

# アプリケーションサーバ **JBoss超入門** ～ 10分で始めるJBoss ～

株式会社 日立ソリューションズ  
OSSソリューションビジネス推進センタ  
山本 慎悟

# Contents

- 1. 自己紹介**
- 2. JBoss 概要**
- 3. JBossのインストールおよび初期設定**
- 4. デモ (10分でセットアップ)**
- 5. 日立ソリューションズのオープンソースソリューションのご紹介**
- 6. まとめ**

## 2.JBoss概要

JBossはJava Web Application Server であるJBoss Application Server を中心としたオープンソースコミュニティプロジェクトです。

JBoss Application Server 以外にも複数のソフトウェアで構成されています。

単にJBossというと、JBoss Application Server を指していることが多いです。

JBoss のアプリケーションサーバにはコミュニティが提供している **JBoss Application Server (JBoss AS)**と、Red Hat が提供している **JBoss Enterprise Application Platform (JBoss EAP)**があります。

JBoss EAP はRed Hat がサブスクリプション契約に基づき提供している企業向けエンタープライズJBoss アプリケーションサーバとなります。

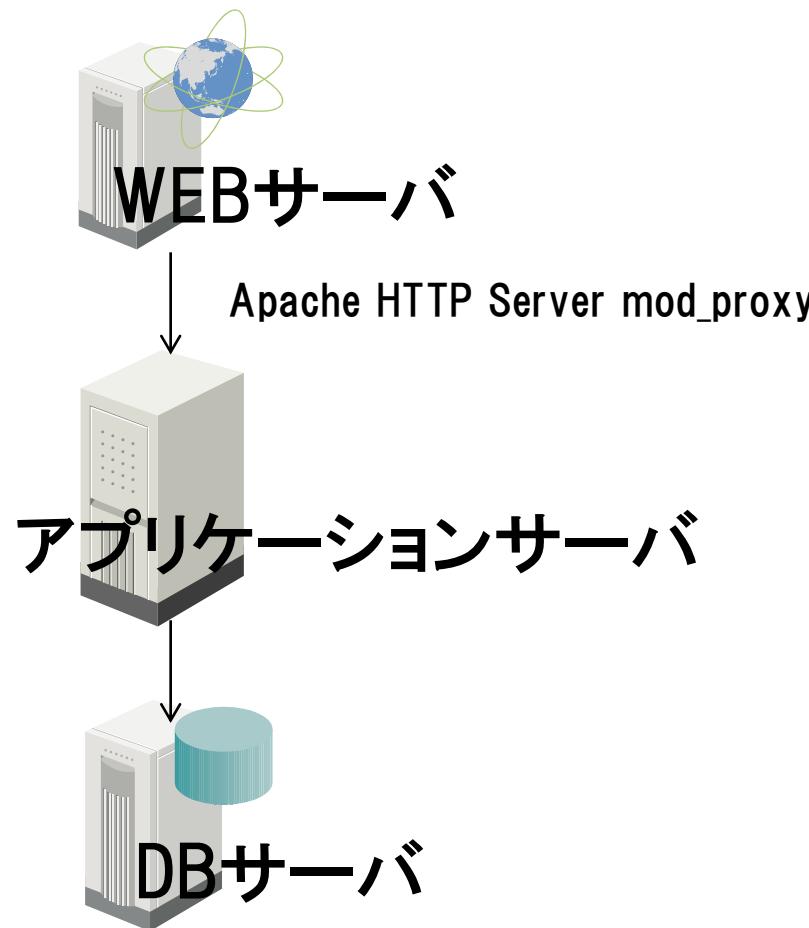
Linux で例えると、FedoraとRed Hat Enterprise Linux の関係になります。

## 3.JBossのインストールおよび初期設定

## JBossのインストールから基本的な設定の流れ

1. Javaのインストール
2. JBossのインストール
3. JBoss管理者ユーザの作成
4. JBoss サーバインスタンスの設定
5. データソースの設定
6. アプリケーションのデプロイ
7. WEBサーバとのコネクション設定

## 3-2 本日説明する環境の概要



Apache HTTP Server (OSバンドル)

**JBoss EAP 6.0**  
OpenJDK 1.7

JBoss AS7およびJBoss EAP6から、JBossの管理手法としてドメインという概念がありますが、今回は以前のバージョンに近いスタンドアローン構成でご説明します。

PostgreSQL Server (OSバンドル)

今回は1台のOS上で全て動作させます。

OSはRHEL6.4をセットアップ済みです。(最小構成をベース) (IPアドレス : 192.168.56.101)

DBサーバは予め設定を完了させています。

Apache HTTP Serverはインストール済みです。

JBossを利用するためにはJavaが必要です。  
今回は、Red Hat Enterprise Linux 6.4 に付属のOpenJDKを利用します。

#### 1. OpenJDKのインストール

```
# yum install java-1.7.0-openjdk
```

※ **java-1.5.0-gcj-1.5.0.0-29.1.el6.x86\_64**パッケージがインストールされている場合、javaコマンドが既にインストールされていますが、JBoss の実行基盤としては利用しないでください。

今回はRed Hat が提供しているJBoss Enterprise Application Platform (JBoss EAP) を利用します。

JBoss EAPのインストールはいくつか方法がありますが、

今回は最も汎用的なzipファイルでインストールします。

JBossのインストールは基本的に **zipファイルを解凍するだけ** です。

### 1. JBossの実行ユーザ(OSユーザ)の作成

```
# useradd jboss
```

### 2. zipファイルのダウンロードおよび配置

(今回、jboss-eap-6.0.0.zipを/optに配置済みです)

### 3. zipファイルの展開

```
# cd /opt
# unzip jboss-eap-6.0.0.zip
# chown -R jboss:jboss jboss-eap-6.0
```

## 4. 管理者ユーザの作成 (インストーラ版ではインストール時に作成します)

```
# su - jboss
$ /opt/jboss-eap-6.0/bin/add-user.sh
```

a) Management User (mgmt-users.properties) を選択し、  
ユーザ名とパスワードを入力します。

```
What type of user do you wish to add?
a) Management User (mgmt-users.properties)
b) Application User (application-users.properties)
(a): a
```

Enter the details of the new user to add.

