    "relative-to" => "jboss.server.log.dir",
    "path" => "server.log"
  },
  "filter" => undefined,
  "formatter" => "%d{HH:mm:ss,SSS} %-5p [%c] (%t) %s%E%n",
  "level" => undefined,
  "suffix" => ".yyyy-MM-dd"
}},
"root-logger" => {"ROOT" => {
  "filter" => undefined,
  "handlers" => ["FILE"],
  "level" => "INFO"
}}
},
"response-headers" => {"process-state" => "reload-required"}
}
```

## 6. 전자정부 표준프레임워크 응용 어플리케이션 운영 환경 검증

<http://192.168.100.4:9990/console/> , <http://192.168.100.4:10090/console/>

admin/rplinux



**JBoss Enterprise Application Platform 6.0**

Profile Runtime

Server

Configuration

Manage Deployments

Status

JVM

Datasources

JPA

JNDI View

Transactions

Web

Webservices

Runtime Operations

OSGi

**Deployments**

Currently deployed application components. Deployments that have been added through the filesystem will not be manageable through the web interface.

**Available Deployments**

Filter:

Add Remove En/Disable Update

Name	Runtime Name	Enabled
enterprise.war File System Deployment	enterprise.war	✓
mobile.war File System Deployment	mobile.war	✓
portal.war File System Deployment	portal.war	✓
simple.war File System Deployment	simple.war	✓

1-4 of 4

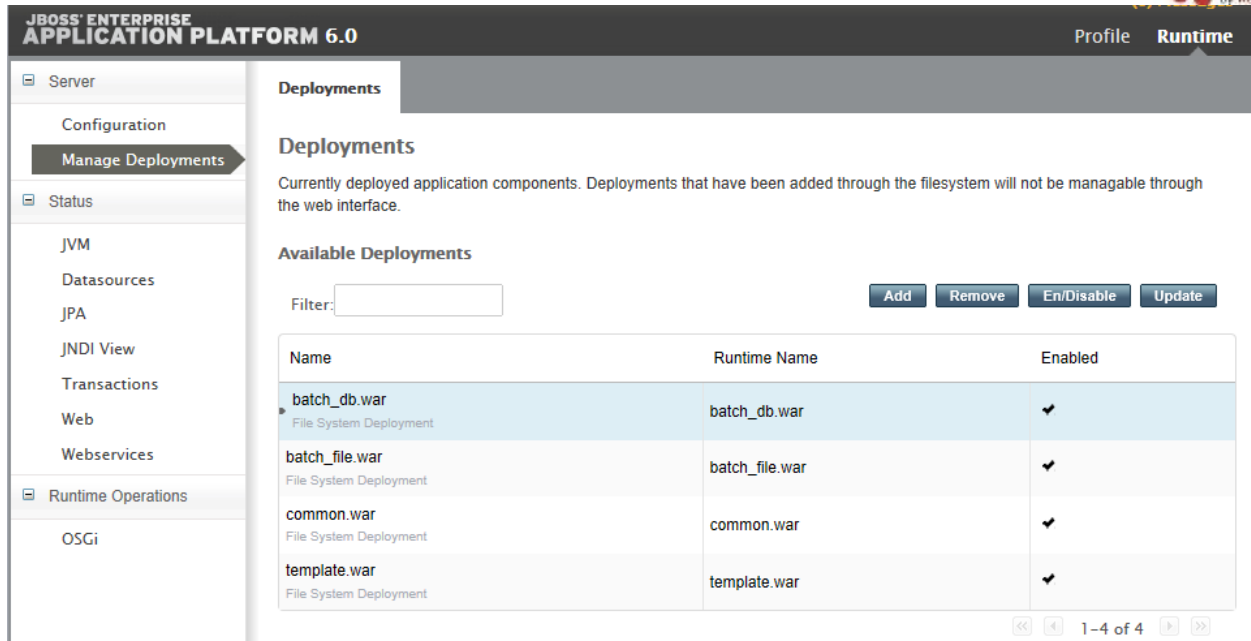
<http://192.168.100.4:8080/simple>

<http://192.168.100.4:8080/portal>

<http://192.168.100.4:8080/enterprise>

<http://192.168.100.4:8080/mobile>





**JBoss ENTERPRISE APPLICATION PLATFORM 6.0** Profile Runtime

**Deployments**

Currently deployed application components. Deployments that have been added through the filesystem will not be manageable through the web interface.

**Available Deployments**

Filter:  Add Remove En/Disable Update

Name	Runtime Name	Enabled
batch_db.war File System Deployment	batch_db.war	✓
batch_file.war File System Deployment	batch_file.war	✓
common.war File System Deployment	common.war	✓
template.war File System Deployment	template.war	✓

1-4 of 4

<http://192.168.100.4:8180/template/>

[http://192.168.100.4:8180/batch\\_db](http://192.168.100.4:8180/batch_db)

[http://192.168.100.4:8180/batch\\_file](http://192.168.100.4:8180/batch_file)

<http://192.168.100.4:8180/common>

## 6.1 eGovFrame 운영 환경 어플리케이션 구성 및 배포

### 6.1.1 JBoss EAP 6 에서 valang.tld 파일 수정

JBoss 에서 개발환경은 편의상 JBoss EAP 6.0.1 이상의 버전에서 구성하는 것을 권장하며, 유의할 점은 "spring-modules-validation-0.9.jar/META-INF/valang.tld" 에 포함된 내용 중 잘못된 "body-content" 값이 있는 버그가 있다.

다음의 "spring-modules-validation-0.9.jar" 파일의 META-INF/valang.tld 파일에서 "None"을 "empty" 로 설정한다.

### valang.tld 파일 수정하기

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<!DOCTYPE taglib PUBLIC "-//Sun Microsystems, Inc.//DTD JSP Tag Library 1.2//EN" "http://java.sun.com/dtd/web-
jsptaglibrary_1_2.dtd">
<taglib>
  <tlib-version>1.1.1</tlib-version>
  <jsp-version>1.2</jsp-version>
  <short-name>Valang</short-name>
  <uri>http://www.springmodules.org/tags/valang</uri>
  <description>A set of tags that enable the translation of Valang
    validators into JavaScript.</description>
  <tag>
    <name>codebase</name>
    <tag-class>org.springmodules.validation.valang.javascript.taglib.ValangCodebaseTag</tag-class>
    <body-content>empty</body-content>
    <description>Generates the JavaScript code that is necessary for the
      translated JavaScript validation rules to execute. This code needs
      to be included once per page and before any generated validation
      rules.</description>
```

### 6.1.2 JBoss EAP 6 에서 pom.xml 파일 수정

Portal 어플리케이션에서는 jdom 의 dependency 를 POM 파일에서 아래와 같이 추가해 주어야 한다.

#### pom.xml 파일 수정하기

```
<dependency>
  <groupId>org.jdom</groupId>
  <artifactId>jdom</artifactId>
  <version>1.1</version>
</dependency>
```

### 6.1.3 JBoss EAP 6 에서 Spring 2.5 사용시 수정

만약 JBoss EAP 6.0.X 이상에서 Spring 2.5.X 을 사용하고자 하는 경우 아래와 같은 방법으로 적용해야 한다.

- Snowdrop binary download
  - <https://access.redhat.com> 로그인 -> Download -> JBOSS Enterprise Middleware Downloads -> Product : Web Framework Kit -> Version : 2.1.0 -> Web Framework Kit 2.1.0 Snowdrop Binary 다운로드
- Copy Snowdrop binary
  - jboss-wfk-2.1.0-snowdrop-bin.zip 압축해제.(아래의 것만 사용)  
|\_snowdrop-subsystem-as7 |\_module-deployer (org/jboss/snowdrop/main/\*)
  - module-deployer 밑의 폴더 구조 그대로 <JBOSS\_HOME>/modules 밑으로 복사 (<JBOSS\_HOME>/modules/org/jboss/snowdrop/main/\*)
- Snowdrop with Spring download
  - <http://www.jboss.org/snowdrop/downloads> -> 2.0.4.Final 버전의 With Spring 2.5, Spring 3 and Spring 3.1 modules 다운로드
- Copy Snowdrop with spring
  - jboss-wfk-2.1.0-snowdrop-bin.zip 압축해제.(아래의 것만 사용)  
|\_module-spring-2.5(org/springframework/spring/snowdrop/\*)
  - module-spring-2.5 밑의 폴더 구조 그대로 <JBOSS\_HOME>/modules 밑으로 복사 (<JBOSS\_HOME>/modules/org/springframework/spring/snowdrop/\*>)
- standalone.xml 수정