Realm (ManagementRealm) :

Username : jboss

Password :

Re-enter Password :

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

Is this correct yes/no? yes

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

Added user 'jboss' to file '/opt/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? no

\$

JBossの設定はWEBベースの管理コンソール、CUIベースの管理クライアントを利用するか、xmlファイルを直接編集します。

サーバインスタンスの初期設定はxmlファイルを編集する必要がありますが、今回はデフォルトのスタンドアローン定義を利用します。

今回は設定に管理コンソールを利用します。

## 1. JBossの起動

```
$ /opt/jboss-eap6/bin/standalone.sh -b 0.0.0.0 -bmanagement=0.0.0.0  
-bオプションで公開用アドレスをバインドしています  
-bmanagementオプションで管理画面をバインドしています。
```

## 2. ブラウザから管理コンソールにアクセスして設定を行います。

(<http://192.168.56.101:9990>)

今回は特に設定は実施しません。

## 1. JDBCドライバのインストール

JDBCドライバはwarファイルのようなアプリケーションと同様にデプロイすることでインストールできます。

ドライバは管理コンソールのデプロイ画面からデプロイします。

1.RuntimeタブでServer → Manage Deploymentsを選択します。

2.Add Contextボタンをクリックし、Deployment Selectionの画面でJDBCドライバを選択し、[ Next ]ボタンをクリックします。

今回は、OSにバンドルのドライバを利用します。

今回の環境ではJDBCドライバはpostgresql-jdbc-8.4.701-8.el6.noarchパッケージに含まれているものを使用します。

実際のファイルは次のファイルになります。

/usr/share/java/postgresql-jdbc-8.4.701.jar

3.Verify Deployment Namesの画面で[Save]ボタンをクリックします。

4.Deployments画面でデプロイしたアプリケーション(JDBCドライバ)を有効化(Enableをクリック)します。

### 2. データソースの設定

管理コンソールを利用してデータソースの設定を行います。

1. ProfileタブでProfile → Connector → Datasourcesを選択します。

2. Datasourcesタブで [ Add ] ボタンをクリックします。

3. Datasource Attributes画面で以下の通り入力します

Name : SampleDS (名前は任意です)

JNDI Name : java:jboss/SampleDS

4. JDBC Driver画面で適切なJDBCドライバを選択し、[ Next ]ボタンをクリックします。

5. Connection Settingsの画面で以下のとおり入力し、[ Done ]ボタンをクリックします。

**Connection URL** : `jdbc:postgresql://localhost:5432/postgres`

localhost : DBサーバのホスト名

5432 : PostgreSQLのListenポート

postgres : データベース名

**Username** : `postgres`

**Password** : `secret`

6. JDBC Datasources画面で作成したデータソースを選択し、[ Enable ] ボタンをクリックします

7. SelectionメニューでConnectionタブを選択し、[ Test Connection ] ボタンをクリックします。

Successfullyと表示されればデータソースの定義は完了です。

管理コンソールからサンプルアプリケーションをデプロイします。  
今回、簡単なサンプルアプリケーションを事前に作成済みです。

アプリケーションのデプロイ手順はJDBCドライバのデプロイで説明済みです  
ので、割愛します。

JBossはWEBサーバの機能を持っていますが、実運用で利用する際は、専用のWEBサーバをフロントに置くことが一般的です。

WEBサーバと通信する場合、JBossはAJPプロトコルを利用します。  
まず、管理コンソールを利用してAJPコネクタを有効にします。

## 1. AJPコネクタの作成(JBoss側)

1. Profile → Web → Servlet/HTTPを選択します。

2. Servlet/HTTP Configuration画面のConnectorsタブ [ Add ] ボタンをクリックします。

3. Create Connector画面で以下の通り入力します

Name : ajp

Socket Binding : ajp

Protocol : AJP/1.3

Schema : http

Enabled? : チェック

JNDI Name :

## 2. mod\_proxyの設定(Apache側)

### 1. mod\_proxy用の設定ファイルを作成します

```
# echo "ProxyPass / ajp://localhost:8009/" >  
/etc/httpd/conf.d/mod_proxy_ajp.conf
```

今回の設定では、Apacheにきたリクエストを全てJBossに転送します。

### 2. Apache HTTP Server を起動(再起動)します

```
# service httpd start [ restart ]
```

#### 備考

mod\_proxyはApache HTTP Serverの標準モジュールです。

mod\_proxyの他に、mod\_jk (Apache Tomcatプロジェクト) や mod\_cluster (JBoss コミュニティ) があります。

アプリケーションの動作確認を行います。

今回のデモで使用するサンプルアプリケーションは、  
sampleテーブルからmessage列を取り出して画面に表示するだけの  
単純なアプリケーションです。

## 4. デモ(10分でセットアップ)

## 5. まとめ

- ✓ JBossは非常に簡単に利用できます
- ✓ 今回はOSおよびJBossにEnterprise版を利用していますが、  
Fedora(centosとかScientific Linuxとか)やJBoss ASであれば無償で利用で  
きるため、気軽に試すことができます。
- ✓ 今回は説明を簡易にするために、実運用では行わない設定  
や実施すべき設定についての説明を割愛しています。

実際に運用環境で利用する際は、様々な設定やチューニン  
グが必要です ← **重要**

**HITACHI**  
Inspire the Next<sup>®</sup>