#### Standalone.xml 파일 수정하기

```
<extensions>
    .....
    <extension module="org.jboss.snowdrop"/> ==> 추가
</extensions>

<profile>
    <subsystem xmlns="urn:jboss:domain:snowdrop:1.0"/> ==> 추가
    .....
</profile>
```

- jboss-deployment-structure.xml 생성

#### WEB-INF/Jboss-deployment-structure.xml 파일 생성하기

```
<jboss-deployment-structure xmlns="urn:jboss:deployment-structure:1.0">
  <deployment>
    <dependencies>
      <module name="org.springframework.spring" slot="snowdrop"
        export="true">
        <exports>
```

```

        <include path="META-INF**" />
        <include path="org**" />
    </exports>
    <imports>
        <include path="META-INF**" />
        <include path="org**" />
    </imports>
</module>
<module name="org.jboss.snowdrop" export="true" />
</dependencies>
</deployment>
</jboss-deployment-structure>

```

- web.xml 수정

#### WEB-INF/web.xml 파일 수정하기

```

<context-param>
    <param-name>contextClass</param-name>
    <param-value>org.jboss.spring.vfs.context.VFSXMLWebApplicationContext</param-value>
</context-param>

<init-param>
    <param-name>contextClass</param-name>
    <param-value>org.jboss.spring.vfs.context.VFSXMLWebApplicationContext</param-value>
</init-param>

```

## 7. 주요 질문과 답변들

### 7.1.1 Basic 30-Day Trial 신청 방법

1. 어플리케이션 테스트를 위해 JBoss 평가판이 필요하다면,  
<https://www.redhat.com/solutions/industries/developer/products/basic.html> 을 통해 30-Day Trial 을 신청할 수 있다 .
2. 신청후 바로 Subscription No. 와 설치 안내 이메일이 발송된다.
3. 발급받은 Subscription No. 를 <https://www.redhat.com/apps/activate/> 에 등록한다.
4. RHN ([rhn.redhat.com](http://rhn.redhat.com)) 에서 JBoss EAP 와 그 밖의 JBoss 제품들의 ISOs(이미지 파일)를 다운로드 받아 테스트한다.
5. ISOs(이미지 파일)은 등록 후 30 일간 제공된다.

### 7.1.2 JBoss Community 버전과 Enterprise 버전의 차이점은 무엇인가요?

JBoss Enterprise Middleware 제품화 과정은 구조화된 일련의 단계이며, 이를 통해 다양한 오픈 소스 프로젝트의 코드가 잘 정리되고 완전한 기술 지원이 가능한 미들웨어 제품으로 탄생하게 된다. JBoss.org 프로젝트를 관리하는 팀과 JBoss Enterprise Middleware 제품을 제공하는 팀 간에 상당한 상호 작용이 발생하며 이러한 팀들은 각기 서로 다른 목표와 목적을 갖게 된다. JBoss.org 프로젝트 팀은 기술 수준을 더욱 발전시키고 광범위한 공헌자 커뮤니티를 관리하는 업무를 담당하게 되고, 반면에 JBoss Enterprise Middleware 팀은 일반적으로 4-5 개월이 걸리는 과정을 통해 완성된 소프트웨어 제품을 제공하는 역할을 맡고 있다. 이 과정의 목표는 다음과 같다.

- 어플리케이션을 개발하기 위한 기능적이고 안정적인 최고의 플랫폼을 엔터프라이즈 IT 조직과 ISV 에 제공

- IT 조직이 당면한 다량의 통합 업무, 테스트 업무, 기존에 오픈 소스 소프트웨어 사용과 관련된 위험 등을 제거
- 1 년 365 일 24 시간 제품 사용에 적합한 서비스 레벨 계약을 고객에게 제공

	JBoss.ORG 또는 다른 오픈 소스 프로젝트 소프트웨어	JBoss ENTERPRISE MIDDLEWARE 제품
라이선스 비용	없음	없음
소스 코드에 대한 액세스	완전한 접근 가능	완전한 접근 가능
대상	누구나	<b>JBoss Enterprise Middleware 서브스크립션 고객</b>
개발	오픈소스 프로젝트 커뮤니티	오픈 소스 프로젝트 커뮤니티 및 JBoss Enterprise Middleware 제품화 팀
개선 및 향상	커뮤니티의 요구에 따른 임시 또는 실험적인 소프트웨어 변경	주요 릴리스, 소규모 릴리스, 패치 및 마이그레이션 도구를 포함한 구조화된 릴리스 과정
사용 설명서	프로젝트 구성 요소에 따라 다양	전문적이고 로컬라이즈된 전체 소프트웨어 설명서
품질	기능 테스트 및 일부 호환성 테스트	매우 다양한 테스트 등이 통합된 우수한 품질
안정성	끊임없는 혁신으로 프로젝트 구성 요소별로 다양	제품화 과정의 특징으로 인해 더욱 안정적. 누적 패치 릴리스로 안정성을 해치지 않고 결함 해결
로드 테스트	제한됨	완전한 로드 테스트 가능
현지화	커뮤니티의 관심도와 노력에 따라 달라짐	주요 언어에 대해 전문적으로 로컬라이즈됨
플랫폼 및 데이터베이스 호환성	인기 있는 플랫폼 하나에서 일반적으로 테스트 됨	16 개가 넘는 JVM/운영 체제 조합 및 주요 데이터베이스에서 인증
솔루션 통합	개별 오픈 소스 구성 요소만 해당	잘 규정된 제품을 만들기 위해 구성 요소 조합. 포괄적 솔루션을 만들기 위해 여러 제품을 함께 테스트
보안 범위	커뮤니티의 관심도에 따라 달라짐	Red Hat 보안 팀에서 문제를 식별 및 해결
교육	구성 요소에 따라 다양	전문적인 전체 제품 교육 제공
기술 지원	커뮤니티의 상호 지원만 가능 — <b>지원 계약 또는 SLA 없음</b>	고유한 SLA 요구 사항에 따라 지원 서브스크립션 사용 가능
지원 보증	없음	릴리스 날짜 이후로 5 년간 지원 제공
법적 보증	없음	코드 관련 지적 자산 문제로부터 고객을 보호

### 7.1.3 JBoss EAP 6.0.x 설치 지원 환경은 어떻게 되나요?

앞서의 내용에서 해당 내용을 참조한다.

다음 URL 에서 JBoss EAP 6.0.x 버전에서 인증된 운영환경에 대한 정보를 확인 할 수 있다.

<https://access.redhat.com/knowledge/ko/articles/297473>

#### 7.1.4 JBoss EAP 6.0.x 에서 표준 지원 현황은 어떻게 되나요?

앞서의 내용에서 해당 내용을 참조한다.

아래의 URL 에서 최신 표준 지원 현황에 대한 확인이 가능하다.

<https://access.redhat.com/knowledge/ko/articles/299283>

#### 7.1.5 JBoss EAP 6.0.x 의 라이선스 정책은 어떻게 되나요?

아래의 URL 에서 상세한 내용을 참조한다.

[http://www.redhat.com/f/pdf/licenses/Enterprise\\_Agr\\_Korea.pdf](http://www.redhat.com/f/pdf/licenses/Enterprise_Agr_Korea.pdf)

#### 7.1.6 JBoss EAP 6.0.x 의 문의는 어디로 하나요?

영업담당 : 펜타시스템 홍석모 부장 / Tel : 02-769-9722 / email : [smhong@penta.co.kr](mailto:smhong@penta.co.kr)

기술문의 : 펜타시스템 최현우 책임 / Tel : 02-769-9853 / email : [sensewoo@penta.co.kr](mailto:sensewoo@penta.co.kr)